



webMethods Oracle Applications Adapter 10.7SC Predefined Transaction Services User's Guide

VERSION 6.0

webMethods Administrator, webMethods Broker, webMethods Dashboard, webMethods Developer, webMethods Fabric, webMethods Glue, webMethods Installer, webMethods Integration Server, webMethods Mainframe, webMethods Manager, webMethods Mobile, webMethods Modeler, webMethods Monitor, webMethods Optimize, webMethods Portal, webMethods Trading Networks, webMethods Workflow, and the webMethods logo are trademarks of webMethods, Inc. "webMethods" is a registered trademark of webMethods, Inc.

Acrobat, Adobe, and Reader are registered trademarks of Adobe Systems Incorporated. Amdocs and ClarifyCRM are registered trademarks of Amdocs Ltd. Ariba is a registered trademark of Ariba Inc. BEA is a registered trademark, and BEA WebLogic Platform and BEA WebLogic Server are trademarks of BEA Systems, Inc. BMC Software and PATROL are registered trademarks of BMC Software, Inc. BroadVision is a registered trademark of BroadVision, Inc. Chem eStandards and CIDX are trademarks of Chemical Industry Data Exchange. Unicenter is a registered trademark of Computer Associates International, Inc. PopChart is a registered trademark of CORDA Technologies, Inc. Kenan and Arbor are registered trademarks of CSG Systems, Incorporated. SNAP-IX is a registered trademark, and Data Connection is a trademark of Data Connection Ltd. DataDirect, DataDirect Connect, and SequeLink are registered trademarks of DataDirect Technologies. D&B and D-U-N-S are registered trademarks of D&B, Inc. Entrust is a registered trademark of Entrust. Hewlett-Packard, HP, HP-UX, and OpenView are trademarks of Hewlett-Packard Company. i2 is a registered trademark of i2 Technologies, Inc. AIX, AS/400, CICS, DB2, IBM, Infoprint, Informix, MQSeries, OS/390, OS/400, RACF, RS/6000, SQL/400, S/390, System/390, VTAM, and WebSphere are registered trademarks; and Communications System for Windows NT, IMS, MVS, SQL/DS, Universal Database, and z/OS are trademarks of IBM Corporation. InnoDB is a trademark of Innobase Oy. JBoss and JBoss Group are trademarks of Marc Fleury under operation by JBoss Group, LLC. J.D. Edwards and OneWorld are registered trademarks, and WorldSoftware is a trademark of J.D. Edwards. Linux is a registered trademark of Linus Torvalds and others. X Window System is a trademark of Massachusetts Institute of Technology. MetaSolv is a registered trademark of Metasolv Software, Inc. ActiveX, Microsoft, Outlook, Visual Basic, Windows, and Windows NT are registered trademarks; and SQL Server is a trademark of Microsoft Corporation. MySQL is a registered trademark of MySQL AB. Teradata is a registered trademark of NCR. Netscape is a registered trademark of Netscape Communications Corporation. New Atlanta and ServletExec are trademarks of New Atlanta Communications, LLC. CORBA is a registered trademark of Object Management Group, Inc. UNIX is a registered trademark of Open Group. Oracle is a registered trademark of Oracle Corporation. PeopleSoft and Vantive are registered trademarks, and PeopleSoft Pure Internet Architecture is a trademark of PeopleSoft, Inc. Intranet and Portal are trademarks of Portal Software, Inc. RosettaNet is a trademark of "RosettaNet," a non-profit organization. SAP and R/3 are trademarks or registered trademarks of SAP AG. Siebel is a trademark of Siebel Systems, Inc. SPARC and SPARCStation are trademarks of SPARC International, Inc. SSA Global is a trademark and SSA Baan is a registered trademark of SSA Global Technologies, Inc. EJB, Enterprise JavaBeans, Java, Java Naming and Directory Interface, JavaServer Pages, JDBC, JSP, J2EE, Solaris, Sun Microsystems, and SunSoft are trademarks of Sun Microsystems, Inc. SWIFT and SWIFTNet are trademarks of S.W.I.F.T. SCRL. Sybase is a registered trademark of Sybase, Inc. UCCnet is a trademark of UCCnet. eBusinessReady is a trademark of Uniform Code Council, Inc. (UCC) and Drummond Group, Inc. (DGI). Verisign is a registered trademark of Verisign. VERITAS, VERITAS SOFTWARE, and VERITAS Cluster Server are trademarks of VERITAS Software. W3C is a registered trademark of World Wide Web Consortium.

All other marks are the property of their respective owners.

Copyright © 2004 by webMethods, Inc. All rights reserved, including the right of reproduction in whole or in part in any form.

Document ID: ADAPTER-ORACLEAPPS-107SCTUG-20040716

Contents

About This Guide	15
Document Conventions	15
Additional Information	16
Chapter 1. Predefined Transaction Services	17
Overview	18
Predefined Transaction Service Packages	20
Service Types	21
Packaging Structure	22
Transaction Definitions	25
Updating Transaction Definitions	26
Database Scripts	26
Core Database Scripts	27
Database Scripts for Receive Services	27
Database Scripts for Query and Send Services	28
Installing Components for Predefined Transaction Services on the Oracle Applications Database	28
Removing Components for Predefined Transaction Services from the Oracle Applications Database	29
Business Documents	30
Receive Services	31
Overview of Receive Service Transaction Processing	31
Using Receive Services	33
Optimizing Receive Services	34
Query Services	35
Overview of Query Service Transaction Processing	35
Using Query Services	36
Send Services	37
Overview of Send Service Transaction Processing	38
Using Send Services	42
Using Send Services in Debug Mode	43
Custom Tables Used with Send Services	43
WM_TRACKCHANGES Table	43
Example of How the WM_TRACKCHANGES Table is Populated	46
WM_CONTROL Table	47

Chapter 2. Financial Predefined Transaction Services	49
Overview	51
Query Accounts Payable Invoice Service	53
Database Scripts	53
Supporting Transaction Definitions	54
Flow Control	54
Business Document Structure	54
Query Accounts Payable Payment Service	54
Database Scripts	55
Supporting Transaction Definitions	55
Flow Control	55
Business Document Structure	55
Query Accounts Receivable Transactions Service	55
Database Scripts	56
Supporting Transaction Definition	56
Flow Control	56
Business Document Structure	57
Query Customer Service	57
Database Scripts	58
Supporting Transaction Definitions	58
Flow Control	58
Business Document Structure	58
Query General Ledger Balance Service	59
Database Scripts	59
Supporting Transaction Definitions	60
Flow Control	60
Business Document Structure	60
GLBalance	60
Query Journal Service	62
Database Scripts	63
Supporting Transaction Definitions	63
Flow Control	63
Business Document Structure	63
Query Vendor Service	64
Database Scripts	64
Supporting Transaction Definitions	64
Flow Control	65
Business Document Structure	65

Receive Accounts Payable Invoice Service	65
Database Scripts	66
Supporting Transaction Definitions	66
Flow Control	66
Business Document Structure	68
1.0 AP_INVOICE (Maps to AP_INVOICES_INTERFACE)	68
1.1 AP_INVOICE_LINES (Maps to AP_INVOICES_LINES_INTERFACE)	69
Receive Auto Invoice Service	71
Database Scripts	72
Supporting Transaction Definitions	72
Flow Control	72
Business Document Structure	74
1.0 TRANSACTIONS (Maps to RA_INTERFACE_LINES_ALL)	75
1.1 TRANSACTION_LINES (Maps to RA_INTERFACE_LINES_ALL)	79
1.1.1 TAX_LINES (Maps to RA_INTERFACE_LINES_ALL)	84
1.1.2 FREIGHT_LINES (Maps to RA_INTERFACE_LINES_ALL)	86
1.1.3 SALES_CREDITS (RA_INTERFACE_SALESCREDITS_ALL)	88
GL_DISTRIBUTIONS	90
1.2 CHARGES (Maps to RA_INTERFACE_LINES_ALL)	93
1.3 FREIGHT_LINES (RA_INTERFACE_LINES_ALL)	97
Receive Bank Statement Service	100
Database Scripts	101
Supporting Transaction Definitions	101
Flow Control	101
Business Document Structure	102
1.0 STATEMENT (Maps to CE_STATEMENT_HEADERS_INT_ALL)	102
1.1 STATEMENT_LINES (Maps to CE_STATEMENT_LINES_INTERFACE)	103
Receive Customer Service	104
Database Scripts	105
Supporting Transaction Definitions	105
Flow Control	105
Business Document Structure	106
1.0 CUSTOMERS (Maps to RA_CUSTOMERS_INTERFACE_ALL)	107
1.1 SITES (Maps RA_CUSTOMERS_INTERFACE_ALL)	108
1.1.1 SITE_USES (Maps to RA_CUSTOMERS_INTERFACE_ALL)	108
1.1.1.1 SITE_PAYMENT_METHODS (Maps to RA_CUST_PAY_METHOD_INT_ALL) ..	109
1.1.1.2 SITE_BANK_ACCOUNTS (Maps to RA_CUSTOMER_BANKS_INT_ALL)	109
1.1.1.3 SITE_PROFILES (Maps to RA_CUSTOMER_PROFILES_INT_ALL)	111
1.1.1.3.1 SITE_PROFILE_AMOUNTS (Maps to RA_CUSTOMER_PROFILES_INT_ALL)	113

1.1.2 SITE_PHONES (Maps to (Maps to RA_CONTACT_PHONES_INT_ALL)	113
1.1.3 SITE_CONTACTS (Maps to RA_CONTACT_PHONES_INT_ALL)	115
Receive FA Budget Service	115
Database Scripts	115
Supporting Transaction Definitions	116
Flow Control	116
Business Document Structure	116
FA_BUDGET (Maps to FA_BUDGET_INTERFACE)	116
Receive General Ledger Budget Service	118
Database Scripts	119
Supporting Transaction Definitions	119
Flow Control	119
Business Document Structure	120
GL_BUDGET (Maps to GL_BUDGET_INTERFACE)	120
Receive Journal Service	121
Database Scripts	121
Supporting Transaction Definitions	121
Flow Control	122
Business Document Structure	123
GL (Maps to GL_INTERFACE)	124
Receive Mass Additions Service	126
Database Scripts	126
Supporting Transaction Definitions	127
Flow Control	127
Business Document Structure	128
MASS_ADDITIONS_HEADER (Maps to FA_MASS_ADDITIONS)	128
Send Accounts Payable Invoice Service	131
Database Scripts	131
Supporting Transaction Definitions	132
Flow Control	132
Business Document Structure	133
1.0 AP_INVOICE	133
1.1 AP_INVOICE_LINES	136
Send Accounts Payable Payment Service	139
Database Scripts	139
Supporting Transaction Definitions	140
Flow Control	140
Business Document Structure	141
1.0 AP_CHECKS	141
1.1 AP_INVOICE_PAYMENTS	146

Send Accounts Receivable Transactions Service	148
Database Scripts	149
Supporting Transaction Definition	150
Flow Control	150
Business Document Structure	151
1.0 TRANSACTIONS	152
1.1 TRANSACTION_LINES	157
1.1.1 TAX_LINES	162
1.1.2 FREIGHT_LINES	165
1.2.3 SALES_CREDITS	167
1.1.4 GL_DISTRIBUTIONS	169
1.2 CHARGES	171
1.3 FREIGHT_LINES	175
Send Customer Service	179
Database Scripts	180
Supporting Transaction Definitions	181
Flow Control	182
Business Document Structure	183
1.0 CUSTOMERS	184
1.1 SITES	187
1.1.1 SITE_CONTACTS	189
1.1.1.1 SITE_CONTACT_ROLES	190
1.1.1.2 SITE_CONTACT_PHONES	190
1.1.2 SITE_USES	191
1.1.2.1 SITE_PAYMENT_METHODS	192
1.1.2.2 SITE_BANK_ACCOUNTS	193
1.1.2.3 SITE_PROFILES	195
1.1.2.3.1 SITE_PROFILE_AMOUNTS	197
1.7 CUSTOMER_RELATIONSHIPS	198
Send Journal Service	199
Database Scripts	199
Supporting Transaction Definitions	200
Flow Control	200
Business Document Structure	201
JOURNAL	201
Send Vendor Service	204
Database Scripts	204
Supporting Transaction Definitions	205
Flow Control	205

Business Document Structure	207
1.0 VENDORS	207
1.1 BANK_ACCOUNTS	213
1.2 VENDOR SITES	214
1.2.1 SITE_BANK_ACCOUNTS	219
1.2.2 VENDOR_CONTACTS	220
Chapter 3. Human Resources Predefined Transaction Services	221
Overview	222
Receive Employee Service	223
Database Scripts	223
Supporting Transaction Definitions	224
Flow Control	224
Business Document Structure	225
1.0 EMPLOYEE	225
1.1 EMPLOYEE_ASSIGNMENTS	227
Chapter 4. Manufacturing Predefined Transaction Services	231
Overview	233
Query BOM Revision Service	234
Database Scripts	234
Supporting Transaction Definitions	234
Flow Control	235
Business Document Structure	235
BOM_REVISIONS	235
Query BOM Routing Service	236
Database Scripts	236
Supporting Transaction Definitions	237
Flow Control	237
Business Document Structure	237
1.0. BOM_PO_ROUTINGS	237
1.1. MTL_RTG_ITEM_REVS	238
1.2. BOM_OP_SEQUENCES	239
1.2.1. BOM_OP_RESOURCES	241
Query Engineering BOM Service	243
Database Scripts	243
Supporting Transaction Definitions	243
Flow Control	244
Business Document Structure	244

Query Item Service	244
Database Scripts	244
Supporting Transaction Definitions	245
Flow Control	245
Business Document Structure	245
Query Item Category Service	245
Database Scripts	246
Supporting Transaction Definitions	246
Flow Control	246
Business Document Structure	247
ITEM_CATEGORY	247
Query Manufacturing BOM Service	247
Database Scripts	248
Supporting Transaction Definitions	248
Flow Control	248
Business Document Structure	248
Query On-Hand Quantity Service	249
Database Scripts	249
Supporting Transaction Definitions	249
Flow Control	250
Business Document Structure	250
ON_HAND_QUANTITY	250
Receive Bills and Route Service	252
Database Scripts	252
Supporting Transaction Definitions	252
Flow Control	253
Business Document Structure	254
1.0. BOM_BILL_OF_MTLS (Maps to BOM_BILL_OF_MTLS_INTERFACE)	255
1.1. MTL_ITEM_REVISIONS (Maps to MTL_ITEM_REVISIONS_INTERFACE)	256
1.1.1. BOM_INVENTORY_COMPONENTS (Maps to BOM_INVENTORY_COMPS_INTERFACE)	256
1.1.1.1. BOM_SUB_COMPS (Maps to BOM_SUB_COMPS_INTERFACE)	260
1.1.1.2. BOM_REF_DESGS (Maps to BOM_REF_DESGS_INTERFACE)	261
2.0. BOM_OP_ROUTINGS (Maps to BOM_OP_ROUTINGS_INTERFACE)	262
2.1. MTL_RTG_ITEM_REVS (Maps to MTL_RTG_ITEM_REVS_INTERFACE)	263
2.2. BOM_OP_SEQUENCES (Maps to BOM_OP_SEQUENCES_INTERFACE)	264
2.2.1. BOM_OP_RESOURCES (Maps to BOM_OP_RESOURCES_INTERFACE)	265
Receive Customer Item Service	266
Database Scripts	267
Supporting Transaction Definitions	267

Flow Control	267
Business Document Structure	269
CUST_ITEM (Maps to MTL_CI_INTERFACE)	269
Receive Customer Item Cross Reference Service	271
Database Scripts	271
Supporting Transaction Definitions	272
Flow Control	272
Business Document Structure	274
CUST_ITEM_CROSS_REF (Maps to MTL_CI_XREFS_INTERFACE)	274
Receive Inventory Transaction Service	275
Database Scripts	275
Supporting Transaction Definitions	276
Flow Control	277
Business Document Structure	279
1.0. MTL_TRANSACTIONS (Maps to MTL_TRANSACTIONS_INTERFACE)	280
1.1. MTL_SERIAL_NUMBERS (Maps to MTL_SERIAL_NUMBERS_INTERFACE)	283
1.2. MTL_TRANSACTION_LOTS (MTL_TRANSACTION_LOTS_INTERFACE)	283
1.2.1. MTL_SERIAL_NUMBERS (MTL_SERIAL_NUMBERS_INTERFACE)	283
Receive Item Service	284
Database Scripts	285
Supporting Transaction Definitions	285
Flow Control	285
Business Document Structure	289
1.0. MTL_SYSTEM_ITEMS (Maps to MTL_SYSTEM_ITEMS_INTERFACE)	289
1.1. MTL_ITEM_REVISIONS (Maps to MTL_ITEM_REVISIONS_INTERFACE)	302
Receive Master Schedule Service	303
Database Scripts	303
Supporting Transaction Definitions	303
Flow Control	304
Business Document Structure	305
MRP_SCHEDULE (Maps to MRP_SCHEDULE_INTERFACE)	305
Receive Move Transaction Service	306
Database Scripts	306
Supporting Transaction Definitions	306
Flow Control	307
Business Document Structure	308
WIP_MOVE_TXN (Maps to WIP_MOVE_TXN_INTERFACE)	308
Receive Quality Collection Service	310
Database Scripts	310
Supporting Transaction Definitions	311

Flow Control	311
Business Document Structure	312
QA_COLLECTION (Maps to QA_RESULTS_INTERFACE)	312
Receive Replenishment Service	314
Database Scripts	315
Supporting Transaction Definitions	315
Flow Control	315
Business Document Structure	317
1.0. MTL_REPLENISH_HEADERS (Maps to MTL_REPLENISH_HEADERS_INT)	317
1.1. MTL_REPLENISH_LINES (Maps to MTL_REPLENISH_LINES_INT)	319
Receive Resource Transaction Service	320
Database Scripts	320
Supporting Transaction Definitions	320
Flow Control	321
Business Document Structure	322
WIP_COST_TXN (Maps to WIP_COST_TXN_INTERFACE)	322
Send Engineering BOM Service	325
Database Scripts	325
Supporting Transaction Definitions	326
Flow Control	326
Business Document Structure	328
1.0. BOM_ENG_BILL_OF_MTLS	328
1.1. MTL_ENG_ITEM_REVISIONS	329
1.2. BOM_ENG_INVENTORY_COMPS	330
1.2.1. BOM_ENG_SUB_COMPS	333
1.2.2. BOM_ENG_REF_DESGS	335
Send Item Service	336
Database Scripts	336
Supporting Transaction Definitions	337
Flow Control	337
Business Document Structure	338
1.0. MTL_SYSTEM_ITEM	338
1.1. MTL_ITEM_CATEGORIES	357
1.1.1. MTL_ITEM_REVISIONS	357
Send Manufacturing BOM Service	359
Database Scripts	359
Supporting Transaction Definitions	360
Flow Control	360

Business Document Structure	361
1.0. BOM_BILL_OF_MTLS	362
1.1. MTL_ITEM_REVISIONS	363
1.2. BOM_INVENTORY_COMPS	364
1.2.1. BOM_SUB_COMPS	367
1.2.2. BOM_REF_DESGS	368
Chapter 5. Order Management Predefined Transaction Services	371
Overview	372
Query Pick Detail Service	373
Database Scripts	373
Supporting Transaction Definitions	373
Flow Control	374
Business Document Structure	374
Query Sales Order Service	374
Database Scripts	374
Supporting Transaction Definitions	374
Flow Control	375
Business Document Structure	375
Receive Sales Order Service	375
Database Scripts	376
Supporting Transaction Definitions	376
Flow Control	376
Business Document Structure	378
1.0 ORDER_HEADERS (Maps to SO_HEADERS_INTERFACE)	378
1.1 ORDER_LINES (Maps to SO_LINES_INTERFACE)	382
1.1.1 PRICE_ADJ (Maps to SO_PRICE_ADJUSTMENTS_INTERFACE)	384
1.1.2 SALES_CREDITS (Maps to SO_SALES_CREDITS_INTERFACE)	385
1.1.3 DETAILS (Maps to SO_LINE_DETAILS_INTERFACE)	385
1.2 HEADER_PRICE_ADJ (Maps to SO_PRICE_ADJUSTMENTS_INTERFACE)	386
1.3. HEADER_SALES_CREDIT (Maps to SO_SALES_CREDITS_INTERFACE)	387
Send Pick Detail Service	387
Database Scripts	388
Supporting Transaction Definitions	388
Flow Control	388
Business Document Structure	389
1.1 PICKING_LINES	392
Send Sales Order Service	394
Database Scripts	395
Supporting Transaction Definitions	396
Flow Control	396

Business Document Structure	397
1.0 ORDER_HEADERS	397
1.1 ORDER_LINES	402
1.1.1 PRICE_ADJ	404
1.1.2 SALES_CREDITS	405
1.1.3 DETAILS	406
1.2 HEADER_PRICE_ADJ	406
1.3 HEADER_SALES_CREDIT	407
Chapter 6. Procurement Predefined Transaction Services	409
Overview	410
Query Product Catalog Service	411
Database Scripts	411
Supporting Transaction Definitions	412
Flow Control	412
Business Document Structure	412
1.0. SUPPLIER_ITEMS	412
1.1. NEGOTIATED_SOURCES	413
1.2. PRIOR_PURCHASES	415
1.3. SOURCING_RULES	416
1.3.1. SOURCING_DOCUMENTS	417
Query Purchase Order Service	418
Database Scripts	419
Supporting Transaction Definitions	419
Flow Control	419
Business Document Structure	419
Query Purchase Order Status Service	420
Database Scripts	421
Supporting Transaction Definitions	421
Flow Control	421
Business Document Structure	422
PURCHASE_ORDER	422
Query Supplier Service	424
Database Scripts	424
Supporting Transaction Definitions	424
Flow Control	425
Business Document Structure	425
Receive Requisition Service	425
Database Scripts	426
Supporting Transaction Definitions	426
Flow Control	426

Business Document Structure	429
1.0. PO_REQUISITIONS (Maps to PO_REQUISITIONS_INTERFACE_ALL)	429
1.1. PO_REQ_LINES (Maps to PO_REQUISITIONS_INTERFACE_ALL)	430
1.1.1. PO_REQ_DISTRIBUTIONS (Maps to PO_REQUISITIONS_INTERFACE_ALL) ...	433
Send Purchase Order Service	434
Database Scripts	434
Supporting Transaction Definitions	435
Flow Control	435
Business Document Structure	436
1.0. PO_HEADERS	436
1.1. PO_LINES	441
1.1.1. PO_LINE_LOCATIONS	445
1.1.1.1. PO_DISTRIBUTIONS	447
Send Supplier Service	451
Database Scripts	452
Supporting Transaction Definitions	453
Flow Control	453
Business Document Structure	454
1.0. SUPPLIER	454
1.1. BANK_ACCOUNTS	461
1.2. SUPPLIER_SITES	462
1.2.1. SITE_BANK_ACCOUNTS	469
1.2.2. SUPPLIER_CONTACTS	470
Chapter 7. Project Predefined Transaction Services	471
Overview	472
Receive Labor Service	473
Database Scripts	473
Supporting Transaction Definitions	474
Flow Control	474
Business Document Structure	475
LABOR_BIZDOC (Maps to PA_TRANSACTION_INTERFACE_ALL)	475
Index.....	481

About This Guide

This guide describes the predefined transaction services for Oracle Applications 10.7SC that are available for use with the webMethods Oracle Applications Adapter. This guide provides a general overview of these services, and instructions for using the services. This information is for application developers who implement integrations between webMethods and Oracle Applications.

To use this guide effectively, you should:

- Be familiar with the operation of the desired Oracle Applications and the definition and use of the Oracle Applications Adapter.
- Understand the basic concepts described in the *webMethods Administrator's Guide* and the *webMethods Developer User's Guide*.

Document Conventions

Convention	Description
Bold	Identifies elements on a screen.
<i>Italic</i>	Identifies variable information that you must supply or change based on your specific situation or environment. Identifies terms the first time they are defined in text. Also identifies service input and output variables.
Narrow font	Identifies storage locations for services on the webMethods Integration Server using the convention <i>folder.subfolder.service</i> .
Typewriter font	Identifies characters and values that you must type exactly or messages that the system displays on the console.
UPPERCASE	Identifies keyboard keys. Keys that you must press simultaneously are joined with the "+" symbol.
\	Directory paths use the "\" directory delimiter unless the subject is UNIX-specific.
[]	Optional keywords or values are enclosed in []. Do not type the [] symbols in your own code.

Additional Information

The webMethods Advantage Web site at <http://advantage.webmethods.com> provides you with important sources of information about the webMethods Integration Server:

- **Troubleshooting Information.** webMethods provides troubleshooting information for many webMethods components in the [webMethods Knowledge Base](#).
- **Documentation Feedback.** To provide documentation feedback to webMethods, go to the [Documentation Feedback Form](#) on the [webMethods Bookshelf](#).
- **Additional Documentation.** All webMethods documentation is available on the [webMethods Bookshelf](#).

Predefined Transaction Services

■ Overview	18
■ Receive Services	31
■ Query Services	35
■ Send Services	37

Overview

Predefined transaction services are complete, ready-to-use flow services that can be used with the webMethods Oracle Applications Adapter to simplify the process of integrating with Oracle Applications systems. After installing the services and defining your Oracle Applications system connection information, the services are ready for use.

There are three types of predefined transaction services used to communicate between the webMethods Integration Server (IS) and Oracle Applications: Receive services, Query services, and Send services.

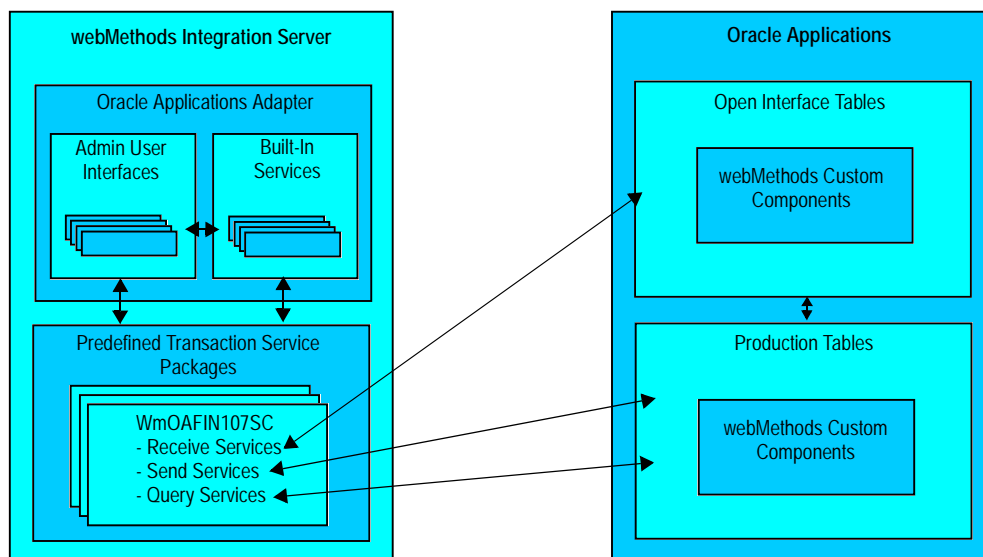
The Receive predefined transaction services enable Oracle Applications to receive data from the IS. These services provide the following:

- Required inputs and outputs for the services
- Mapping and transformation between business documents and Oracle Applications
- Supporting services that perform the data insertion, import (concurrent) process execution, and error handling

The Query and Send predefined transaction services enable you to query or send data from Oracle Applications to the IS. These services provide the following:

- Required inputs and outputs for the services
- Combination of database triggers and polling services that provide notification capabilities from Oracle Applications to IS
- Supporting services to lock and unlock the webMethods process table, and to perform data selection and acknowledgement from Oracle Applications to the IS

The following diagram shows in more detail how the Oracle Applications Adapter works with predefined transaction services.



- **Oracle Applications Adapter.** The adapter is delivered in a package named WmOAAAdapter. The WmOAAAdapter package includes:
 - Administrative user interfaces to configure and manage different adapter connections. Using these interfaces you can:
 - Configure the login parameters that the Oracle Applications Adapter uses to establish a client connection with one or more Oracle Applications.
 - Schedule the notification.
 - Built-in services that provide the basic means to interact with Oracle Applications.

The built-in services perform activities such as inserting and updating data, and managing database connections. The predefined transaction services wrap the built-in functions and make them accessible to the webMethods Integration Server.

- **Predefined Transaction Service Packages.** The predefined transaction services are delivered in separate packages. Each package contains a set of services for a particular Oracle Applications module and version of Oracle Applications. These packages contain the following types of predefined transaction services:
 - Receive services, which deliver data to the Oracle Applications open interface tables for insertion into the production tables.

- Query services, which query the Oracle Applications production tables and return business documents that match the query criteria.
- Send services, which return business documents to the Integration Server according to polling intervals. Send services rely on custom webMethods components such as triggers and custom tables to capture business events on the Oracle Applications production tables, and then send the business documents to the Integration Server.
- **Open Interface Tables and Production Tables.** The predefined transaction services interact with the Oracle Applications open interface tables and production tables to send or receive business documents. Receive, send, and query services interact with these tables differently.
- **webMethods Custom Components.** The predefined transaction services use a combination of triggers, stored procedures, custom tables, views, and sequences (depending on the type of service) to interact with the Oracle Applications open interface tables and production tables.

This chapter describes the predefined transaction services, including how they are packaged, and how you use them to receive data into, or send data from, Oracle Applications systems.

The information in this chapter is intended to help you understand the basic design and functionality of any of the predefined transaction services. After you read this chapter, you can use the information provided in the other chapters of this guide to implement any of the predefined transaction services.

Predefined Transaction Service Packages

The predefined transaction services for Oracle Applications 10.7SC are provided in six packages. The packages correspond to Oracle Applications modules, such as Financials and Manufacturing. You can choose which packages you want to install at the time of installation. The following table lists the packages, and shows you where to find detailed information about the services provided in those packages.

Package	See ...	On Page ...
Financials	Chapter 2, “Financial Predefined Transaction Services”	49
Human Resources	Chapter 3, “Human Resources Predefined Transaction Services”	221
Manufacturing	Chapter 4, “Manufacturing Predefined Transaction Services”	231
Order Management	Chapter 5, “Order Management Predefined Transaction Services”	371

Package	See ...	On Page ...
Procurement	Chapter 6, “Procurement Predefined Transaction Services”	409
Project	Chapter 7, “Project Predefined Transaction Services”	471

Service Types

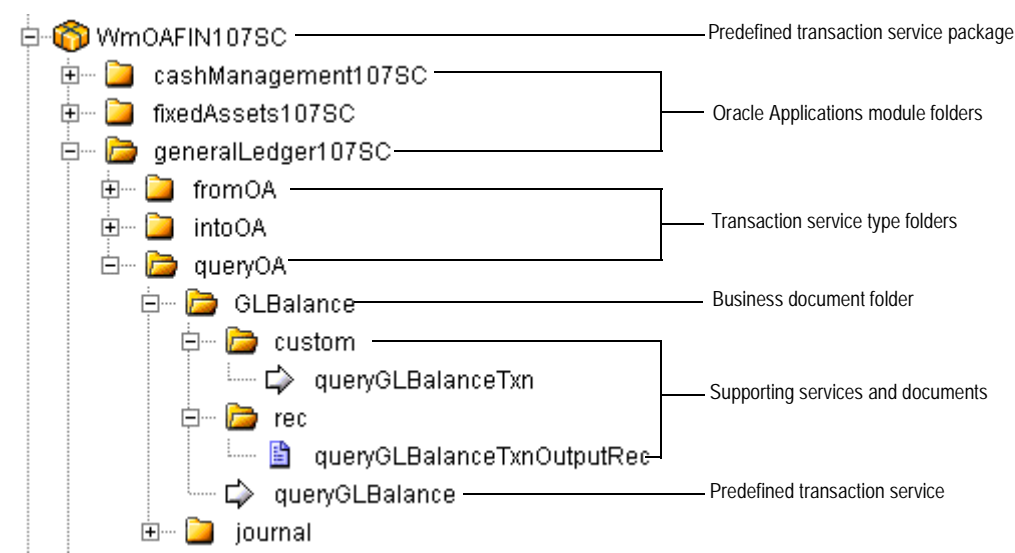
There are three types of predefined transaction services that enable you to build integrations between Oracle Applications systems and the webMethods Integration Platform. The following table briefly describes the types of services and directs you to the sections of this chapter that describe each type of service in more detail.

Service Type	Description	See page ...
Receive	Receive services enable Oracle Applications systems to receive and insert data from the webMethods Integration Platform.	31
Query	Query services enable the webMethods Integration Platform to query an Oracle Applications system and retrieve data based on the query conditions.	35
Send	Send services enable you to notify the webMethods Integration Platform of specific events that occur in your Oracle Applications system, and then send data related to the events to trading partners or other enterprise systems.	37

Packaging Structure

This section describes the transaction service packages as they appear in the webMethods Developer.

This diagram illustrates the package structure for the predefined transaction services:



- **Predefined Transaction Service Packages.** The predefined transaction services are provided in the form of packages. Each package represents a group of Oracle Applications modules.

The adapter provides the following packages:

Package	Abbreviation	Oracle Applications Product
WmOAFIN107SC	FIN	Financials
WmOAHR107SC	HR	Human Resources
WmOAMFG107SC	MFG	Manufacturing
WmOAOMG107SC	OMG	Order Management
WmOAPRC107SC	PRC	Procurement
WmOAPRJ107SC	PRJ	Project
WmOACCommon107SC	Common	None. This package accompanies any of the other installed packages. It includes services such as lockTxnCtrl that other packages use and that you can view in the webMethods Developer.

- **Oracle Applications module folders.** Each package contains one or more folders that correspond to Oracle Applications modules. For example, the WmOAFIN107SC package contains folders for the following modules: Cash Management, Fixed Assets, General Ledger, Payables, and Receivables.
- **Transaction service type folders.** The Oracle Applications module folders contain the three types of predefined transaction services (if applicable) within the package, as follows:

Folder	Service Type
intoOA	Receive Service
fromOA	Send Service
queryOA	Query Service

- **Business document folder.** The transaction service type folders contain folders that correspond to business documents for the Oracle Applications module. There is a sub-folder for each business document in the Oracle Applications module that is supported by the Oracle Applications Adapter. For example, in the General Ledger module of the WmOAFIN107SC package, the queryOA folder contains two business document folders: GLBalance and journal.

The business document folders contain the predefined transaction services and other supporting services and documents.

- **Predefined transaction service.** The services located within the business document folders are the main predefined transaction services. These are the services you implement to create integrations with Oracle Applications systems.

The services follow the naming convention *typeDoc*, where *type* is type of service (receive, query, or send) and *Doc* is the name of the business document.

- **Supporting services and documents.** Each business document folder contains supporting services and documents that are used internally by the main predefined transaction services. The specific documents and supporting services vary depending on the type of service, as described in the sections below.

The information provided in these sections is a high-level description of the types of supporting services and documents that are used by the different types of predefined transaction services. For more specific details about a specific service, see the section for that service in the appropriate chapter of this guide.

In the following descriptions, *Doc* represents the actual name of the business document that the service implements. You can customize the services in the custom folder if the default settings do not meet your needs.

- intoOA folders are structured as follows:

Folder	Supporting documents or services
custom	<p><i>bizDocMapping</i> This service maps the high-level business document structure to the open interface table structure.</p> <p><i>getDocImport_ERR</i> This service retrieves processing errors related to the import process (if applicable).</p> <p><i>setDocTxn</i> This service inserts the data from the business document into the open interface tables.</p>
rec	<p><i>DocBizDoc</i> This document defines the high-level business document structure.</p> <p><i>setDocTxnInputRec</i> This document defines the low-level document structure, which is a direct reflection of the corresponding open interface tables.</p>
utils	<p>The contents of this folder varies for each predefined transaction service. For example:</p> <p><i>importDoc</i> Runs the services listed below.</p> <p><i>execDocConcProg</i> Runs the Oracle Applications concurrent processing for the business document.</p> <p><i>checkDocImportStatus</i> Checks the status of the concurrent processing. If it detects errors, it calls <i>getDocImport_ERR</i>.</p>

For information about the functionality of the receive services, see [“Overview of Receive Service Transaction Processing” on page 31](#).

- queryOA folders are structured as follows:

Folder	Supporting documents or services
custom	queryDocTxn This service retrieves data based on the query conditions.
rec	queryDocTxnOutputRec This document type contains the data structure retrieved by the queryDocTxn service.

For information about the functionality of the query services, see [“Query Services” on page 35](#).

- sendOA folders are structured as follows:

Folder	Supporting documents or services
custom	getDocTxn This service polls for events in the Oracle Applications system. processBizDoc This service defines how the service delivers the data. You must customize this service to define where the data is delivered; otherwise, the service will not deliver any documents, and the events will be returned to the WM_TRACKCHANGES table to be processed again.
rec	getDocTxnOutputRec This document type contains the data structure retrieved by the getDocTxn service.

For information about the functionality of the send services, see [“Overview of Send Service Transaction Processing” on page 38](#).

Transaction Definitions

Transaction definitions define the processing that the predefined transaction services perform. Some of the supporting services within the predefined transaction services have been configured to use transaction definitions.

If you are using the predefined transaction services as they were delivered, you do not need to do anything with the transaction definitions. However, if you want to customize any of the transaction definitions that are used within the supporting services, or if you want to create new transaction definitions, you can do so using the Adapters menu in the Integration Server Administrator. See [“Updating Transaction Definitions” on page 26](#) for more information.

See the *webMethods Oracle Applications Adapter User's Guide* for more information about transaction definitions, including instructions for updating them and using them to configure services.

Updating Transaction Definitions

You can modify the transaction definitions delivered with the predefined transaction services. For example, you might want to customize a transaction definition for a receive service to optimize its performance if you do not need to take advantage of all of the processing the service performs, or if the service delivers more business document fields than you need.



Important! The predefined transaction services were tested to guarantee they function properly as delivered. If you modify any supporting transaction definitions, you will also need to modify other components within the predefined transaction service to match the changes to the transaction definitions. After you modify the service, you will need to thoroughly test it. We cannot guarantee that any services you modify will work properly.

The transaction definitions are provided as .txp files, and are installed into the WmOAXXXnnnn\exchange directories, where WmOAXXXnnnn is the name of the package containing the predefined transaction services.

To see a list of the transaction definition files that were used to configure supporting services within a predefined transaction service, see the Supporting Transaction Definitions section for the specific service in this guide.



Note: All .txp files in the *IntegrationServer_Directory*\packages\WmOAAadapter\exchange directory will be imported automatically when the Integration Server starts or is restarted.

After you import the transaction definition files, you can use the Adapters menu in the Administrator to modify them. The procedures for importing and editing transaction definitions are not described in this guide. See the chapter on transaction definitions in the *webMethods Oracle Applications Adapter User's Guide* for instructions.



Note: If you import transaction definitions after enabling the corresponding predefined transaction service package, you need to reload the predefined transaction service package so that the imported transaction definitions are visible.

Database Scripts

Database scripts are provided for each of its predefined transaction services. You use these database scripts to manage the triggers, views, and stored procedures that are used by the predefined transaction services.

The database scripts are located in the WmOAXXXnnnn\dbscripts directories, where WmOAXXXnnnn is the name of the package containing the predefined transaction services.

Each type of predefined transaction service has a set of database scripts, as described in the following sections.

Core Database Scripts

The predefined transaction services use a set of core components that you must install and manage on the Oracle Applications system to use any of the predefined transaction services.

The database scripts for these core components are located in the WmOACCommon107SC\dbscripts directory.

Database Script File	Description
wm_install_core.sql	Use this script to install the core components of the predefined transaction services. You must install these components to use any of the predefined transaction services.
wm_drop_core.sql	Use this script to uninstall the core components of the predefined transaction services from the Oracle Applications system.

Database Scripts for Receive Services

Each receive service provides the following database scripts:

Database Script File	Description
wm_install_into_Doc.sql	Use this script to install the Oracle Applications-related components of the service (such as triggers, views, and stored procedures) onto the Oracle Applications system.
wm_drop_into_Doc.sql	Use this script to uninstall all components of the service from the Oracle Applications system.

Each service might also have supporting database scripts. You do not have to run these scripts directly; they are initiated from the database scripts listed above.

Database Scripts for Query and Send Services

Query and send services that operate on the same business object share the following database scripts.

Database Script File	Description
wm_install_from_Doc.sql	Use this script to install all components of the service onto the Oracle Applications system.
wm_drop_from_Doc.sql	Use this script to uninstall all components of the service from the Oracle Applications system.
wm_enable_from_Doc.sql	Use this script to enable all triggers associated with the service. Use this script if you have disabled the triggers. By default, triggers are enabled when you install them.
wm_disable_from_Doc.sql	<p>Use this script to disable all triggers associated with the service. Use this script if you are using the query service but not its associated send service.</p> <p>If you do not disable triggers for send services that you are not using, the custom tables will continue to grow in your Oracle Applications system, consuming resources.</p>

Each service might also have supporting database scripts. You do not have to run these scripts directly; they are initiated from the database scripts listed above.

Installing Components for Predefined Transaction Services on the Oracle Applications Database

According to the Oracle Applications customization standards, you should create views, triggers, and packages under APPS schema. Tables, indexes, and sequences should be created under Custom User schema, and private synonyms for these tables and sequences should be created under APPS schema.

Typically each schema should have a corresponding data tablespace and index tablespace. You can use any tablespace name for where you intend to store the tables (for example, WM_TRACKCHANGES) and indexes (if there are any).

Before running the database scripts, create a custom user and a corresponding data tablespace and index tablespace. We suggest a user name of WEBM, a data tablespace name of WEBMD, and an index tablespace name of WEBMI. However, you can also use an existing data tablespace and index tablespace.

Whenever you execute database scripts to install objects in the database, you are prompted to supply an APPS username/password, a Custom username/password, custom tablespaces (for data and index), and a connection string.

Note that Custom user schema and APPS schema should be different.

The required grants and synonyms are already incorporated in the database scripts.

To install the Oracle Applications components of the predefined transaction services

- 1 Change your default directory to the *IntegrationServer_Directory*\packages\WmOAXXX107SC\dbscripts directory, where WmOAXXX107SC is the name of the Oracle Applications package containing the service for which you want to run database scripts.
- 2 Log in to SQL*Plus. You can use any login.
- 3 From SQL*Plus, run the installation scripts. When prompted, supply the APPS username/password, a Custom username/password, custom tablespaces (for data and index), and a connection string:
 - a Install the core components for the 10.7SC predefined transaction services by running the following script:

```
@wm_install_core.sql
```
 - b Run the database script for each of the predefined transaction services you are using. For the name of a predefined transaction service's installation script, see the section called Database Scripts for the specific service in the remaining chapters in this guide.



Note: Make sure you enter the APPS user and its associated password. In addition, the procedures and views being created must be in the APPS schema.

- 4 Exit SQL*Plus.

Removing Components for Predefined Transaction Services from the Oracle Applications Database

To remove the predefined transaction service components from the Oracle Applications database

- 1 Change your default directory to the *IntegrationServer_Directory*\packages\WmOAXXX107SC\dbscripts directory, where WmOAXXX107SC is the name of the Oracle Applications package containing the service for which you want to run database scripts.
- 2 Log in to SQL+. You can use any login.

- 3 From SQL*Plus, run the uninstall scripts. When prompted, supply the APPS username/password, a Custom username/password, custom tablespaces (for data and index), and a connection string:
 - a Uninstall the core components for the 10.7SC predefined transaction services by running the following script:

```
@wm_drop_core.sql
```
 - b Run the uninstall database script for each of the predefined transaction services you are using. For the name of a predefined transaction service's uninstall script, see the section called Database Scripts for the specific service in the remaining chapters in this guide.
- 4 When the uninstallation is complete, you will see a message indicating the completion of the uninstallation.
- 5 Exit SQL*Plus.

Business Documents

Each predefined transaction service is based on an Oracle Applications business object. Each service provides a business document structure that is based on the service's associated business object. The sections in this guide that describe the individual predefined transaction services include descriptions of the services' business document structures, and mappings to webMethods documents (if applicable).

For receive services, the business document structure includes a mapping from a webMethods document to the structure of the service's business object as it is defined in the Oracle Applications open interface tables. If you want to change the mapping for a receive service, you can modify the structure of the webMethods document by updating the receive service's supporting bizDocMapping service; however, you cannot modify the structure of the side of the mapping that represents the Oracle Applications open interface tables. If you do modify the open interface table document structure, the service will no longer correctly map to Oracle Applications.

For send and query services, the business document structure represents the form of the document as it is retrieved from Oracle Applications. If you want to modify the business document structure for a send or query service, you must change the transaction definition for the service's supporting *getDocTxn* or *queryDocTxn* service, refresh that service for the modified transaction definition, and then update the send or query service to reflect the change in the business document's structure.



Important! Query and send services are fully tested using the services' default business document structure. If you modify the business document structure for a service, you will need to test the service.

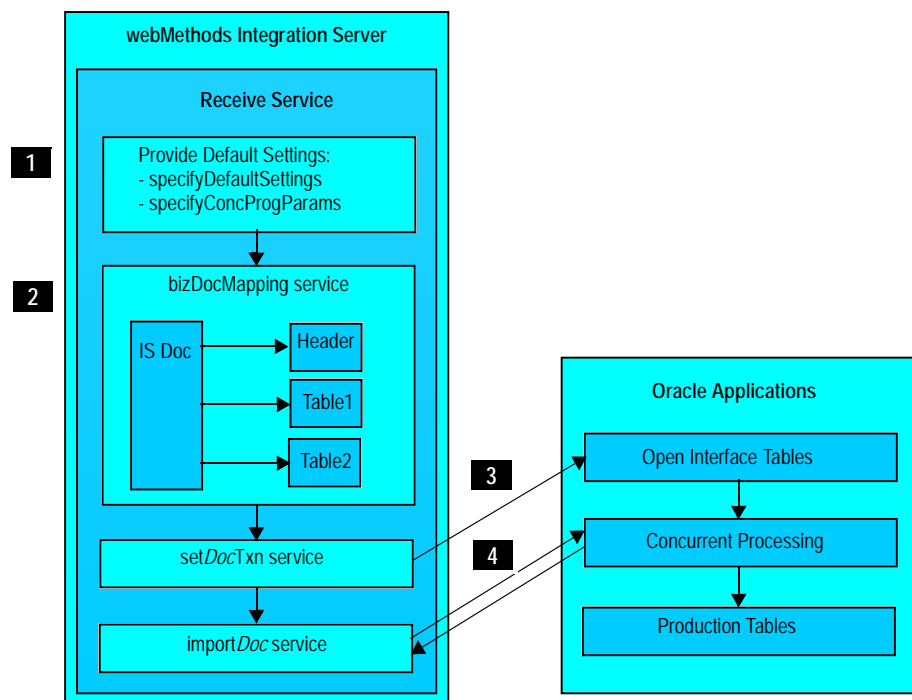
Receive Services

Receive services insert data from the IS into Oracle Applications systems by mapping business documents into Oracle Applications open interface tables, and then invoking Oracle Applications concurrent processing to import the data from the open interface tables to production tables.

You can configure receive services to submit the data to the Oracle Applications production tables automatically, or you can configure them so that they must be submitted manually. If you configure the receive service to submit data automatically, it queries the Oracle Applications system to determine whether there were any errors that occurred during the import process.

Overview of Receive Service Transaction Processing

Receive services insert data into Oracle Applications production tables by calling the following services, where *Doc* is the name of the business document:



Step	Description
1	<p>Specify Default Settings: Receive services use a set of high-level settings to define how the services will execute. The default settings should be adequate for most cases. However, you can modify the default settings if necessary, as follows:</p> <ul style="list-style-type: none">■ In the <code>specifyDefaultSettings</code> MAP step, the <code>autoMode</code> parameter defines whether the receive service automatically submits the business document data to the Oracle Applications production tables using Oracle Applications concurrent processing, or whether you must submit the data manually. <p>A value of <code>TRUE</code> indicates that the data will be submitted automatically. If you choose this setting, the <code>importDoc</code> service calls the <code>execDocConcProg</code> service to execute concurrent programs and return any errors, if applicable. This setting is the default.</p> <p>A value of <code>FALSE</code> indicates that you must submit the data manually using Oracle Applications concurrent processing. You can either execute the concurrent program from the Oracle Applications Standard Request Submission (SRS) window, or you can schedule it using a scheduling program.</p> <ul style="list-style-type: none">■ In the <code>specifyConcProgParams</code> MAP step you can define settings for the Oracle Applications concurrent processing execution, if applicable. By default, the concurrent processing parameters are configured for the Oracle Applications demo database. Refer to your Oracle Applications documentation for information about concurrent processing options for specific business objects. To see the parameters that are defined for a specific service, review the flow service. <hr/> <p>Note: In version 3.0 of the adapter, the <code>\$dbAlias</code> parameter in the <code>specifyDefaultSettings</code> MAP step could be modified to specify the connection the service will use. In version 6.0 of the adapter, that is no longer true. The <code>\$dbAlias</code> parameter is still available however it is no longer used to control a service's connection and cannot be modified.</p> <hr/>
2	<p>bizDocMapping Service: The <code>bizDocMapping</code> service maps the business documents, which are formatted in a logical structure for use within the Integration Server, to the structure of the Oracle Applications open interface tables.</p> <p>The <code>bizDocMapping</code> service uses the <code>DocBizDoc</code> document (which is formatted in the logical layout of the business document) as its input. The <code>bizDocMapping</code> service also uses data transformation services to convert data to the structure of the Oracle Applications open interface tables.</p> <hr/>

Step	Description
3	<p>setDocTxn Service: The <code>setDocTxn</code> service inserts the mapped data into the Oracle Applications open interface tables.</p> <p>The <code>setDocTxnInputRec</code> document contains the flat table structure that reflects the Oracle Applications open interface table structure. The <code>setDocTxnInputRec</code> document is the output from the <code>bizDocMapping</code> service and the input to <code>setDocTxn</code>.</p>
4	<p>importDoc Service: If the service is set to automatically execute concurrent processing (see step 1 above), the <code>importDoc</code> service calls the <code>execDocConcProg</code> service to insert the data from the Oracle Applications open interface table into the Oracle Applications production tables. Additionally, the <code>importDoc</code> service calls the <code>getDocImport_ERR</code> service to report any data that could not be imported into an Oracle Applications production table because it contained invalid business data.</p>

Using Receive Services

Before you can use a receive service you must first:

- 1 Install the package containing the service you want to use. See the *webMethods Oracle Applications Adapter 10.7SC Predefined Transaction Services Installation Guide* for instructions.
- 2 Make sure you have installed the database scripts for the service, if applicable. See [“Database Scripts” on page 26](#) for more information.
- 3 Before you can use any predefined transaction services you must reconfigure the default Oracle Applications Adapter connection parameters to point to your Oracle Applications system. All predefined transaction services for Oracle Applications 10.7SC use a connection named `OraApps107SC`. For instructions about editing connections, see the chapter on adapter connections in the *webMethods Oracle Applications Adapter User’s Guide*.
- 4 Each receive service contains settings that define how the Oracle Applications concurrent processing will run for the service. The parameters for each service are based on the Oracle Applications demo database. If you do not want to use a service’s default settings for concurrent processing, you can modify the settings in the service’s `specifyConcProgParams` MAP step.

To locate receive services in the Developer, see [“Packaging Structure” on page 22](#).

Optimizing Receive Services

You might want to modify a receive service to optimize the service's performance. This section lists some possible scenarios where you might want to update the service.

- If you do not need a receive service to perform transformations, and you do not need the service to organize the input document into a hierarchical record structure, do the following:
 - a Remove the `bizDocMap` service from the receive service.
 - b Update the input signature of the receive service to use the `setDocTxnInputRec`.
 - c Map the input to the input of `setDocTxn` service within the receive service.
- If you want to organize the input document in a hierarchical way, or if you want to use some of the transformers defined in the service, do the following:
 - a Create a custom input document record definition.
 - b Update the input signature of the receive service to match the custom record definition.
 - c Create a new `bizDocMap` service to transform the hierarchical document, as described in the custom input record definition, into flat table entries. This is a complex process requiring a lot of customization of the service; if you do this you must have a strong understanding of the Oracle Applications data schemas involved with the service.
- If the base table columns are different (due to minor difference between versions of Oracle Applications):
 - a Modify the `setDocTxn` transaction definition as necessary.
 - b Refresh the `setDocTxn` service.
 - c Update the structure of the `setDocTxnInputRec` record definition to reflect changes to the `setDocTxn` service.
 - d Update the structure of the `DocBizDoc` record definition to reflect changes to the `setDocTxn` service.
 - e Modify the `bizDocMapping` service to reflect the changes to the data columns.



Important! The predefined transaction services were tested to guarantee they function properly as delivered, with the specified Oracle Applications release. If you modify a predefined transaction service in a way other than as specified in this guide (for example, you need to configure the flow as described), we cannot support the service. You will need to thoroughly test any service that you have modified extensively as we cannot guarantee that it will work properly.

Query Services

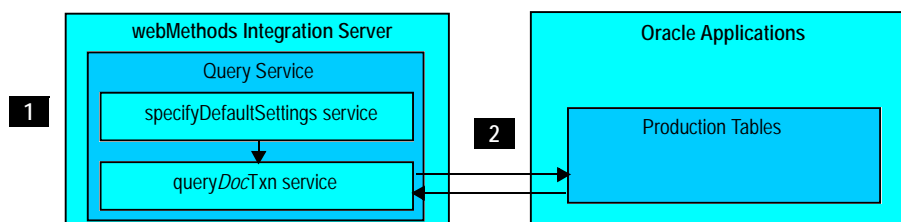
Query services retrieve information from Oracle Applications systems by querying the Oracle Applications production tables. Query services execute SQL trees against the corresponding Oracle Applications production tables to find and retrieve business documents that match the search criteria. For more information about SQL trees, see the *webMethods Oracle Applications Adapter User's Guide*.

Query services return all business documents that match the query criteria you specify in the query. If you do not specify a selection criteria for a particular parameter, the query ignores that parameter when performing the query. For example, if you specify a query to retrieve POs based on a particular PO creator, within a specific date range, the query service returns all purchase orders that meet that criteria. Any parameters that are not specified in the search criteria are ignored as part of the query.

There is no maximum number of business documents that the query service returns. To limit the business documents that the service might return, we recommend that you provide more stringent query criteria.

Overview of Query Service Transaction Processing

Query services retrieve data from Oracle Applications production tables by calling the following services, where *Doc* is the name of the business document:



Step	Description
1	<p>Specify Default Settings: By default, query services use an Oracle Applications Adapter connection named OraApps107SC. Ensure that the default Oracle Applications Adapter connection parameters point to your Oracle Applications system.</p> <hr/> <p>Note: In version 3.0 of the adapter, the \$dbAlias parameter in the specifyDefaultSettings MAP step could be modified to specify the connection the service will use. In version 6.0 of the adapter, that is no longer true. The \$dbAlias parameter is still available however it is no longer used to control a service's connection and cannot be modified.</p> <hr/>
2	<p>queryDocTxn Service: The queryDocTxn service queries the Oracle Applications production tables and returns business objects that match the search criteria in the structure of the queryDocTxnOutputRec document type.</p> <hr/>

Using Query Services

Before you can use a query service you must first:

- 1 Install the package containing the service you want to use. See the *webMethods Oracle Applications Adapter 10.7SC Predefined Transaction Services Installation Guide* for instructions.
- 2 Make sure you have installed the database scripts for the service.

If a query service has a corresponding send service, the query service uses the same database scripts as its corresponding send service. If you will not be using the query service's corresponding send service, you should disable all triggers associated with the send service. If you install triggers associated with a send service but do not use that service, the triggers will populate the WM_TRACKCHANGES table with records (according to the logic of the send service), and the records will never be removed from the table, causing the WM_TRACKCHANGES table to grow, consuming resources in your Oracle Applications system. You should disable triggers that are not being used by send services to avoid this problem.

See [“Database Scripts” on page 26](#) for more information.

- 3 Before you can use any predefined transaction services you must reconfigure the default Oracle Applications Adapter connection parameters to point to your Oracle Applications system. All predefined transaction services for Oracle Applications 10.7SC use a connection named OraApps107SC. For instructions about editing connections, see the chapter on adapter connections in the *webMethods Oracle Applications Adapter User's Guide*.

To locate query services in the Developer, see [“Packaging Structure” on page 22](#).

Send Services

Send services notify the webMethods Integration Platform of changes in Oracle Applications systems when specific events occur in the Oracle Applications system, such as when data is inserted, updated, or deleted. Send services then deliver the data in the form of business documents.

Send services use a two-part process:

- On the Oracle Applications side, custom triggers capture events that occur in the Oracle Applications production tables, and record those events in a custom table called WM_TRACKCHANGES. The custom table maintains the events as records until a send service polls the tables to retrieve the data. Each send service uses one or more custom triggers.

See [“Custom Tables Used with Send Services” on page 43](#) for more information about the custom tables and how they are populated.

- On the Integration Server side, send services poll the custom tables and the Oracle Applications production tables to retrieve business documents related to the events recorded in the custom tables, and then the send services send the documents to the designated recipients of the service.

Send services handle transactions differently depending on whether the event on the business document is an INSERT, UPDATE, or DELETE event, as follows:

- **INSERT and UPDATE events:** Whenever a business object is inserted or updated in Oracle Applications, the service creates a snapshot of the business document in the Oracle Applications production tables at the instance the tables are polled, and sends it to the Integration Server.
- **DELETE events:** Send services process delete events differently depending on how the business object is deleted in Oracle Applications. Business objects may be deleted from Oracle Applications in one of the following two ways:
 - Actions performed in the Oracle Applications system that result in a status update in the transaction header base table that indicates a “logical” delete of a business object are treated as if they had a status of UPDATE.
 - Actions performed in the Oracle Applications system that result in an actual DELETE of the record from the transaction header base table will result in the object being deleted. Business documents corresponding to this type of action will not be complete documents; they will only contain basic header-level information required to properly identify the business document to delete the business object.

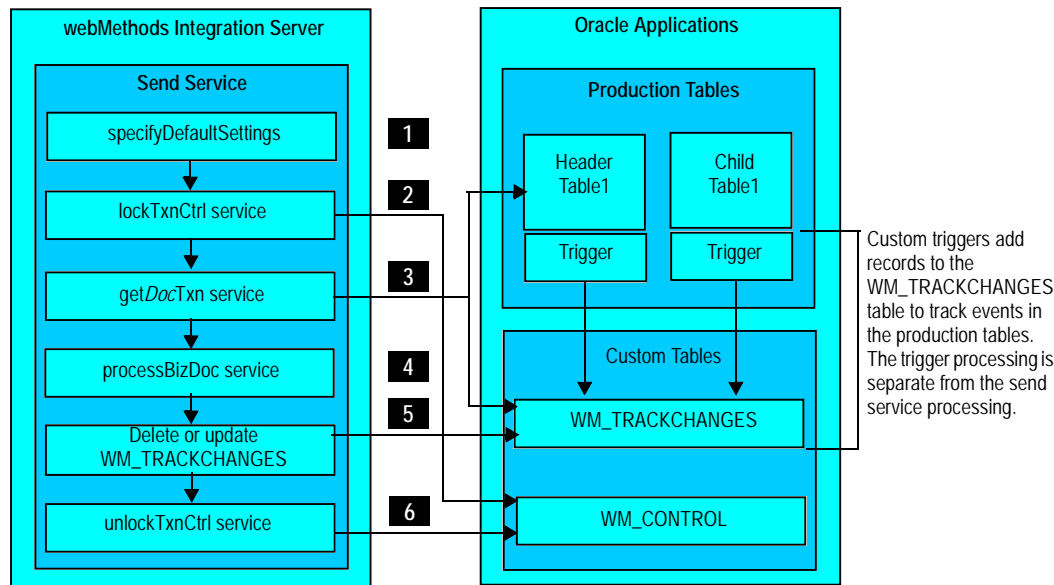
The following sections describe how the custom tables work, and how the send services interact with the custom tables to retrieve data and send it to the webMethods Integration Platform.

Overview of Send Service Transaction Processing

Before you can use a send service you must make sure the Oracle Applications Adapter’s custom tables are installed, and you must install and enable the custom triggers for each send service you want to implement. The custom tables are installed with the core database scripts. See [“Core Database Scripts” on page 27](#) for more information. For information about custom triggers, see [“Database Scripts” on page 26](#). For information about custom tables, see [“Custom Tables Used with Send Services” on page 43](#).

Custom triggers for the send service register changes to the Oracle Applications business objects in the WM_TRACKCHANGES custom table. For more information about how the triggers record these changes, see [“WM_TRACKCHANGES Table” on page 43](#).

Send services process transactions as follows:



Step	Description
1	<p>Specify Default Settings: Send services use a set of high-level settings to define how the services will execute. The default settings should be adequate for most cases. However, you can modify the default settings if necessary, as follows:</p> <ul style="list-style-type: none"> ■ In the specifyDefaultSettings MAP step, the noOfRowsToFetch parameter defines the maximum number of business documents the service will retrieve per polling interval. By default this value is set at 100000. <p>Note: In version 3.0 of the adapter, the \$dbAlias parameter in the specifyDefaultSettings MAP step could be modified to specify the connection the service will use. In version 6.0 of the adapter, that is no longer true. The \$dbAlias parameter is still available however it is no longer used to control a service's connection and cannot be modified.</p>
2	<p>lockTxnCtrl Service: The send service calls the lockTxnCtrl service, which queries the WM_CONTROL table for a specific TRANSACTION_TYPE to determine the polling status of the business document.</p> <ul style="list-style-type: none"> ■ If the lockTxnCtrl service cannot lock the WM_CONTROL table, then the send service exits and waits for next scheduled interval. <p>This action prevents send services from processing the same business documents for a specific TRANSACTION_TYPE multiple times if the service's polling interval is shorter than the time it takes to process the business document.</p> <ul style="list-style-type: none"> ■ If the lockTxnCtrl service can lock the WM_CONTROL table, then the lockTxnCtrl service updates the polling status from READY to IN PROCESS. <p>This action ensures that no other service can retrieve the same business document for the same TRANSACTION_TYPE.</p> <p>Important! If you test a send service by stepping through its supporting services in Developer and you exit after you run the lockTxnCtrl service to lock the WM_CONTROL table but before you run the unlockTxnCtrl service to unlock it, the send service will no longer be able to run successfully because the WM_CONTROL table will continue to be locked. To unlock the table manually, you can run the WmOACommon107SC.commonOA107SC.utils:unlockTxnCtrl service, providing the appropriate TRANSACTION_TYPE parameter.</p> <p>For more information, see “WM_CONTROL Table” on page 47.</p>

Step	Description
3	<p>getDocTxn Service: The send service calls the getDocTxn service. The getDocTxn service uses the getDocTxnOutputRec document to define the business document's logical structure. The getDocTxn service also queries the WM_TRACKCHANGES table and the Oracle Applications production tables to create one or more business documents, as follows:</p> <ul style="list-style-type: none">■ In the WM_TRACKCHANGES table, the getDocTxn service finds the records representing the latest unprocessed events made to the business object. Unprocessed event records have a value of N in the PROCESSED_FLAG field. <p>Note: For some services, the business document must have a status of 'Approved' in Oracle Applications for the service to retrieve it, preventing the service from picking up business documents that are not completed and approved, such as an incomplete Purchase Order.</p> <ul style="list-style-type: none">■ For each unprocessed business document in the WM_TRACKCHANGES table, the getDocTxn service selects records based on the business object type (TRANSACTION_TYPE), groups the records by the specified business object (TRANSACTION_ID), and calculates the total TRANSACTION_STATUS for all records of a specific business object. <p>Note: The TRANSACTION_STATUS is a numeric field. Each record contains a value that corresponds to its status: 0 for UPDATE, 1 for INSERT, and 2 for DELETE. For more information about the WM_TRACKCHANGES table, see "WM_TRACKCHANGES Table" on page 43.</p> <ul style="list-style-type: none">■ Based on the sum of the TRANSACTION_STATUS fields for all of the records in the WM_TRACKCHANGES table for the business object, the service processes the record as follows:<ul style="list-style-type: none">■ 0: The service processes the record as an UPDATE event.■ 1: The service processes the record as an INSERT event.■ 2: The service processes the record as a DELETE event.■ 3 or higher: The service takes no action. <p>For a description of how the service processes INSERT and UPDATE events, see page 37.</p> <p>For a description of how the service processes DELETE events, see page 37.</p> <p>The service uses the WEB_TRANSACTION_ID to process the oldest records first.</p>

Step	Description
3 Cont.	<p>The service generates one or more business documents based on the results of the <i>getDocTxn</i> service.</p> <p>Note: Polling services query the WM_TRACKCHANGES table and Oracle Applications production tables by executing SQL trees against them. For more information about SQL trees, see the <i>webMethods Oracle Applications Adapter User's Guide</i>.</p>
4	<p>processBizDoc Service: The send service calls the processBizDoc service to process the business document. By default, this service does not send the business document anywhere. You must modify this service to send the business document to the desired recipients.</p> <p>After the processBizDoc service processes the business document, it sets the transferStatus field of the docTransferResults document to SUCCESS or ERROR. By default, the transferStatus field for each business document is set to ERROR.</p>
5	<p>WM_TRACKCHANGES Table: The send service manages the records in the WM_TRACKCHANGES table as follows, depending on whether the service is configured to run in debug mode:</p> <ul style="list-style-type: none"> ■ If the service is not running in debug mode (debugMode parameter = FALSE), then it deletes one or more records from WM_TRACKCHANGES table after it processes them. ■ If the service is running in debug mode (debugMode parameter = TRUE), then it keeps the records in the WM_TRACKCHANGES table and updates them to indicate that they have been processed. <p>If a business document fails during transmission, the send service inserts a new record into the table for the failed record. The new record will have a TRANSACTION_STATUS that contains the sum of the TRANSACTION_STATUS fields for all of the related business object records for the business document that failed.</p> <p>You do not need to manually re-process a transaction that fails in the processBizDoc service because it will be processed again during the next scheduled execution of the service.</p>
6	<p>unlockTxnCtrl Service: The send service calls the unlockTxnCtrl service to unlock the WM_CONTROL table after the send service finishes processing.</p>

Using Send Services

To use a send service, do the following:

- 1 Install the package containing the service you want to use. See the *webMethods Oracle Applications Adapter 10.7SC Predefined Transaction Services Installation Guide* for instructions.
- 2 Ensure that all database scripts for the send service are installed. Also, all triggers associated with the send service must be enabled. By default they are enabled. For more information about database scripts, see [“Database Scripts” on page 26](#).
- 3 Before you can use any predefined transaction services you must reconfigure the default Oracle Applications Adapter connection parameters to point to your Oracle Applications system. By default, all predefined transaction services for Oracle Applications 10.7SC use a connection named OraApps107SC. For instructions about editing connections, see the chapter on adapter connections in the *webMethods Oracle Applications Adapter User’s Guide*.
- 4 Specify the maximum number of business documents the service will retrieve per polling interval. By default this value is set at 100000. To modify this setting, update the value of the `noOfRowsToFetch` parameter in the send service’s `specifyDefaultSettings` MAP step.
- 5 You can run any send service in debug mode by setting the `debugMode` parameter to `TRUE`. By default the send services run in debug mode. You should run your services with this option turned off because if you run the services in debug mode, the `WM_TRACKCHANGES` table will grow, consuming resources in your Oracle Applications system. For more information about running send services in debug mode, see [“Using Send Services in Debug Mode” on page 43](#).
- 6 Customize the `procBizDoc` service to define how the send service processes and delivers the business document. Within this service you must be sure to provide a value of either `SUCCESS` or `ERROR` in the `transferStatus` output field of the `docTransferResults` document.
- 7 Schedule the service to run. To schedule the service you use the Integration Server’s scheduler. For information about using the Integration Server’s scheduler, see the chapter on managing services in the *webMethods Administrator’s Guide*.

To locate send services in the Developer, see [“Packaging Structure” on page 22](#).

Using Send Services in Debug Mode

Debug mode enables you to keep track of records that have been processed by send services so that you can review how the records were processed by the services. You configure each send service to run in debug mode independently of other send services.

When a service is running in debug mode, the WM_TRACKCHANGES table retains records after they are processed, marks their PROCESSED_FLAG field to Y, and adds a value to the DATE_PROCESSED field. This enables you to review records that have been processed by the service rather than having the service simply delete the processed records.

You configure a send service to use debug mode by setting the debugMode parameter in the specifyDefaultSettings MAP step to TRUE. By default, debugMode is set to FALSE.

Custom Tables Used with Send Services

All send services use two custom tables to process events in the Oracle Applications system that require data to be sent to the webMethods Integration Platform: WM_TRACKCHANGES and WM_CONTROL.

The following sections describe these custom tables, how they are populated, and how they are used by send services to capture data and send it to the webMethods Integration Platform.

WM_TRACKCHANGES Table

The WM_TRACKCHANGES table records INSERT, UPDATE, and DELETE events in the Oracle Applications production tables for business objects that you want to track. The recording of these events is done by triggers. Each send service depends on one or more triggers to track the events that you want to capture.

Each trigger inserts a new record in the WM_TRACKCHANGES table whenever an event occurs in an Oracle Applications production table that causes the trigger to fire.



Note: You must install and enable triggers on your Oracle Applications system for every send service you want to use. See [“Database Scripts” on page 26](#) for more information about installing and enabling triggers.

Send services poll the WM_TRACKCHANGES table to determine whether they need to send business documents to the webMethods Integration Platform.

This section lists the structure of the table, and describes how the table is populated.

Field	Description
TRANSACTION_TYPE	Type of business object being inserted, updated, or deleted. For example, VENDOR, PO, or INVOICE.
TRANSACTION_ID	Unique Oracle Applications identifier to identify the business object associated with the record. For example, for a Vendor business object this field is PO_VENDORS.PO_VENDOR_ID. For each record in the table associated with this business object, the value for this field will be the specific Vendor ID associated with the Vendor business document.
DATE_CREATED	Date the record was added to the table.
WEB_TRANSACTION_ID	<p>Unique identifier for records in the table. This value is a unique sequence number within this table that determines the order in which the records were added to the table.</p> <p>This information is used to make sure the records are processed in sequence according to the sequence of events in Oracle Applications.</p>
TRANSACTION_STATUS	<p>Type of event occurring on the business object. Valid values include:</p> <ul style="list-style-type: none"> ■ 0 = UPDATE ■ 1 = INSERT ■ 2 = DELETE <p>When you add a record to a header table, it will have a status of 1 (for INSERT). When you add a record to any other table related to the new record, it will have a status of 0 (for UPDATE).</p> <p>The send services determine what action to take on the business documents according to the sum of the TRANSACTION_STATUS field for all related records. That is, the TRANSACTION_STATUS field for all related records are added together to determine what action the send service will take when it polls the WM_TRACKCHANGES table. For information about how the send services process the business documents based on the TRANSACTION_STATUS, see page 40.</p>

Field	Description
COMMENT	Comment describing the record. This value is provided by the trigger that adds the record to the table, or by the service that creates a new record if the business document transfer fails.
PROCESSED_FLAG	<p>Indicates whether the record has been processed (Y), or not processed (N). The PROCESSED_FLAG defaults to N.</p> <p>If you are using a send service in debug mode, the service sets the value of this field to Y and retains the record in the table. If you are not running the service in debug mode, the service deletes the record after it is processed.</p> <p>See “Using Send Services in Debug Mode” on page 43 for more information about debugging send services.</p>
DATE_PROCESSED	<p>Date the polling service retrieved the record from the table.</p> <p>If you are using a send service in debug mode, the service sets the value of this field to the date the record was processed in the database’s date format; for example, 30-Jun-04. If you are not running the service in debug mode, the service deletes the record after it is processed.</p> <p>See “Using Send Services in Debug Mode” on page 43 for more information about debugging send services.</p>

Example of How the WM_TRACKCHANGES Table is Populated

Suppose that a new Vendor with four Sites and five Contacts is added to Oracle Applications. That event causes the WM_TRACKCHANGES table to receive ten records: one for the Vendor, four for the Sites, and Five for the contacts. In this case, the parent record is Vendor and the child records are Site and Contact.

The following table shows how triggers insert the records into the WM_TRACKCHANGES table for this example. For all of the records in the example, the TRANSACTION_TYPE is VENDOR, the TRANSACTION_ID is 101, and the PROCESSED_FLAG is set to N. The other fields are as follows:

DATE_CHANGED	WEB_TRANSACTION_ID	TRANSACTION_STATUS	COMMENT
30-Jun-04	5001	1	Vendor inserted
30-Jun-04	5002	0	Site inserted
30-Jun-04	5003	0	Site inserted
30-Jun-04	5004	0	Contact inserted
30-Jun-04	5005	0	Contact inserted
30-Jun-04	5006	0	Contact inserted
30-Jun-04	5007	0	Site inserted
30-Jun-04	5008	0	Site inserted
30-Jun-04	5009	0	Contact inserted
30-Jun-04	5010	0	Contact inserted

Notice that in this example, the values for the Site and Contact records are assigned a status of 0. This is because these records are child records of the Vendor record. The parent record is treated as an INSERT event. The child records are treated as UPDATE events.

The table records the records like this so that the send service accurately creates a single business document for the parent business document and does not create separate business documents for the child records. The parent record will include the new child data in its business document.

WM_CONTROL Table

The WM_CONTROL table prevents a service from processing the same business document multiple times. This situation could occur if the polling interval for a particular transaction is shorter than the time it takes to execute the service, or if you are using the service in a clustered Integration Server environment.

If a service attempts to process a record and the WM_CONTROL table is locked, the service defers its execution until the next polling interval.

The fields in the table are as follows:

Field	Description
TRANSACTION	Type of business object. For example, VENDOR, PO, or INVOICE.
STATUS	Indicates the status of the service's processing. Valid values are READY and IN PROCESS.

Financial Predefined Transaction Services

■ Overview	51
■ Query Accounts Payable Invoice Service	53
■ Query Accounts Payable Payment Service	54
■ Query Accounts Receivable Transactions Service	55
■ Query Customer Service	57
■ Query General Ledger Balance Service	59
■ Query Journal Service	62
■ Query Vendor Service	64
■ Receive Accounts Payable Invoice Service	65
■ Receive Auto Invoice Service	71
■ Receive Bank Statement Service	100
■ Receive Customer Service	104
■ Receive FA Budget Service	115
■ Receive General Ledger Budget Service	118
■ Receive Journal Service	121
■ Receive Mass Additions Service	126
■ Send Accounts Payable Invoice Service	131
■ Send Accounts Payable Payment Service	139
■ Send Accounts Receivable Transactions Service	148
■ Send Customer Service	179

- [Send Journal Service](#)199
- [Send Vendor Service](#)204

Overview

This chapter describes the predefined transaction services provided in the Oracle Applications Adapter’s 10.7SC Financial package. For more information about using the predefined transaction services, see [Chapter 1, “Predefined Transaction Services”](#) on [page 17](#).

The table below shows the predefined transaction services organized by Oracle Applications module. This chapter lists the transactions in alphabetical order.

Oracle Applications Module	Predefined Transactions
Cash Management	■ “Receive Bank Statement Service” on page 100
Fixed Assets	■ “Receive FA Budget Service” on page 115 ■ “Receive Mass Additions Service” on page 126
General Ledger	■ “Query General Ledger Balance Service” on page 59 ■ “Query Journal Service” on page 62 ■ “Receive General Ledger Budget Service” on page 118 ■ “Receive Journal Service” on page 121 ■ “Send Journal Service” on page 199

Oracle Applications Module	Predefined Transactions
Payables	<ul style="list-style-type: none">■ “Query Accounts Payable Invoice Service” on page 53■ “Query Accounts Payable Payment Service” on page 54■ “Query Vendor Service” on page 64■ “Receive Accounts Payable Invoice Service” on page 65■ “Send Accounts Payable Invoice Service” on page 131■ “Send Accounts Payable Payment Service” on page 139■ “Send Vendor Service” on page 204
Receivables	<ul style="list-style-type: none">■ “Query Accounts Receivable Transactions Service” on page 55■ “Query Customer Service” on page 57■ “Receive Auto Invoice Service” on page 71■ “Receive Customer Service” on page 104■ “Send Accounts Receivable Transactions Service” on page 148■ “Send Customer Service” on page 179

Query Accounts Payable Invoice Service

The name of this service is:

WmOAFIN107SC.payables107SC.queryOA.APInvoice:queryAPInvoice

This service queries for new or changed invoice records. You set up Accounts Payable Invoices in Oracle Applications Payables to record invoice-related information.

You can use the following parameters to query accounts payable data:

- VENDOR_NAME
- INVOICE_NUM
- TERMS_NAME
- VENDOR_SITE_ADDRESS_LINE1
- VENDOR_SITE_ADDRESS_LINE2
- VENDOR_SITE_ADDRESS_LINE3
- VENDOR_TOWN_OR_CITY
- VENDOR_COUNTY
- VENDOR_POSTAL_CODE
- VENDOR_STATE
- STATUS
- ORGANIZATION_NAME

Database Scripts

This service uses the same database scripts as the Send Accounts Payable Invoice.



Note: If you use this service but you do *not* use the Send Accounts Payable Invoice service, you should run the `wm_disable_from_apinvoice.sql` script to disable the triggers installed by the Send Accounts Payable Invoice service.

For a detailed description of these database scripts, see [“Send Accounts Payable Invoice Service” on page 131](#). For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service uses the following transaction definition:

- `queryAPInvoiceTxn107SC.txp`

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow executes as follows:

- `queryAPInvoiceTxn` queries the Oracle Applications database for any APInvoice information matching the parameter values. The parameters are defined as the input to this service.

For more details on query transactions, see [“Using Query Services” on page 36](#).

Business Document Structure

This service uses the same business document structure as the Send Accounts Payable Invoice service. For a detailed description of the business document structure, see [“Send Accounts Payable Invoice Service” on page 131](#).

Query Accounts Payable Payment Service

The name of this service is:

`WmOAFIN107SC.payables107SC.queryOA.APPayment:queryAPPayment`

This service queries Accounts Payable payments. You set up Accounts Payable Payments in Oracle Applications Payables to record payment related information.

You can use the following parameters for querying Accounts Payable Payment data:

- `ORGANIZATION_NAME`
- `CURRENT_BANK_ACCOUNT_NAME`
- `PAYMENT_TYPE`
- `VENDOR_NAME`
- `INVOICE_NUM`

Database Scripts

This service uses the same database scripts as the Send Accounts Payable Payments service.



Note: If you use this service but you do *not* use the Send Accounts Payable Payments service, you should run the `wm_disable_from_appayment.sql` script to disable the triggers installed by the Send Accounts Payable Payments service.

For a detailed description of these database scripts, see [“Send Accounts Payable Payment Service” on page 139](#). For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service uses the following transaction definition:

- `queryAPPaymentTxn107SC.txp`

Flow Control

The main flow executes as follows:

- `queryAPPaymentTxn` queries for any Accounts Payable Payment information matching the parameter values. The parameters are the input to this service.

For more details on query transactions, see [“Using Query Services” on page 36](#).

Business Document Structure

This service uses the same business document structure as the Send Accounts Payable Payment Service. For a detailed description of the business document structure, see [“Send Accounts Payable Payment Service” on page 139](#).

Query Accounts Receivable Transactions Service

The name of this service is:

`WmOAFIN107SC.receiveables107SC.queryOA.ARTransaction:queryARTransaction`

This service queries the Accounts Receivable Transactions.

You can use the following parameters for querying Accounts Receivable Transaction data:

- `TRX_NUMBER`
- `TRX_TYPE_NAME`

- SHIP_TO_CUSTOMER_NAME
- SHIP_TO_CUSTOMER_NUMBER
- BILL_TO_CUSTOMER_NAME
- BILL_TO_CUSTOMER_NUMBER
- ORGANIZATION_NAME

Database Scripts

This service uses the same database scripts as the Send Accounts Receivable Transaction service.



Note: If you use this service but you do *not* use the Send Accounts Receivable Transaction service, you should run the `wm_disable_from_artrans.sql` script to disable the triggers installed by the Send Accounts Receivable Transaction service.

For a detailed description of these database scripts, see [“Send Accounts Receivable Transactions Service” on page 148](#). For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definition

This service uses the following transaction definition:

- `queryARTransactionTxn107SC.txp`

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow executes as follows:

- `specifyDefaultSettings` specifies the default parameter settings required for service execution. You should change these settings accordingly.
- `queryARTransactionTxn` queries the Oracle Applications database for any AR Transactions information matching the parameter values. The parameters are defined as the input to this service.

For more details on query transactions, see [“Using Query Services” on page 36](#).

Business Document Structure

This service uses the same business document structure as the Send Accounts Receivable Transactions service. For a detailed description of the business document structure, see [“Send Accounts Receivable Transactions Service” on page 148](#).

Query Customer Service ---

The name of this service is:

WmOAFIN107SC.receivables107SC.queryOA.customer:queryCustomer

This service provides the list of new or changed customers.

You set up Customers in Oracle Applications Receivables to record customer-related information related to persons or organizations.

You can use the following parameters to query Customer data:

- CUSTOMER_NUMBER: Unique identifier for a customer.
- CUSTOMER_NAME
- ADDRESS1
- ADDRESS2
- ADDRESS3
- ADDRESS4
- CITY
- COUNTY
- STATE
- COUNTRY
- POSTAL_CODE
- PROVINCE

Database Scripts

This service uses the same database scripts as the Send Customer Service service.:



Note: If you use this service but you do *not* use the Send Customer Service service, you should run the `wm_disable_from_customer.sql` script to disable the triggers installed by the Send Customer Service service.

For a detailed description of these database scripts, see [“Send Customer Service” on page 179](#). For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service uses the following transaction definition:

- `queryCustomerTxn107SC.txp`

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow executes as follows:

- `specifyDefaultSettings` specifies the default parameter settings required for service execution. You should change these settings accordingly.
- `queryCustomerTxn` queries the Oracle Applications database for any Customer information matching the parameter values. The parameters are defined as the input to this service.

For more details on query transactions, see [“Using Query Services” on page 36](#).

Business Document Structure

This service uses the same business document structure as the Send Customer service. For a detailed description of the business document structure, see [“Send Customer Service” on page 179](#).

Query General Ledger Balance Service

The name of the service is:

WmOAFIN107SC.generalLedger107SC.queryOA.GLBalance:queryGLBalance

This service retrieves General Ledger (GL) Budget, Encumbrance, Actual, and Funds Available account balances based on the following parameters:

- PERIOD_NAME: Period for which the data would be retrieved from Oracle Applications.
- PERIOD_TYPE: Period type.
- CURRENCY_CODE: Currency for the required General Ledger balance.
- SET_OF_BOOKS_NAME: Set of books name.

While configuring the queryGLBalanceTxn service, a java.outOfMemory error can occur. Because the database can have so many records, it may be unable to configure the service. One workaround is to use a restricting condition in the query, that is, 1 = 2 so that the service configures successfully. After you have configured this service, remove the restricting condition from your transaction definitions. In addition, update the SQLOut parameters of the service's transactionRecord in the webMethods Developer, and remove the restricting condition. (In this example, you remove the 1 = 2 condition.)

Database Scripts

This service uses the following database scripts:

Database Script	Description
wm_install_from_glbalance.sql	Runs all the scripts listed below, except the uninstall script.
wm_from_glbalance_vw.sql	Creates the following required view component: <ul style="list-style-type: none"> ■ WM_GL_BALANCES_QRY_VW
wm_drop_from_glbalance.sql	Uninstalls all components created by wm_install_from_glbalance.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service uses the following transaction definition:

- queryGLBalanceTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow executes as follows:

- specifyDefaultSettings specifies the default parameter settings required for service execution. You should change these settings accordingly.
- queryGLBalanceTxn queries the Oracle Applications database for any GL Balance matching the parameter values. The parameters are defined as the input to this service.

For more details on query transactions, see [“Using Query Services” on page 36](#).

Business Document Structure

This service uses the following business document structure:

- GLBalance

GLBalance

Document Field	Oracle Applications Table/View Name	Column Name	Description
WEB_TRANSACTION_ID		Not used	Populated from a sequence and used internally in the Flow. Will contain NULL value for Queried Vendor data.
DOCUMENT_TYPE		Not used	Valid value is GLBALANCE.
DOCUMENT_STATUS		Not used	Valid value is QUERY
SET_OF_BOOKS_NAME	GL_SETS_OF_BOOKS	Name	Populated by joining the Set of Books Id with the corresponding Id in GL_SETS_OF_BOOKS table.

Document Field	Oracle Applications Table/View Name	Column Name	Description
ACCOUNT_CODE	GL_CODE_COMBINATIONS_KFV	CONCATENATED_SEGMENTS	Code Combination ID stored in GL_BALANCES table is used to join with the Code Combination Id in GL_CODE_COMBINATIONS_KFV table.
CURRENCY_CODE	GL_BALANCES	CURRENCY_CODE	Currency Code.
PERIOD_NAME	GL_BALANCES	PERIOD_NAME	Name of the period.
ACTUAL_FLAG	GL_BALANCES	ACTUAL_FLAG	Actual Flag.
BUDGET_VERSION_NAME	GL_BUDGET_VERSIONS	BUDGET_NAME	Joined with the Budget Version Id.
ENCUMBRANCE_TYPE	GL_ENCUMBRANCE_TYPES	ENCUMBRANCE_TYPE	ENCUMBRANCE_TYPE_ID is joined with GL_BALANCES. ENCUMBRANCE_TYPE_ID.
REVALUATION_STATUS	GL_BALANCES	REVALUATION_STATUS	Revaluation status.
PERIOD_TYPE	GL_BALANCES	PERIOD_TYPE	Type of period.
PERIOD_YEAR	GL_BALANCES	PERIOD_YEAR	Year of the period.
PERIOD_NUMBER	GL_BALANCES	PERIOD_NUMBER	Period number.
PERIOD_NET_DR	GL_BALANCES	PERIOD_NET_DR	Period Net Debit balance.
PERIOD_NET_CR	GL_BALANCES	PERIOD_NET_CR	Period Net Credit Balance.
PERIOD_TO_DATE_ADB	GL_BALANCES	PERIOD_TO_DATE_ADB	Period to date ADB balance.
QUARTER_TO_DATE_DR	GL_BALANCES	QUARTER_TO_DATE_DR	Quarter to date debit balance.
QUARTER_TO_DATE_CR	GL_BALANCES	QUARTER_TO_DATE_CR	Quarter to date credit balance.
QUARTER_TO_DATE_ADB	GL_BALANCES	QUARTER_TO_DATE	Quarter to date ADB balance.
YEAR_TO_DATE_ADB	GL_BALANCES	YEAR_TO_DATE_ADB	Year to date ADB balance.

Document Field	Oracle Applications Table/View Name	Column Name	Description
PROJECT_TO_DATE_DR	GL_BALANCES	PROJECT_TO_DATE_DR	Accumulated project debit balance.
PROJECT_TO_DATE_CR	GL_BALANCES	PROJECT_TO_DATE_CR	Accumulated project credit balance.
PROJECT_TO_DATE_ADB	GL_BALANCES	PROJECT_TO_DATE_ADB	Accumulated project ADB balance.
BEGIN_BALANCE_DR	GL_BALANCES	BEGIN_BALANCE_DR	Beginning debit balance.
BEGIN_BALANCE_CR	GL_BALANCES	BEGIN_BALANCE_CR	Beginning credit balance.
PERIOD_NET_DR_BEQ	GL_BALANCES	PERIOD_NET_DR_BEQ	Period net debitbalance, base currency.
PERIOD_NET_CR_BEQ	GL_BALANCES	PERIOD_NET_CR_BEQ	Period Net Credit Balance, base currency
BEGIN_BALANCE_DR_BEQ	GL_BALANCES	BEGIN_BALANCE_DR_BEQ	Beginning debitbalance, base currency.
BEGIN_BALANCE_CR_BEQ	GL_BALANCES	BEGIN_BALANCE_CR_BEQ	Beginning Credit Balance, base currency.
SUMMARY_TEMPLATE_NAME	GL_SUMMARY_TEMPLATES	TEMPLATE_NAME	Joins TEMPLATE_ID with GL_BALANCES. TEMPLATE_ID.

Query Journal Service

The name of this service is:

WmOAFIN107SC.generalLedger107SC.queryOA.journal:queryJournal

The Query Journal service queries Journal information.

This service enables you to use the following parameters:

- BATCH_NAME: Name of the batch
- SOURCE_NAME: Source Name
- CATEGORY_NAME: Category
- POSTED_DATE: Posted Date

While configuring the queryJournalTxn service, a java.outOfMemory error can occur. Because the database can have so many records, it would be unable to configure the service. One workaround is to use a restricting condition in the query, that is, 1 = 2, so that the service gets configured successfully. After you have configured this service, remove the restricting condition from your transaction definitions. In addition, update the SQLOut parameters of the service's transactionRecord in the webMethods Developer and remove the restricting condition. (In this example, you remove the 1 = 2 condition.)

Database Scripts

This service uses the same database scripts as the Send Journal service.



Note: If you use this service but you do *not* use the Send Journal service, you should run the wm_disable_from_journal.sql script to disable the triggers installed by the Send Journal service.

For a detailed description of these database scripts, see [“Send Journal Service” on page 199](#). For more information about using database scripts, see [“Database Scripts” on page 26](#).

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service uses the following transaction definition:

- queryJournalTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow executes as follows:

- queryJournalTxn queries the Oracle Applications database for any Journal information matching the parameter values. The parameters are defined as the input to this service.

For more details on query transactions, see [“Using Query Services” on page 36](#).

Business Document Structure

This service uses the same business document structure as the Send Journal service. For a detailed description of the business document structure, see [“Send Journal Service” on page 199](#).

Query Vendor Service

The name of this service is:

WmOAFIN107SC.payables107SC.queryOA.vendor:queryVendor

This service queries for new or changed Vendors.

You set up Vendors (also known as Suppliers) in Oracle Applications Payables and Oracle Applications Purchasing to record information about individuals and companies from which you purchase goods and services.

Use the following parameters to query Vendor data:

- **VENDOR_ NUMBER:** a unique vendor identifier in Oracle Applications and is assigned to the vendor on creation.
- **VENDOR_SITE_ CODE:** a unique site identifier for a vendor. The same site name can exist for multiple vendors.
- **INACTIVE_ DATE FROM:** the beginning date the vendor is inactive.
- **INACTIVE_ DATE_ TO:** the ending date the vendor is inactive.
- **TAXPAYER_ ID** of the Vendor.
- **TAX_ REGISTRATION_ NUMBER.**

Database Scripts

This service uses the same database scripts as the Send Vendor service.



Note: If you use this service but you do *not* use the Send Vendor service, you should run the `wm_disable_from_vendor.sql` script to disable the triggers installed by the Send Vendor service.

For a detailed description of these database scripts, see [“Send Vendor Service” on page 204](#). For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service uses the following transaction definitions:

- `queryVendorTxn107SC.txp`

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow executes as follows:

- `specifyDefaultSettings` specifies the default parameter settings required for service execution. You should change these settings accordingly.
- `queryVendorTxn` queries the Oracle Applications database for any Vendor matching the parameter values. The parameters are defined as the input to this service.

For more details on query transactions, see [“Using Query Services” on page 36](#).

Business Document Structure

This service uses the same business document structure as the Send Vendor service. For a detailed description of the business document structure, see [“Send Vendor Service” on page 204](#).

Receive Accounts Payable Invoice Service

The name of this service is:

`WmOAFIN107SC.payables107SC.intoOA.APInvoice:receiveAPInvoice`

The Payables Open Invoice Import interface tables process and validate receipt data that comes from sources other than the Invoice Gateway window in Payables.

The invoice type can be either Credit or Standard depending on the invoice amount. You can create multiple distributions for an invoice line by populating the line interface table. You can also import an invoice matched to a purchase order.

The Payables Open Invoice Import interface tables do not have a group ID or batch ID column to track running the interface. There is also no request ID column to track the records that are processed by a particular request submission. The source name tracks errors for the instance of a flow, so the errors that display cannot be restricted to the particular flow instance. This is evident when you run the request from the Oracle Applications SRS screen that also displays the previous errors as output.

Database Scripts

This service uses the following database scripts:

Database Script	Description
wm_install_into_apinvoice.sql	Runs all the scripts listed below, except the uninstall script.
wm_into_apinvoice_pkg.sql	Installs WM_AP_INV_IMP_HANDLER_PKG, WM_HANDLE_OPENAP, which calls the WM_CONC_REQUEST.WM_REQUEST_SUBMIT procedure to submit the Accounts Payable import process.
wm_into_apinvoice_seq.sql	Creates the following components: WM_AP_GROUP_S, which creates the GROUP_ID.
wm_drop_into_apinvoice.sql	Uninstalls all components created by wm_install_into_apinvoice.sql.

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definitions:

- setAPInvoiceTxn107SC.txp
- APInvoiceTxn107SC.txp

Flow Control

The main flow executes as follows:

- **specifyDefaultSettings** specifies the default parameter settings required for service execution. You should change these settings accordingly.
- **specifyConcProgParams** specifies the default parameter settings required for concurrent program execution. You should change these settings accordingly.
- **bizDocMapping** maps the incoming business document structure to the required Oracle Applications data structures (interface tables).
 - **pickSequence** generates the report header ID that requires inserting a line in the AP_EXPENSE_REPORT_LINES_ALL table. It uses sequence AP_EXPENSE_REPORT_HEADERS_S to generate unique ID.
 - **getVendorId** is a transformer that maps the business doc Idata structure to the interface table Idata structure. It takes the vendor name as the input parameter to get the vendor id.

- **getVendorSiteId** is a transformer that maps the business doc Idata structure to the interface table Idata structure. It takes address details as the input parameter to get the vendor site code.
- **getSOBId** is a transformer that maps the business doc Idata structure to interface table Idata structure. It takes SET_OF_BOOK_NAME as input parameter to get the set of book id.
- **getAWTGroupId** is a transformer that maps the business the Idata structure to the interface table Idata structure. It takes AWT_GROUP_NAME name from the business document and gets the GROUP_ID from the Oracle Applications database.
- **getHROrgId** is a transformer that gets the Organization name as an input parameter to get the org_id from HR_ALL_ORGANIZATION_UNITS.
- **getProjectAndTaskId** is a transformer that maps the business doc Idata structure to the interface table Idata structure. It takes Project Name and Task Name as input and returns Project ID and Task Id for the same.
- **setAPIInvoiceTxn** inserts data into the interface table AP_EXPENSE_REPORT_HEADERS_ALL and AP_EXPENSE_REPORT_LINES_ALL_INTERFACE from the Idata structure resulted in the bizDocMapping service.
- **importAPIInvoice** imports data to the production table from the interface table. To monitor the import process this in turn sequentially invokes following services execAPIInvoiceConcProg, checkAPIInvoiceImportStatus, and getAPIInvoiceImport_ERR.

If the status of execution is FAILED, this step appends the dbErrorMsg and concProgMsg record list. Otherwise it checks for data error if occurred during the import.

- **execAPIInvoiceConcProg** inserts data into the production table. It picks up data from the interface table AP_EXPENSE_REPORT_HEADERS_ALL and AP_EXPENSE_REPORT_LINES_ALL_INTERFACE and insert data into the AP_INVOICES_ALL, AP_DISTRIBUTIONS_ALL production tables.
- **checkAPIInvoiceImportStatus** queries the AP_EXPENSE_REPORT_HEADERS_ALL table to find the number of rows for the passed SOURCE.
- **getAPIInvoiceImport_ERR** gets the error messages that occurs during the data import to the production table from interface table.

For more details on receive transactions, see [“Overview of Receive Service Transaction Processing” on page 31](#).

Business Document Structure

This services uses the business document APInvoiceBizDoc. Its structure is as follows:

- 1.0 AP_INVOICE
 - 1.1 AP_INVOICE_LINES

1.0 AP_INVOICE (Maps to AP_INVOICES_INTERFACE)

Field Name	Maps to Column	Description
WEEK_END_DATE	WEEK_END_DATE	Required. Week ending date for the invoice.
TOTAL	TOTAL	Required. Total amount for the invoice.
VENDOR_NAME	VENDOR_ID	Required. Derived from PO_VENDORS to create the vendor name.
VENDOR_SITE_ADDRESS_LINE1	VENDOR_SITE_ID	Required. Address of the vendor site. Derives the Vendor Site ID in the interface table.
VENDOR_SITE_ADDRESS_LINE2	VENDOR_SITE_ID	
VENDOR_SITE_ADDRESS_LINE3	VENDOR_SITE_ID	
VENDOR_TOWN_OR_CITY	VENDOR_SITE_ID	
VENDOR_COUNTY	VENDOR_SITE_ID	
VENDOR_STATE	VENDOR_SITE_ID	
VENDOR_POSTAL_CODE	VENDOR_SITE_ID	
VENDOR_COUNTRY	VENDOR_SITE_ID	
REFERENCE_1	REFERENCE_1	Projects related reference.
REFERENCE_2	REFERENCE_2	Projects related reference.
LIABILITY_ACCOUNT	ACCTS_PAY_CODE_COMBINATION_ID	Required. Accounts Payable Liability GL Code Combination Id derived from the Accounts Code Combination.
SET_OF_BOOKS_NAME	SET_OF_BOOKS_ID	Required. Derived from the name of GL_SETS_OF_BOOKS.
SOURCE	SOURCE	Required. Predefined Invoice Source to indicate external system.
ACCOUNTING_DATE	ACCOUNTING_DATE	Required. Date to be used for Accounting.

Field Name	Maps to Column	Description
HOLD_LOOKUP_CODE	HOLD_LOOKUP_CODE	Required. Specifies reason for holding the invoice.
DEFAULT_CURRENCY_CODE	DEFAULT_CURRENCY_CODE	Required. Default currency code for the Invoice.
DEFAULT_EXCHANGE_RATE_TYPE	DEFAULT_EXCHANGE_RATE_TYPE	Needed if the Invoice currency is different from the functional currency.
DEFAULT_EXCHANGE_RATE	DEFAULT_EXCHANGE_RATE	Needed if the Invoice currency is different from the functional currency.
DEFAULT_EXCHANGE_DATE	DEFAULT_EXCHANGE_DATE	Needed if the Invoice currency is different from the functional currency.
VOUCHER_NUM	VOUCHER_NUM	Voucher Number to be assigned to the imported invoice.
USSGL_TRANSACTION_CODE	USSGL_TRANSACTION_CODE	US General Ledger Transaction Code.
DOC_CATEGORY_CODE	DOC_CATEGORY_CODE	Document Category you want Invoice Import to assign.
WORKFLOW_APPROVED_FLAG	WORKFLOW_APPROVED_FLAG	Indicates whether the Workflow is approved.
ORGANIZATION_NAME	ORG_ID	Derived from HR_ALL_ORGANIZATION_UNITS.
INVOICE_NUM	INVOICE_NUM	Invoice Number. Not required if automatic numbering is used.
DESCRIPTION	DESCRIPTON	Description of the Invoice.
AWT_GROUP_NAME	AWT_GROUP_ID	Withholding tax name. Derived from AP_AWT_GROUPS.

1.1 AP_INVOICE_LINES (Maps to AP_INVOICES_LINES_INTERFACE)

Field Name	Maps to Column	Description
SET_OF_BOOKS_NAME	SET_OF_BOOKS_ID	Required. Derived from GL_SETS_OF_BOOKS.
CODE_COMBINATION	CODE_COMBINATION_ID	Derive from GL_CODE_COMBINATIONS_KFV for the Set of Book and concatenated segment.
ITEM_DESCRIPTION	ITEM_DESCRIPTION	Description for the Invoice Distribution.
AMOUNT	AMOUNT	Line Amount.

Field Name	Maps to Column	Description
CURRENCY_CODE	CURRENCY_CODE	Currency Code.
VAT_CODE	VAT_CODE	VAT Code.
LINE_TYPE_LOOKUP_CODE	LINE_TYPE_LOOKUP_CODE	Lookup code for the type of Invoice Distribution. Valid values are ITEM, TAX, MISC and FREIGHT.
REFERENCE_1	REFERENCE_1	Projects-related reference.
REFERENCE_2	REFERENCE_2	Projects-related reference.
STAT_AMOUNT	STAT_AMOUNT	Amount associated with distribution line for measuring statistical quantities.
USSGL_TRANSACTION_CODE	USSGL_TRANSACTION_CODE	USSGL transaction code for creating US Standard General Ledger Journal entries.
PROJECT_ACCOUNTING_CONTEXT	PROJECT_ACCOUNTING_CONTEXT	Project Accounting Context.
PROJECT_NAME	PROJECT_ID	Project Name validated against PA_PROJECTS_ALL.NAME.
TASK	TASK_ID	Project Task Name validated against PA_TASKS.
EXPENDITURE_TYPE	EXPENDITURE_TYPE	Project expenditure type.
EXPENDITURE_ITEM_DATE	EXPENDITURE_ITEM_DATE	Project expenditure item date.
EXPENDITURE_ORGANIZATION_NAME	EXPENDITURE_ORGANIZATION_ID	Project Organization Name validated against HR_ALL_ORGANIZATION_UNITS.NAME.
PA_QUANTITY	PA_QUANTITY	Quantity.
ADJUSTMENT_REASON	ADJUSTMENT_REASON	Adjustment reason.
JUSTIFICATION_REQUIRED_FLAG	JUSTIFICATION_REQUIRED_FLAG	Justification Required flag.
RECEIPT_REQUIRED_FLAG	RECEIPT_REQUIRED_FLAG	Receipt Required flag.
RECEIPT_VERIFIED_FLAG	RECEIPT_VERIFIED_FLAG	Receipt verified flag.
RECEIPT_MISSING_FLAG	RECEIPT_MISSING_FLAG	Receipt missing flag.
JUSTIFICATION	JUSTIFICATION	Justification.

Field Name	Maps to Column	Description
EXPENSE_GROUP	EXPENSE_GROUP	Expense Group for web Employees.
START_EXPENSE_DATE	START_EXPENSE_DATE	Start expense date.
END_EXPENSE_DATE	END_EXPENSE_DATE	End expense date.
RECEIPT_CURRENCY_CODE	RECEIPT_CURRENCY_CODE	Receipt currency code.
RECEIPT_CONVERSION_DATE	RECEIPT_CONVERSION_DATE	Receipt conversion date.
DAILY_AMOUNT	DAILY_AMOUNT	Daily amount for web employees.
RECEIPT_CURRENCY_AMOUNT	RECEIPT_CURRENCY_AMOUNT	Receipt currency amount for web employees.
WEB_PARAMETER_ID	WEB_PARAMETER_ID	Web parameter ID for web employees.
AMOUNT_INCLUDES_TAX_FLAG	AMOUNT_INCLUDES_TAX_FLAG	Amount includes tax.
ORGANIZATION_NAME	ORG_ID	Organization name validated against HR_ALL_ORGANIZATION_UNITS.NAME.

Receive Auto Invoice Service

The name of this service is:

WmOAFIN107SC.receiveables107SC.intoOA.autoInvoice:receiveAutoInvoice

This service imports and validates transaction data from other financial systems, and creates invoices, debit memos, credit memos, and on-account credits in Oracle Receivables.

Although you can run the AutoInvoice Import Concurrent Program (AutoInvoice Master Program) with this service, it is recommended to schedule the program from Oracle Applications for the following reasons:

- The parameters for the AutoInvoice Master Program do not allow selective documents to be processed by the program in an instance of the flow. Therefore, the program execution also processes documents not inserted into interface tables during that instance of the flow.

- Although import errors are logged in tables, you cannot selectively retrieve errors for documents loaded during an instance of the flow. All errors for the Batch Source Name will display in the flow.

Database Scripts

This service uses the following database scripts:

Database Script	Description
wm_install_into_autoinvoice.sql	Installs all components for the service.
wm_into_autonvoice_pkg.sql	Installs WM_AUTOINVOICE_INV_IMP_HANDLER_PKG.WM_HANDLE_AUTOINVOICE, which calls the WM_CONC_REQUEST.WM_REQUEST_SUBMIT procedure to submit the Accounts Receivable process.
wm_drop_into_autoinvoice.sql	Uninstalls all components created by wm_install_into_autoinvoice.sql.

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definitions:

- setAutoInvoiceTxn107SC.txp
- AutoInvoiceTransactions107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow executes as follows:

- **specifyDefaultSettings** specifies the default parameter settings required for service execution. You should change these settings accordingly.
- **specifyConcProgParams** specifies the default parameter settings required for concurrent program execution. You should change these settings accordingly.
- **bizDocMapping** maps the incoming business document structure to the required Oracle Applications data structures (interface tables).
 - **getSetOfBooksId** is a transformer that maps the business document IData structure to the interface table IData structure. It takes SET_OF_BOOKS_NAME as the

input parameter and queries the table GL_SETS_OF_BOOKS and gets the SET_OF_BOOKS_ID.

- **getInventoryItemId** gets the INVENTORY_ITEM_ID from MTL_SYSTEM_ITEMS_KFV and HR_ALL_ORGANIZATION_UNITS based on the ITEM_CODE and ORGANIZATION_NAME.
- **getCodeCombinationID** is a transformer that maps the business doc Idata structure to the interface table Idata structure. It takes CONCATENATED_SEGMENTS and SET_OF_BOOKS_NAME as the input parameters and queries the tables GL_CODE_COMBINATIONS_KFV and GL_SETS_OF_BOOKS to get the CODE_COMBINATION_ID.
- **convertToDateObject** converts a date string to the date-object.
- **getOrgId** takes ORGANIZATION_NAME as the input parameter and queries the table ORG_ORGANIZATION_DEFINITIONS. It also gets the ORGANIZATION_ID corresponding to the ORGANIZATION_NAME.
- **getTerritoryId** is a transformer that maps the business doc Idata structure to the interface table Idata structure. It takes TERRITORY_NAME as the input parameter and queries the table RA_TERRITORIES and gets the TERRITORY_ID.
- **getSetOfBooksId** is a transformer that maps the business doc Idata structure to the interface table Idata structure. It takes SET_OF_BOOKS_NAME as the input parameter and queries the table GL_SETS_OF_BOOKS and gets the SET_OF_BOOKS_ID.
- **setAutoInvoiceTxn** inserts data into the interface tables.
- **importAutoInvoice** imports data to the production table from the interface table. It calls the execAutoInvConcProg, checkAutoInvImportStatus, and getAutoInvoiceImport_ERR services to execute the corresponding concurrent program that inserts data into the production table and to generate the error/acknowledgement message. If the status of the execution is SUCCESS (returned by the service execAutoInvConcProg), it checks for the record with the current BATCH_NAME in the interface table. If any are found, it generates an error during import. In this case, this service calls getAutoInvoiceImport_ERR to retrieve the errors. If no record is found, it comes out of the flow and indicates that the data import process is successful. If the status of the execution is FAILED, it discontinues execution.
- **execAutoInvConcProg** invokes the stored procedure WM_AUTOINVOICE_IMP_HANDLER_PKG.WM_HANDLE_AUTOINVOICE that then calls corresponding concurrent subroutine to execute the data import process for AutoInvoice into Oracle Applications. This service provides Status ID, Request ID, and Execution Status Message for normal concurrent program completion and a database Stored Procedure error message if an exception occurs in the Stored Procedure execution.

- **checkAutoInvlmportStatus** checks the status of the execution by checking the interface table for any rejected record corresponding to the current `BATCH_SOURCE_NAME`. If the query does not return any rows, it indicates a successful import. If the query returns any row, it indicates that the concurrent program could not import data successfully into the production tables of Oracle Applications.
- **getAutoInvoiceImport_ERR** gets the error message that occurs during the data import to the production table from interface table. Based on the parameter `BATCH_SOURCE_NAME`, it scans table `RA_INTERFACE_LINES_ALL` and `RA_INTERFACE_ERRORS_ALL` to get the corresponding message for the `BATCH_SOURCE_NAME`.

For more details on receive transactions, see [“Overview of Receive Service Transaction Processing” on page 31](#).

Business Document Structure

This service uses the business document `autoInvoiceBizDoc`. Its structure is as follows:

- 1.0 TRANSACTIONS
 - 1.1 TRANSACTION_LINES
 - 1.1.1 TAX_LINES
 - **1.1.1.1 GL_DISTRIBUTIONS
 - 1.1.2 FREIGHT_LINES
 - **1.1.2.1 GL_DISTRIBUTIONS
 - 1.1.3 SALES_CREDITS
 - **1.1.3.1 GL_DISTRIBUTIONS
 - 1.2 CHARGES
 - **1.2.1 GL_DISTRIBUTIONS
 - 1.3 FREIGHT_LINES
 - **1.3.1 GL_DISTRIBUTIONS
 - 1.4 ***DEFAULT_SALES_CREDITS
 - 1.5 ***NOTES

**All `GL_DISTRIBUTIONS` use the same table. See [“GL_DISTRIBUTIONS” on page 90](#).

1.0 TRANSACTIONS (Maps to RA_INTERFACE_LINES_ALL)



Note: ***NOTES and DEFAULT_SALES_CREDITS are currently not used.

Field Name	Maps to Column	Description
AGREEMENT_NAME	AGREEMENT_NAME	The name of the customer agreement for this transaction.
COMMENTS	COMMENTS	Comments about this transaction.
CONS_BILLING_NUM	CONS_BILLING_NUM	The number for this consolidated bill. A consolidated bill number groups a set of invoices under one bill.
CONVERSION_DATE	CONVERSION_DATE	The exchange rate date for this transaction. If you do not enter a date, AutoInvoice uses the transaction date as the default.
CONVERSION_RATE	CONVERSION_RATE	The exchange rate for this transaction.
CONVERSION_TYPE	CONVERSION_TYPE	Required. The exchange rate type for this transaction. If the currency of the transaction is the same as the base currency, use the value <code>User</code> and set <code>CONVERSION_RATE</code> to 1.
CREDIT_METHOD_FOR_RULES	CREDIT_METHOD_FOR_RULES	The credit method for crediting a transaction, which uses an accounting rule. Valid values are PRORATE, LIFO, and UNIT.
CREDIT_METHOD_FOR_INSTALLMENTS	CREDIT_METHOD_FOR_INSTALLMENTS	The credit method for crediting a transaction that uses split payment terms. Valid values are PRORATE, LIFO, and UNIT.
INVOICE_CURRENCY_CODE	CURRENCY_CODE	Required. The currency code for this transaction.
CUSTOMER_BANK_ACCOUNT_NAME	CUSTOMER_BANK_ACCOUNT_NAME	The Bill-to customer bank account name for this transaction.
TRX_TYPE_NAME	CUST_TRX_TYPE_NAME	The Transaction Type name for this transaction.
DOCUMENT_NUMBER	DOCUMENT_NUMBER	The Document Number for this transaction.

Field Name	Maps to Column	Description
GL_DATE	GL_DATE	The General Ledger Date for this transaction. This date determines the accounting period that you record this transaction to your general ledger.
HEADER_ATTRIBUTE_CATEGORY	HEADER_ATTRIBUTE_CATEGORY	Descriptive flexfield attribute information for the transaction information flexfield. Descriptive flexfield attributes let you store additional columns, the contents of which you define.
HEADER_ATTRIBUTE1	HEADER_ATTRIBUTE1	
HEADER_ATTRIBUTE2	HEADER_ATTRIBUTE2	
HEADER_ATTRIBUTE3	HEADER_ATTRIBUTE3	
HEADER_ATTRIBUTE4	HEADER_ATTRIBUTE4	
HEADER_ATTRIBUTE5	HEADER_ATTRIBUTE5	
HEADER_ATTRIBUTE6	HEADER_ATTRIBUTE6	
HEADER_ATTRIBUTE7	HEADER_ATTRIBUTE7	
HEADER_ATTRIBUTE8	HEADER_ATTRIBUTE8	
HEADER_ATTRIBUTE9	HEADER_ATTRIBUTE9	
HEADER_ATTRIBUTE10	HEADER_ATTRIBUTE10	
HEADER_ATTRIBUTE11	HEADER_ATTRIBUTE11	
HEADER_ATTRIBUTE12	HEADER_ATTRIBUTE12	
HEADER_ATTRIBUTE13	HEADER_ATTRIBUTE13	
HEADER_ATTRIBUTE14	HEADER_ATTRIBUTE14	
HEADER_ATTRIBUTE15	HEADER_ATTRIBUTE15	
INTERNAL_NOTES	INTERNAL_NOTES	Internal notes for this transaction.
INVOICING_RULE_NAME	INVOICING_RULE_NAME	The invoicing rule name for this transaction.
ORIG_SYSTEM_BILL_ADDRESS_REF	ORIG_SYSTEM_BILL_ADDRESS_REF	The Bill-To customer address reference from your original system. This reference is for the Bill-To customer you entered in ORIG_SYSTEM_BILL_CUSTOMER_REF.
ORIG_SYSTEM_BILL_CONTACT_REF	ORIG_SYSTEM_BILL_CONTACT_REF	The Bill-To contact reference from your original system. This reference is for the Bill-To customer that you entered in ORIG_SYSTEM_BILL_CUSTOMER_REF.

Field Name	Maps to Column	Description
ORIG_SYSTEM_BILL_CUSTOMER_REF	ORIG_SYSTEM_BILL_CUSTOMER_REF	Uniquely identifies this Bill-To customer in your original system. The reference value you enter here provides you with an audit trail from Oracle Receivables back to your original system.
ORIG_SYSTEM_SHIP_ADDRESS_REF	ORIG_SYSTEM_SHIP_ADDRESS_REF	Uniquely identifies this Ship-To customer address in your original system.
ORIG_SYSTEM_SHIP_CONTACT_REF	ORIG_SYSTEM_SHIP_CONTACT_REF	Uniquely identifies this Ship-To contact in your original system.
ORIG_SYSTEM_SHIP_CUSTOMER_REF	ORIG_SYSTEM_SHIP_CUSTOMER_REF	Uniquely identifies this Ship-to customer in your original system.
ORIG_SYSTEM_SOLD_CUSTOMER_REF	ORIG_SYSTEM_SOLD_CUSTOMER_REF	Uniquely identifies this Sold-to customer in your original system.
ORIG_SYSTEM_BATCH_NAME	ORIG_SYSTEM_BATCH_NAME	The batch name for this transaction.
PRIMARY_SALESREP_NUMBER	PRIMARY_SALESREP_NUMBER	The primary Salesperson Number for this transaction.
PRINTING_OPTION	PRINTING_OPTION	The printing option for this transaction.
PURCHASE_ORDER	PURCHASE_ORDER	The Purchase Order Number for this transaction.
PURCHASE_ORDER_REVISION	PURCHASE_ORDER_REVISION	The Purchase Order Revision for this transaction.
PURCHASE_ORDER_DATE	PURCHASE_ORDER_DATE	The date of the purchase order for this transaction.
REASON_CODE	REASON_CODE	The reason code for this transaction.
RECEIPT_METHOD_NAME	RECEIPT_METHOD_NAME	The name of the Payment Method for this transaction.
RELATED_CUSTOMER_TRX_NUMBER	RELATED_TRX_NUMBER	The document number to which this transaction is related.
SET_OF_BOOKS_NAME	SET_OF_BOOKS_ID	Required. Maps to the set of Books ID for this transaction.
TERRITORY	TERRITORY	Maps to the Territory ID for this transaction.
TERM_NAME	TERM_NAME	The name of the payment term for this transaction.
TRX_DATE	TRX_DATE	The transaction date for this transaction.

Field Name	Maps to Column	Description
TRX_NUMBER	TRX_NUMBER	The number for this transaction.
ATTRIBUTE_CATEGORY	ATTRIBUTE_CATEGORY	The Invoice Line Information flexfield attribute information for this transaction. Descriptive flexfield attributes allow you to store additional columns, the contents of which you define.
ATTRIBUTE1	ATTRIBUTE1	
ATTRIBUTE8	ATTRIBUTE8	
ATTRIBUTE2	ATTRIBUTE2	
ATTRIBUTE3	ATTRIBUTE3	
ATTRIBUTE4	ATTRIBUTE4	
ATTRIBUTE5	ATTRIBUTE5	
ATTRIBUTE6	ATTRIBUTE6	
ATTRIBUTE7	ATTRIBUTE7	
ATTRIBUTE9	ATTRIBUTE9	
ATTRIBUTE10	ATTRIBUTE10	
ATTRIBUTE11	ATTRIBUTE11	
ATTRIBUTE12	ATTRIBUTE12	
ATTRIBUTE13	ATTRIBUTE13	
ATTRIBUTE14	ATTRIBUTE14	
ATTRIBUTE15	ATTRIBUTE15	
RELATED_BATCH_SOURCE_NAME	RELATED_BATCH_SOURCE_NAME	The name of the batch source of the document to which this transaction is related.
BATCH_SOURCE_NAME	BATCH_SOURCE_NAME	The name of the batch source for this transaction. AutoInvoice uses your batch source to determine your transaction and batch numbering method and your AutoInvoice processing options. You must enter a value in this column.
FOB_POINT	FOB_POINT	The FOB point for this transaction.
SHIP_DATE_ACTUAL	SHIP_DATE_ACTUAL	The Shipment Date for this transaction.
SHIP_VIA	SHIP_VIA	The Ship Via Code for this transaction.
WAYBILL_NUMBER	WAYBILL_NUMBER	The Waybill number for this transaction.

Field Name	Maps to Column	Description
DEFAULT_USSGL_TRANSACTION_CODE	USSGL_TRANSACTION_CODE	The transaction code for this transaction. If this transaction is linked to another transaction, you must enter the same transaction code as the one to which it is linked.
DEFAULT_USSGL_TRX_CODE_CONTEXT	DEFAULT_USSGL_TRX_CODE_CONTEXT	AutoInvoice does not currently use column.
ORGANIZATION_NAME	ORG_ID	Added for future Oracle Applications functionality (leave this field blank). However, note that ORGANIZATION_NAME is used to retrieve other Oracle Applications Ids such as Inventory ID.

1.1 TRANSACTION_LINES (Maps to RA_INTERFACE_LINES_ALL))

Field Name	Maps to Column	Description
DESCRIPTION	DESCRIPTION	Required. Enter the description for this transaction.
QUANTITY	QUANTITY	This column is optional if the transaction is an Invoice or Credit Memo Line and LINE_TYPE = LINE, or you are passing Header Freight. Type values as follows: <ul style="list-style-type: none"> ■ Invoice Lines: the number of units shipped. ■ Credit Memo lines: the number of units you are crediting. If you do not enter a value in this column, AutoInvoice uses AMOUNT as the extended amount for this transaction.
QUANTITY_ORDERED	QUANTITY_ORDERED	The original number of units ordered for this transaction.
UNIT_STANDARD_PRICE	UNIT_STANDARD_PRICE	The standard price per unit for this transaction.

Field Name	Maps to Column	Description
UNIT_SELLING_PRICE	UNIT_SELLING_PRICE	The selling price per unit for this transaction.
AMOUNT	AMOUNT	The revenue amount for this transaction.
AMOUNT_INCLUDES_TAX_FLAG	AMOUNT_INCLUDES_TAX_FLAG	This column controls whether the amount for this transaction line includes tax.
UOM_CODE	UOM_CODE	The unit of measure code for this transaction.
UOM_NAME	UOM_NAME	The unit of measure name for this transaction.
ACCOUNTING_RULE_DURATION	ACCOUNTING_RULE_DURATION	The accounting rule duration for this transaction.
ACCOUNTING_RULE_NAME	ACCOUNTING_RULE_NAME	The Accounting Rule name for this transaction.
RULE_START_DATE	RULE_START_DATE	The date that you want to start the aCcounting Rule for this transaction.
LAST_PERIOD_TO_CREDIT	LAST_PERIOD_TO_CREDIT	For Unit Credit Memos, enter the last period number from which you want to start crediting.
INVENTORY_ITEM	INVENTORY_ID	The concatenated Inventory flexfiled, which will be used to derive the INVENTORY_ID for this transaction.
MEMO_LINE_NAME	MEMO_LINE_NAME	The name of the standard memo line for this transaction.
TAX_EXEMPT_FLAG	TAX_EXEMPT_FLAG	If LINE_TYPE = LINE, this column is optional. The value you enter here controls how a line is taxed. For all other line types and Credit Memos,, do not enter a value in this column.
TAX_EXEMPT_NUMBER	TAX_EXEMPT_NUMBER	The Tax Exempt Number for this transaction.
TAX_EXEMPT_REASON_CODE	TAX_EXEMPT_REASON_CODE	The Tax Exempt Reason Code for this transaction.

Field Name	Maps to Column	Description
TAX_EXEMPT_REASON_CODE_MEANING	TAX_EXEMPT_REASON_CODE_MEANING	The Tax Exempt Reason Code meaning for this transaction.
SALES_ORDER_SOURCE	SALES_ORDER_SOURCE	The source of the Sales Order for this transaction.
SALES_ORDER	SALES_ORDER	The Sales Order Number for this transaction.
SALES_ORDER_REVISION	SALES_ORDER_REVISION	The Sales Order Revision for this transaction.
SALES_ORDER_LINE	SALES_ORDER_LINE	The Sales Order Line Number for this transaction.
SALES_ORDER_DATE	SALES_ORDER_DATE	The date of the Sales Order for this transaction.
WHEREHOUSE_NAME		Name of the warehouse.
TRANSLATED_DESCRIPTION		Translated description.
INTERFACE_LINE_CONTEXT	INTERFACE_LINE_CONTEXT	The line transaction flexfield for this transaction. the line transaction flexfield is a combination of attribute values that you use to uniquely identify this transaction line in your original system. The reference value you enter here provides you with an audit trail from Receivables back to your original system. You must enter values for enabled attributes.
INTERFACE_LINE_ATTRIBUTE1	INTERFACE_LINE_ATTRIBUTE1	
INTERFACE_LINE_ATTRIBUTE2	INTERFACE_LINE_ATTRIBUTE2	
INTERFACE_LINE_ATTRIBUTE3	INTERFACE_LINE_ATTRIBUTE3	
INTERFACE_LINE_ATTRIBUTE4	INTERFACE_LINE_ATTRIBUTE4	
INTERFACE_LINE_ATTRIBUTE5	INTERFACE_LINE_ATTRIBUTE5	

Field Name	Maps to Column	Description
INTERFACE_LINE_ATTRIBUTE6	INTERFACE_LINE_ATTRIBUTE6	
INTERFACE_LINE_ATTRIBUTE7	INTERFACE_LINE_ATTRIBUTE7	
INTERFACE_LINE_ATTRIBUTE8	INTERFACE_LINE_ATTRIBUTE8	
INTERFACE_LINE_ATTRIBUTE9	INTERFACE_LINE_ATTRIBUTE9	
INTERFACE_LINE_ATTRIBUTE10	INTERFACE_LINE_ATTRIBUTE10	
INTERFACE_LINE_ATTRIBUTE11	INTERFACE_LINE_ATTRIBUTE11	
INTERFACE_LINE_ATTRIBUTE13	INTERFACE_LINE_ATTRIBUTE13	
INTERFACE_LINE_ATTRIBUTE15	INTERFACE_LINE_ATTRIBUTE15	
INTERFACE_LINE_ATTRIBUTE12	INTERFACE_LINE_ATTRIBUTE12	
INTERFACE_LINE_ATTRIBUTE14	INTERFACE_LINE_ATTRIBUTE14	
REFERENCE_LINE_CONTEXT	REFERENCE_LINE_CONTEXT	The transaction flexfield of the transaction line you are crediting in these columns.
REFERENCE_LINE_ATTRIBUTE1	REFERENCE_LINE_ATTRIBUTE1	
REFERENCE_LINE_ATTRIBUTE2	REFERENCE_LINE_ATTRIBUTE2	
REFERENCE_LINE_ATTRIBUTE3	REFERENCE_LINE_ATTRIBUTE3	
REFERENCE_LINE_ATTRIBUTE4	REFERENCE_LINE_ATTRIBUTE4	
REFERENCE_LINE_ATTRIBUTE5	REFERENCE_LINE_ATTRIBUTE5	
REFERENCE_LINE_ATTRIBUTE6	REFERENCE_LINE_ATTRIBUTE6	

Field Name	Maps to Column	Description
REFERENCE_LINE_ATTRIBUTE7	REFERENCE_LINE_ATTRIBUTE7	
REFERENCE_LINE_ATTRIBUTE8	REFERENCE_LINE_ATTRIBUTE8	
REFERENCE_LINE_ATTRIBUTE9	REFERENCE_LINE_ATTRIBUTE9	
REFERENCE_LINE_ATTRIBUTE10	REFERENCE_LINE_ATTRIBUTE10	
REFERENCE_LINE_ATTRIBUTE11	REFERENCE_LINE_ATTRIBUTE11	
REFERENCE_LINE_ATTRIBUTE12	REFERENCE_LINE_ATTRIBUTE12	
REFERENCE_LINE_ATTRIBUTE13	REFERENCE_LINE_ATTRIBUTE13	
REFERENCE_LINE_ATTRIBUTE14	REFERENCE_LINE_ATTRIBUTE14	
REFERENCE_LINE_ATTRIBUTE15	REFERENCE_LINE_ATTRIBUTE15	
LINK_TO_LINE_CONTEXT	LINK_TO_LINE_CONTEXT	The link to your transaction flexfield attribute values.
LINK_TO_LINE_ATTRIBUTE1	LINK_TO_LINE_ATTRIBUTE1	
LINK_TO_LINE_ATTRIBUTE2	LINK_TO_LINE_ATTRIBUTE2	
LINK_TO_LINE_ATTRIBUTE3	LINK_TO_LINE_ATTRIBUTE3	
LINK_TO_LINE_ATTRIBUTE4	LINK_TO_LINE_ATTRIBUTE4	
LINK_TO_LINE_ATTRIBUTE5	LINK_TO_LINE_ATTRIBUTE5	
LINK_TO_LINE_ATTRIBUTE6	LINK_TO_LINE_ATTRIBUTE6	
LINK_TO_LINE_ATTRIBUTE7	LINK_TO_LINE_ATTRIBUTE7	
LINK_TO_LINE_ATTRIBUTE8	LINK_TO_LINE_ATTRIBUTE8	
LINK_TO_LINE_ATTRIBUTE9	LINK_TO_LINE_ATTRIBUTE9	
LINK_TO_LINE_ATTRIBUTE10	LINK_TO_LINE_ATTRIBUTE10	
LINK_TO_LINE_ATTRIBUTE11	LINK_TO_LINE_ATTRIBUTE11	
LINK_TO_LINE_ATTRIBUTE12	LINK_TO_LINE_ATTRIBUTE12	
LINK_TO_LINE_ATTRIBUTE13	LINK_TO_LINE_ATTRIBUTE13	

Field Name	Maps to Column	Description
LINK_TO_LINE_ATTRIBUTE14	LINK_TO_LINE_ATTRIBUTE14	
LINK_TO_LINE_ATTRIBUTE15	LINK_TO_LINE_ATTRIBUTE15	

1.1.1 TAX_LINES (Maps to RA_INTERFACE_LINES_ALL)

Field Name	Maps to Column	Description
AMOUNT	AMOUNT	The revenue amount for this transaction. A value must be entered in either this column or the TAX_RATE column.
VAT_TAX_CODE	TAX_CODE	The tax code for this tax line.
TAX_PRECEDENCE	TAX_PRECEDENCE	The precedence number for this tax line. Computes tax compounding.
TAX_RATE	TAX_RATE	The tax rate for this tax line. You must enter a value in this column or the AMOUNT column.
INTERFACE_LINE_CONTEXT	INTERFACE_LINE_CONTEXT	The line transaction flexfield for this transaction. The line transaction flexfield is a combination of attribute values that you use to uniquely identify this transaction line in your original system. The reference value you enter here provides you with an audit trail from receivables back to your original system. You must enter values for enabled attributes.
INTERFACE_LINE_ATTRIBUTE1	INTERFACE_LINE_ATTRIBUTE1	
INTERFACE_LINE_ATTRIBUTE2	INTERFACE_LINE_ATTRIBUTE2	
INTERFACE_LINE_ATTRIBUTE3	INTERFACE_LINE_ATTRIBUTE3	
INTERFACE_LINE_ATTRIBUTE4	INTERFACE_LINE_ATTRIBUTE4	
INTERFACE_LINE_ATTRIBUTE5	INTERFACE_LINE_ATTRIBUTE5	
INTERFACE_LINE_ATTRIBUTE6	INTERFACE_LINE_ATTRIBUTE6	
INTERFACE_LINE_ATTRIBUTE7	INTERFACE_LINE_ATTRIBUTE7	

Field Name	Maps to Column	Description
INTERFACE_LINE_ATTRIBUTE8	INTERFACE_LINE_ATTRIBUTE8	
INTERFACE_LINE_ATTRIBUTE9	INTERFACE_LINE_ATTRIBUTE9	
INTERFACE_LINE_ATTRIBUTE10	INTERFACE_LINE_ATTRIBUTE10	
INTERFACE_LINE_ATTRIBUTE11	INTERFACE_LINE_ATTRIBUTE11	
INTERFACE_LINE_ATTRIBUTE12	INTERFACE_LINE_ATTRIBUTE12	
INTERFACE_LINE_ATTRIBUTE13	INTERFACE_LINE_ATTRIBUTE13	
INTERFACE_LINE_ATTRIBUTE14	INTERFACE_LINE_ATTRIBUTE14	
INTERFACE_LINE_ATTRIBUTE15	INTERFACE_LINE_ATTRIBUTE15	
ATTRIBUTE_CATEGORY	ATTRIBUTE_CATEGORY	The invoice line information flexfield attribute information for this transaction. Descriptive flexfield attributes allow you to store additional columns, the contents of which you define.
ATTRIBUTE1	ATTRIBUTE1	
ATTRIBUTE2	ATTRIBUTE2	
ATTRIBUTE3	ATTRIBUTE3	
ATTRIBUTE4	ATTRIBUTE4	
ATTRIBUTE5	ATTRIBUTE5	
ATTRIBUTE6	ATTRIBUTE6	
ATTRIBUTE7	ATTRIBUTE7	
ATTRIBUTE8	ATTRIBUTE8	
ATTRIBUTE9	ATTRIBUTE9	
ATTRIBUTE10	ATTRIBUTE10	
ATTRIBUTE11	ATTRIBUTE11	
ATTRIBUTE12	ATTRIBUTE12	
ATTRIBUTE13	ATTRIBUTE13	

Field Name	Maps to Column	Description
ATTRIBUTE14	ATTRIBUTE14	
ATTRIBUTE15	ATTRIBUTE15	

1.1.2 FREIGHT_LINES (Maps to RA_INTERFACE_LINES_ALL)

Field Name	Maps to Column	Description
AMOUNT	AMOUNT	Required. The revenue amount for this transaction.
INTERFACE_LINE_CONTEXT	INTERFACE_LINE_CONTEXT	Required for enabled attributes. The line transaction flexfield for this transaction. The line transaction flexfield is a combination of attribute values that you use to uniquely identify this transaction line in your original system. The reference value you enter here provides you with an audit trail from receivables back to your original system.
INTERFACE_LINE_ATTRIBUTE1	INTERFACE_LINE_ATTRIBUTE1	
INTERFACE_LINE_ATTRIBUTE2	INTERFACE_LINE_ATTRIBUTE2	
INTERFACE_LINE_ATTRIBUTE3	INTERFACE_LINE_ATTRIBUTE3	
INTERFACE_LINE_ATTRIBUTE4	INTERFACE_LINE_ATTRIBUTE4	
INTERFACE_LINE_ATTRIBUTE5	INTERFACE_LINE_ATTRIBUTE5	
INTERFACE_LINE_ATTRIBUTE6	INTERFACE_LINE_ATTRIBUTE6	
INTERFACE_LINE_ATTRIBUTE7	INTERFACE_LINE_ATTRIBUTE7	
INTERFACE_LINE_ATTRIBUTE8	INTERFACE_LINE_ATTRIBUTE8	
INTERFACE_LINE_ATTRIBUTE9	INTERFACE_LINE_ATTRIBUTE9	
INTERFACE_LINE_ATTRIBUTE10	INTERFACE_LINE_ATTRIBUTE10	

Field Name	Maps to Column	Description
INTERFACE_LINE_ATTRIBUTE11	INTERFACE_LINE_ATTRIBUTE11	
INTERFACE_LINE_ATTRIBUTE12	INTERFACE_LINE_ATTRIBUTE12	
INTERFACE_LINE_ATTRIBUTE13	INTERFACE_LINE_ATTRIBUTE13	
INTERFACE_LINE_ATTRIBUTE14	INTERFACE_LINE_ATTRIBUTE14	
INTERFACE_LINE_ATTRIBUTE15	INTERFACE_LINE_ATTRIBUTE15	
ATTRIBUTE_CATEGORY	ATTRIBUTE_CATEGORY	The invoice line information flexfield attribute information for this transaction. Descriptive flexfield attributes allow you to store additional columns, the contents of which you define.
ATTRIBUTE1	ATTRIBUTE1	
ATTRIBUTE2	ATTRIBUTE2	
ATTRIBUTE3	ATTRIBUTE3	
ATTRIBUTE4	ATTRIBUTE4	
ATTRIBUTE5	ATTRIBUTE5	
ATTRIBUTE6	ATTRIBUTE6	
ATTRIBUTE7	ATTRIBUTE7	
ATTRIBUTE8	ATTRIBUTE8	
ATTRIBUTE9	ATTRIBUTE9	
ATTRIBUTE10	ATTRIBUTE10	
ATTRIBUTE11	ATTRIBUTE11	
ATTRIBUTE12	ATTRIBUTE12	
ATTRIBUTE13	ATTRIBUTE13	
ATTRIBUTE14	ATTRIBUTE14	
ATTRIBUTE15	ATTRIBUTE15	

1.1.3 SALES_CREDITS (RA_INTERFACE_SALESCREDITS_ALL)

Field Name	Maps to Column	Description
SALESREP_NUMBER	SALESREP_NUMBER	Required. Enter the salesperson number for this sales credit assignment.
REVENUE_PERCENT_SPLIT	SALES_CREDIT_PERCENT_SPLIT	The sales credit percent for this salesperson. You must enter a value in this column or in the REVENUE_AMOUNT_SPLIT.
REVENUE_AMOUNT_SPLIT	SALES_CREDIT_AMOUNT_SPLIT	The sales credit amount for this salesperson. You must enter a value in this column or in the REVENUE_PERCENT_SPLIT.
NON_REVENUE_PERCENT_SPLIT		Not used
NON_REVENUE_AMOUNT_SPLIT		Not used
SALESREP_NAME	SALESREP_NAME	Not used
SALES_CREDIT_TYPE_NAME	SALES_CREDIT_TYPE_NAME	Required. The name of the sales credit type for this sales credit assignment.
INTERFACE_LINE_CONTEXT	INTERFACE_LINE_CONTEXT	The line transaction flexfield for this transaction. The line transaction flexfield is a combination of attribute values that you use to uniquely identify this transaction line in your original system. The reference value you enter here provides you with an audit trail from receivables back to your original system. You must enter values for enabled attributes.
INTERFACE_LINE_ATTRIBUTE1	INTERFACE_LINE_ATTRIBUTE1	
INTERFACE_LINE_ATTRIBUTE2	INTERFACE_LINE_ATTRIBUTE2	
INTERFACE_LINE_ATTRIBUTE3	INTERFACE_LINE_ATTRIBUTE3	
INTERFACE_LINE_ATTRIBUTE4	INTERFACE_LINE_ATTRIBUTE4	
INTERFACE_LINE_ATTRIBUTE5	INTERFACE_LINE_ATTRIBUTE5	

Field Name	Maps to Column	Description
INTERFACE_LINE_ATTRIBUTE6	INTERFACE_LINE_ATTRIBUTE6	
INTERFACE_LINE_ATTRIBUTE7	INTERFACE_LINE_ATTRIBUTE7	
INTERFACE_LINE_ATTRIBUTE8	INTERFACE_LINE_ATTRIBUTE8	
INTERFACE_LINE_ATTRIBUTE9	INTERFACE_LINE_ATTRIBUTE9	
INTERFACE_LINE_ATTRIBUTE10	INTERFACE_LINE_ATTRIBUTE10	
INTERFACE_LINE_ATTRIBUTE11	INTERFACE_LINE_ATTRIBUTE11	
INTERFACE_LINE_ATTRIBUTE12	INTERFACE_LINE_ATTRIBUTE12	
INTERFACE_LINE_ATTRIBUTE13	INTERFACE_LINE_ATTRIBUTE13	
INTERFACE_LINE_ATTRIBUTE14	INTERFACE_LINE_ATTRIBUTE14	
INTERFACE_LINE_ATTRIBUTE15	INTERFACE_LINE_ATTRIBUTE15	
ATTRIBUTE_CATEGORY	ATTRIBUTE_CATEGORY	The invoice line information flexfield attribute information for this transaction. Descriptive flexfield attributes allow you to store additional columns, the contents of which you define.
ATTRIBUTE1	ATTRIBUTE1	
ATTRIBUTE2	ATTRIBUTE2	
ATTRIBUTE3	ATTRIBUTE3	
ATTRIBUTE4	ATTRIBUTE4	
ATTRIBUTE5	ATTRIBUTE5	
ATTRIBUTE6	ATTRIBUTE6	
ATTRIBUTE7	ATTRIBUTE7	
ATTRIBUTE8	ATTRIBUTE8	
ATTRIBUTE9	ATTRIBUTE9	
ATTRIBUTE10	ATTRIBUTE10	

Field Name	Maps to Column	Description
ATTRIBUTE11	ATTRIBUTE11	
ATTRIBUTE12	ATTRIBUTE12	
ATTRIBUTE13	ATTRIBUTE13	
ATTRIBUTE14	ATTRIBUTE14	
ATTRIBUTE15	ATTRIBUTE15	

GL_DISTRIBUTIONS



Note: All GL Distributions use the same table, as follows.

Field Name	Maps to Column	Description
ACCOUNT_CLASS	ACCOUNT_CLASS	Required. Enter the account class for this accounting distribution. AutoInvoice uses the account class you enter here to determine the type of account you are supplying for this accounting distribution.
ACCTD_AMOUNT	ACCTD_AMOUNT	The accounted amount for this distribution.
AMOUNT	AMOUNT	The amount for this accounting distribution. If this accounting distribution is for a transaction that does not use an accounting rule and depending on the value you entered for your batch source, you must enter either a value in this column or in PERCENT.
ATTRIBUTE_CATEGORY	ATTRIBUTE_CATEGORY	The invoice line information flexfield attribute information for this transaction. Descriptive flexfield attributes allow you to store additional columns, the contents of which you define.
ATTRIBUTE1	ATTRIBUTE1	
ATTRIBUTE2	ATTRIBUTE2	
ATTRIBUTE3	ATTRIBUTE3	
ATTRIBUTE4	ATTRIBUTE4	
ATTRIBUTE5	ATTRIBUTE5	
ATTRIBUTE6	ATTRIBUTE6	
ATTRIBUTE7	ATTRIBUTE7	

Field Name	Maps to Column	Description
ATTRIBUTE8	ATTRIBUTE8	
ATTRIBUTE9	ATTRIBUTE9	
ATTRIBUTE10	ATTRIBUTE10	
ATTRIBUTE11	ATTRIBUTE11	
ATTRIBUTE12	ATTRIBUTE12	
ATTRIBUTE13	ATTRIBUTE13	
ATTRIBUTE14	ATTRIBUTE14	
ATTRIBUTE15	ATTRIBUTE15	
ACCOUNT_NUMBER	CODE_COMBINATION_ID	Required. The concatenated segments value for the Accounting flexfields, which maps to the CODE_COMBINATION_ID.
SEGMENT1	SEGMENT1	Required for enabled segments. The accounting flexfield for this transaction. The accounting flexfield is a combination of segment values that you use to uniquely identify this account number in your original system. The reference value you enter here provides you with an audit trail from receivables back to your original system.
SEGMENT2	SEGMENT2	
SEGMENT3	SEGMENT3	
SEGMENT4	SEGMENT4	
SEGMENT5	SEGMENT5	
SEGMENT6	SEGMENT6	
SEGMENT7	SEGMENT7	
SEGMENT8	SEGMENT8	
SEGMENT9	SEGMENT9	
SEGMENT10	SEGMENT10	
SEGMENT11	SEGMENT11	
SEGMENT12	SEGMENT12	
SEGMENT13	SEGMENT13	
SEGMENT14	SEGMENT14	
SEGMENT15	SEGMENT15	

Field Name	Maps to Column	Description
SEGMENT16	SEGMENT16	
SEGMENT17	SEGMENT17	
SEGMENT18	SEGMENT18	
SEGMENT19	SEGMENT19	
SEGMENT20	SEGMENT20	
SEGMENT21	SEGMENT21	
SEGMENT22	SEGMENT22	
SEGMENT23	SEGMENT23	
SEGMENT24	SEGMENT24	
SEGMENT25	SEGMENT25	
SEGMENT26	SEGMENT26	
SEGMENT27	SEGMENT27	
SEGMENT28	SEGMENT28	
SEGMENT29	SEGMENT29	
SEGMENT30	SEGMENT30	
COMMENTS	COMMENTS	Comments about this accounting distribution.
INTERFACE_LINE_CONTEXT	INTERFACE_LINE_CONTEXT	Required for enabled attributes. The line transaction flexfield for this transaction. The line transaction flexfield is a combination of attribute values that you use to uniquely identify this transaction line in your original system. The reference value you enter here provides you with an audit trail from receivables back to your original system.
INTERFACE_LINE_ATTRIBUTE1	INTERFACE_LINE_ATTRIBUTE1	
INTERFACE_LINE_ATTRIBUTE2	INTERFACE_LINE_ATTRIBUTE2	
INTERFACE_LINE_ATTRIBUTE3	INTERFACE_LINE_ATTRIBUTE3	
INTERFACE_LINE_ATTRIBUTE4	INTERFACE_LINE_ATTRIBUTE4	
INTERFACE_LINE_ATTRIBUTE5	INTERFACE_LINE_ATTRIBUTE5	

Field Name	Maps to Column	Description
INTERFACE_LINE_ATTRIBUTE6	INTERFACE_LINE_ATTRIBUTE6	
INTERFACE_LINE_ATTRIBUTE7	INTERFACE_LINE_ATTRIBUTE7	
INTERFACE_LINE_ATTRIBUTE8	INTERFACE_LINE_ATTRIBUTE8	
INTERFACE_LINE_ATTRIBUTE9	INTERFACE_LINE_ATTRIBUTE9	
INTERFACE_LINE_ATTRIBUTE10	INTERFACE_LINE_ATTRIBUTE10	
INTERFACE_LINE_ATTRIBUTE11	INTERFACE_LINE_ATTRIBUTE11	
INTERFACE_LINE_ATTRIBUTE12	INTERFACE_LINE_ATTRIBUTE12	
INTERFACE_LINE_ATTRIBUTE13	INTERFACE_LINE_ATTRIBUTE13	
INTERFACE_LINE_ATTRIBUTE14	INTERFACE_LINE_ATTRIBUTE14	
INTERFACE_LINE_ATTRIBUTE15	INTERFACE_LINE_ATTRIBUTE15	
ACCT_DISTRIBUTION_PERCENT	PERCENT	The percentage for this accounting distribution. You must enter a value in this column or in AMOUNT if the accounting distribution is for a transaction that does not use an accounting rule, and depending on the value you entered for your batch source.

1.2 CHARGES (Maps to RA_INTERFACE_LINES_ALL)

Field Name	Maps to Column	Description
DESCRIPTION	DESCRIPTION	Required. Type the description for this transaction.
QUANTITY	QUANTITY	Number of units shipped. Use: 1 for Debit Memos 1 or -1 for Credit Memos
QUANTITY_ORDERED	QUANTITY_ORDERED	The original number of units ordered for this transaction.

Field Name	Maps to Column	Description
INVENTORY_ITEM	INVENTORY_ITEM	The concatenated inventory flexfield, which will be used to derive the INVENTORY_ID for this transaction.
UNIT_STANDARD_PRICE	UNIT_STANDARD_PRICE	The standard price per unit for this transaction.
UNIT_SELLING_PRICE	UNIT_SELLING_PRICE	The selling price per unit for this transaction.
MEMO_LINE_NAME	MEMO_LINE_NAME	The name of the standard memo line for this transaction.
SALES_ORDER_SOURCE	SALES_ORDER_SOURCE	The source of the sales order for this transaction.
SALES_ORDER	SALES_ORDER	The sales order number for this transaction.
SALES_ORDER_REVISION	SALES_ORDER_REVISION	The sales order revision for this transaction.
SALES_ORDER_LINE	SALES_ORDER_LINE	The sales order line number for this transaction.
SALES_ORDER_DATE	SALES_ORDER_DATE	The date of the sales order for this transaction.
INTERFACE_LINE_CONTEXT	INTERFACE_LINE_CONTEXT	The line transaction flexfield for this transaction. The line transaction flexfield is a combination of attribute values that you use to uniquely identify this transaction line in your original system. The reference value you enter here provides you with an audit trail from receivables back to your original system. You must enter values for enabled attributes
INTERFACE_LINE_ATTRIBUTE1	INTERFACE_LINE_ATTRIBUTE1	
INTERFACE_LINE_ATTRIBUTE2	INTERFACE_LINE_ATTRIBUTE2	
INTERFACE_LINE_ATTRIBUTE3	INTERFACE_LINE_ATTRIBUTE3	
INTERFACE_LINE_ATTRIBUTE4	INTERFACE_LINE_ATTRIBUTE4	
INTERFACE_LINE_ATTRIBUTE5	INTERFACE_LINE_ATTRIBUTE5	
INTERFACE_LINE_ATTRIBUTE6	INTERFACE_LINE_ATTRIBUTE6	

Field Name	Maps to Column	Description
INTERFACE_LINE_ATTRIBUTE7	INTERFACE_LINE_ATTRIBUTE7	
INTERFACE_LINE_ATTRIBUTE8	INTERFACE_LINE_ATTRIBUTE8	
INTERFACE_LINE_ATTRIBUTE9	INTERFACE_LINE_ATTRIBUTE9	
INTERFACE_LINE_ATTRIBUTE10	INTERFACE_LINE_ATTRIBUTE10	
INTERFACE_LINE_ATTRIBUTE11	INTERFACE_LINE_ATTRIBUTE11	
INTERFACE_LINE_ATTRIBUTE12	INTERFACE_LINE_ATTRIBUTE12	
INTERFACE_LINE_ATTRIBUTE13	INTERFACE_LINE_ATTRIBUTE13	
INTERFACE_LINE_ATTRIBUTE14	INTERFACE_LINE_ATTRIBUTE14	
INTERFACE_LINE_ATTRIBUTE15	INTERFACE_LINE_ATTRIBUTE15	
REFERENCE_LINE_CONTEXT	REFERENCE_LINE_CONTEXT	If this transaction is a credit memo, you must enter the transaction flexfield of the transaction line you are crediting in these columns, or the transaction you are crediting.
REFERENCE_LINE_ATTRIBUTE1	REFERENCE_LINE_ATTRIBUTE1	
REFERENCE_LINE_ATTRIBUTE2	REFERENCE_LINE_ATTRIBUTE2	
REFERENCE_LINE_ATTRIBUTE3	REFERENCE_LINE_ATTRIBUTE3	
REFERENCE_LINE_ATTRIBUTE4	REFERENCE_LINE_ATTRIBUTE4	
REFERENCE_LINE_ATTRIBUTE5	REFERENCE_LINE_ATTRIBUTE5	
REFERENCE_LINE_ATTRIBUTE6	REFERENCE_LINE_ATTRIBUTE6	
REFERENCE_LINE_ATTRIBUTE7	REFERENCE_LINE_ATTRIBUTE7	

Field Name	Maps to Column	Description
REFERENCE_LINE_ATTRIBUTE8	REFERENCE_LINE_ATTRIBUTE8	
REFERENCE_LINE_ATTRIBUTE9	REFERENCE_LINE_ATTRIBUTE9	
REFERENCE_LINE_ATTRIBUTE10	REFERENCE_LINE_ATTRIBUTE10	
REFERENCE_LINE_ATTRIBUTE11	REFERENCE_LINE_ATTRIBUTE11	
REFERENCE_LINE_ATTRIBUTE12	REFERENCE_LINE_ATTRIBUTE12	
REFERENCE_LINE_ATTRIBUTE13	REFERENCE_LINE_ATTRIBUTE13	
REFERENCE_LINE_ATTRIBUTE14	REFERENCE_LINE_ATTRIBUTE14	
REFERENCE_LINE_ATTRIBUTE15	REFERENCE_LINE_ATTRIBUTE15	
ATTRIBUTE_CATEGORY	ATTRIBUTE_CATEGORY	Descriptive flexfield attributes allow you to store additional columns, the contents of which you define.
ATTRIBUTE1	ATTRIBUTE1	
ATTRIBUTE2	ATTRIBUTE2	
ATTRIBUTE3	ATTRIBUTE3	
ATTRIBUTE4	ATTRIBUTE4	
ATTRIBUTE5	ATTRIBUTE5	
ATTRIBUTE6	ATTRIBUTE6	
ATTRIBUTE7	ATTRIBUTE7	
ATTRIBUTE8	ATTRIBUTE8	
ATTRIBUTE9	ATTRIBUTE9	
ATTRIBUTE10	ATTRIBUTE10	
ATTRIBUTE11	ATTRIBUTE11	
ATTRIBUTE12	ATTRIBUTE12	
ATTRIBUTE13	ATTRIBUTE13	

Field Name	Maps to Column	Description
ATTRIBUTE14	ATTRIBUTE14	
ATTRIBUTE15	ATTRIBUTE15	

1.3 FREIGHT_LINES (RA_INTERFACE_LINES_ALL)

Field Name	Maps to Column	Description
DESCRIPTION	DESCRIPTION	Required. Provide a description for this transaction.
QUANTITY	QUANTITY	Number of Units shipped
QUANTITY_ORDERED	QUANTITY_ORDERED	The original number of units ordered for this transaction.
UNIT_STANDARD_PRICE	UNIT_STANDARD_PRICE	The standard price per unit for this transaction.
UNIT_SELLING_PRICE	UNIT_SELLING_PRICE	The selling price per unit for this transaction.
AMOUNT	AMOUNT	The revenue amount for this transaction.
ACCOUNTING_RULE_DURATION	ACCOUNTING_RULE_DURATION	If this transaction uses a variable duration accounting rule, you must enter a value in this column.
ACCOUNTING_RULE_NAME	ACCOUNTING_RULE_NAME	The accounting rule name for this transaction.
RULE_START_DATE	RULE_START_DATE	The date that you want to start the accounting rule for this transaction.
LAST_PERIOD_TO_CREDIT	LAST_PERIOD_TO_CREDIT	For unit credit memos, enter the last period number from which you want to start crediting.
MEMO_LINE_NAME	MEMO_LINE_NAME	The name of the standard memo line for this transaction.
SALES_ORDER_SOURCE	SALES_ORDER_SOURCE	The source of the sales order for this transaction.
SALES_ORDER	SALES_ORDER	The sales order number for this transaction.
SALES_ORDER_REVISION	SALES_ORDER_REVISION	The sales order revision for this transaction.
SALES_ORDER_LINE	SALES_ORDER_LINE	The sales order line number for this transaction.
SALES_ORDER_DATE	SALES_ORDER_DATE	The date of the sales order for this transaction.

Field Name	Maps to Column	Description
INTERFACE_LINE_CONTEXT	INTERFACE_LINE_CONTEXT	The line transaction flexfield for this transaction. The line transaction flexfield is a combination of attribute values that you use to uniquely identify this transaction line in your original system. The reference value provides an audit trail from Receivables back to your original system. You must enter values for enabled attributes
INTERFACE_LINE_ATTRIBUTE1	INTERFACE_LINE_ATTRIBUTE1	
INTERFACE_LINE_ATTRIBUTE2	INTERFACE_LINE_ATTRIBUTE2	
INTERFACE_LINE_ATTRIBUTE3	INTERFACE_LINE_ATTRIBUTE3	
INTERFACE_LINE_ATTRIBUTE4	INTERFACE_LINE_ATTRIBUTE4	
INTERFACE_LINE_ATTRIBUTE5	INTERFACE_LINE_ATTRIBUTE5	
INTERFACE_LINE_ATTRIBUTE6	INTERFACE_LINE_ATTRIBUTE6	
INTERFACE_LINE_ATTRIBUTE7	INTERFACE_LINE_ATTRIBUTE7	
INTERFACE_LINE_ATTRIBUTE8	INTERFACE_LINE_ATTRIBUTE8	
INTERFACE_LINE_ATTRIBUTE9	INTERFACE_LINE_ATTRIBUTE9	
INTERFACE_LINE_ATTRIBUTE10	INTERFACE_LINE_ATTRIBUTE10	
INTERFACE_LINE_ATTRIBUTE11	INTERFACE_LINE_ATTRIBUTE11	
INTERFACE_LINE_ATTRIBUTE12	INTERFACE_LINE_ATTRIBUTE12	
INTERFACE_LINE_ATTRIBUTE13	INTERFACE_LINE_ATTRIBUTE13	
INTERFACE_LINE_ATTRIBUTE14	INTERFACE_LINE_ATTRIBUTE14	

Field Name	Maps to Column	Description
INTERFACE_LINE_ATTRIBUTE15	INTERFACE_LINE_ATTRIBUTE15	
REFERENCE_LINE_CONTEXT	REFERENCE_LINE_CONTEXT	If this transaction is a credit memo, you must enter the transaction flexfield of the transaction line you are crediting in these columns of the transaction you are crediting.
REFERENCE_LINE_ATTRIBUTE1	REFERENCE_LINE_ATTRIBUTE1	
REFERENCE_LINE_ATTRIBUTE2	REFERENCE_LINE_ATTRIBUTE2	
REFERENCE_LINE_ATTRIBUTE3	REFERENCE_LINE_ATTRIBUTE3	
REFERENCE_LINE_ATTRIBUTE4	REFERENCE_LINE_ATTRIBUTE4	
REFERENCE_LINE_ATTRIBUTE5	REFERENCE_LINE_ATTRIBUTE5	
REFERENCE_LINE_ATTRIBUTE6	REFERENCE_LINE_ATTRIBUTE6	
REFERENCE_LINE_ATTRIBUTE7	REFERENCE_LINE_ATTRIBUTE7	
REFERENCE_LINE_ATTRIBUTE8	REFERENCE_LINE_ATTRIBUTE8	
REFERENCE_LINE_ATTRIBUTE9	REFERENCE_LINE_ATTRIBUTE9	
REFERENCE_LINE_ATTRIBUTE10	REFERENCE_LINE_ATTRIBUTE10	
REFERENCE_LINE_ATTRIBUTE11	REFERENCE_LINE_ATTRIBUTE11	
REFERENCE_LINE_ATTRIBUTE12	REFERENCE_LINE_ATTRIBUTE12	
REFERENCE_LINE_ATTRIBUTE13	REFERENCE_LINE_ATTRIBUTE13	
REFERENCE_LINE_ATTRIBUTE14	REFERENCE_LINE_ATTRIBUTE14	
REFERENCE_LINE_ATTRIBUTE15	REFERENCE_LINE_ATTRIBUTE15	

Field Name	Maps to Column	Description
ATTRIBUTE_CATEGORY	ATTRIBUTE_CATEGORY	Descriptive flexfield attributes allow you to store additional columns, the contents of which you define.
ATTRIBUTE1	ATTRIBUTE1	
ATTRIBUTE2	ATTRIBUTE2	
ATTRIBUTE3	ATTRIBUTE3	
ATTRIBUTE4	ATTRIBUTE4	
ATTRIBUTE5	ATTRIBUTE5	
ATTRIBUTE6	ATTRIBUTE6	
ATTRIBUTE7	ATTRIBUTE7	
ATTRIBUTE8	ATTRIBUTE8	
ATTRIBUTE9	ATTRIBUTE9	
ATTRIBUTE10	ATTRIBUTE10	
ATTRIBUTE11	ATTRIBUTE11	
ATTRIBUTE12	ATTRIBUTE12	
ATTRIBUTE13	ATTRIBUTE13	
ATTRIBUTE14	ATTRIBUTE14	
ATTRIBUTE15	ATTRIBUTE15	

Receive Bank Statement Service

The name of this service is:

WmOAFIN107SC.cashManagement107SC.intoOA.bankStatement:receiveBankStatement

This service loads bank statements into the Cash Management module of Oracle Applications and reconciles the bank statements automatically. The Auto Reconciliation program submits a request to reconcile the bank statement when all import errors are resolved.

Other characteristics include:

- All transactions in the batch must use the same currency.
- You can process multiple statements for the same bank account in a batch.

Database Scripts

This service uses the following database scripts:

Database Script	Description
wm_install_into_bankstmt.sql	Runs all the scripts listed below, except the uninstall script.
wm_into_bankstmt_pkg.sql	Installs WM_CE_BANK_IMP_HANDLER_PKG. WM_HANDLE_CEBANK, which calls the WM_CONC_REQUEST.WM_REQUEST_SUBMIT procedure to submit the bank statement process.
wm_drop_into_bankstmt.sql	Uninstalls all components created by wm_install_into_bankstmt.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definitions:

- setBankStatementTxn107SC.txp
- BankStatementTransactions107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow executes as follows:

- **specifyDefaultSettings** specifies the default parameter settings required for service execution. You should change these settings accordingly.
- **specifyConcProgParams** specifies the default parameter settings required for concurrent program execution. You should change these settings accordingly.
- **bizDocMapping** maps the incoming business document structure to the required Oracle Applications data structures (interface tables). It loops over the parent and the child to extract data to flat format.
 - **convertToDateObject** converts a date string to the date object.
 - **getOrgId** takes ORGANIZATION_NAME as the input parameter and queries the table ORG_ORGANIZATION_DEFINITIONS and gets the ORGANIZATION_ID corresponding to the ORGANIZATION_NAME.

- **setBankStatementTxn** inserts data into the interface table. It extracts data from the Idata structure and loads the data into CE_STATEMENT_HEADERS_INT_ALL, CE_STATEMENT_LINES_INTERFACE interface tables.
- **ImportBankStatement** imports bank statements to the production table from the interface table. To monitor the import process, it sequentially invokes the execBankStatementConcProg, checkBankStatementImportStatus, getBankStatementImport_ERR services. The DB Errors and Concurrent Program execution results are stored in the dbErrorMsg RecordsSet and concProgMsg RecordSet. The errorsDoc record list captures data- related errors.
 - **execBankStatementConcProg** invokes custom package and stored procedure WM_CE_BANK_IMP_HANDLER_PKG.WM_HANDLE_CEBANK to execute the concurrent program.
 - **checkBankStatementImportStatus**, this service checks the CE_STATEMENT_HEADERS_INT_ALL table after the data import execution to find errors related to data import for the corresponding Bank Account for the Bank Statement(s).
 - **getBankStatementImport_ERR** gets the data import error for the specific Bank Account. This service queries the following table CE_HEADER_INTERFACE_ERRORS and CE_LINE_INTERFACE_ERRORS and gets relevant error messages for the header and the lines.

For more details on receive transactions, see [“Overview of Receive Service Transaction Processing” on page 31](#).

Business Document Structure

A bank statement business document is one or more statements from one bank account. This services uses the following business document structure:

- 1.0 STATEMENT
 - 1.1 STATEMENT_LINES

1.0 STATEMENT (Maps to CE_STATEMENT_HEADERS_INT_ALL)

Field Name	Maps to Column	Description
STATEMENT_NUMBER	STATEMENT_NUMBER	Required. Bank statement number.
STATEMENT_DATE	STATEMENT_DATE	Required. Statement date.
CONTROL_BEGIN_BALANCE	CONTROL_BEGIN_BALANCE	Control beginning balance.
CONTROL_END_BALANCE	CONTROL_END_BALANCE	Control end balance.

Field Name	Maps to Column	Description
CONTROL_TOTAL_DR	CONTROL_TOTAL_DR	Total receipt amount of the bank statement that can be compared to the actual receipt Entry totals for control purposes.
CONTROL_TOTAL_CR	CONTROL_TOTAL_CR	Total payment amount of the bank statement that can be Compared to the actual payment entry totals for control.
CONTROL_DR_LINE_COUNT	CONTROL_DR_LINE_COUNT	Total receipt line count of the statement that can be compared to the actual number of receipts entered for control purposes.
CONTROL_CR_LINE_COUNT	CONTROL_CR_LINE_COUNT	Total payment line count of the statement that can be compared to the actual number of payments entered for control purposes.
CONTROL_LINE_COUNT	CONTROL_LINE_COUNT	Total line count of the statement that can be compared to the actual number of lines entered for control purposes.
RECORD_STATUS_FLAG	RECORD_STATUS_FLAG	Statement upload status, lookup type HEADER_INTERFACE_STATUS. Use: C: for Corrected E: for Error N: for New T for Transferred
CURRENCY_CODE	CURRENCY_CODE	Bank statement currency code.
ORG_NAME	ORG_ID	Derived from organization definitions.

1.1 STATEMENT_LINES (Maps to CE_STATEMENT_LINES_INTERFACE)

Field Name	Maps to Column	Description
LINE_NUMBER	LINE_NUMBER	Required. Line number.
TRX_DATE	TRX_DATE	Required. Line transaction date, used to update AP check's cleared date and AR receipt's cleared date if EFFECTIVE_DATE is null.
TRX_CODE	TRX_CODE	Bank transaction code.
EFFECTIVE_DATE	EFFECTIVE_DATE	Statement line effective date.

Field Name	Maps to Column	Description
TRX_TEXT	TRX_TEXT	Statement line description.
INVOICE_TEXT	INVOICE_TEXT	Invoice number for finding available receipts by invoice.
AMOUNT	AMOUNT	Statement line amount.
CURRENCY_CODE	CURRENCY_CODE	Statement line currency code.
EXCHANGE_RATE	EXCHANGE_RATE	Exchange rate.
USER_EXCHANGE_RATE_TYPE	USER_EXCHANGE_RATE_TYPE	Currency conversion rate type.
BANK_TRX_NUMBER	BANK_TRX_NUMBER	Transaction number that identifies the transaction to be matched against the statement line.
CUSTOMER_TEXT	CUSTOMER_TEXT	Customer note.
EXCHANGE_RATE_DATE	EXCHANGE_RATE_DATE	Exchange rate date.
ORIGINAL_AMOUNT	ORIGINAL_AMOUNT	Statement line amount in currency code.
CHARGES_AMOUNT	CHARGES_AMOUNT	Bank charges amount.

Receive Customer Service

The name of this service is:

WmOAFIN107SC.receivables107SC.intoOA.customer:receiveCustomer

This service monitors the import process. Customer Import Error Handling

The Customer Interface process deletes the rows from the interface tables which were imported successfully or contained warning messages. The error records in the interface table have an error interface status code. To identify the error records for a particular Customer Interface import run, query the interface table with the actual concurrent request ID. The import process changes the request ID to null for those records not deleted from the interface table after an unsuccessful import.

The Customer Interface process does not link the import process and the records in the interface table that it processes. In addition, no interface error table exists in Oracle Applications to store the Customer Import errors. Instead, the Customer Interface Transfer Report online report generates and includes exceptions from interface records for a particular run. To obtain the Customer interface status, check the Customer Interface Transfer Report generated along with the concurrent request.

Database Scripts

This service uses the following database scripts:

Database Script	Description
wm_install_into_customer.sql	Runs any scripts for the service.
wm_into_customer_pkg.sql	Installs WM_CUSTOMER_IMP_HANDLER_PKG, WM_HANDLE_CUSTOMER, which calls the WM_CONC_REQUEST.WM_REQUEST_SUBMIT procedure to submit the customer concurrent process.
wm_drop_into_customer.sql	Uninstalls all components created by wm_install_into_customer.sql.

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definitions:

- setCustomerTxn107SC.txp
- customerTransactions107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow executes as follows:

- **specifyDefaultSettings** specifies the default parameter settings required for service execution. You should change these settings accordingly.
- **specifyConcProgParams** specifies the default parameter settings required for concurrent program execution. You should change these settings accordingly.
- **bizDocMapping** maps the incoming business document structure to the required Oracle Applications data structures (interface tables).
- **setCustomerTxn** inserts data into the interface table. It extracts data from the IData structure resulted in the bizDocMapping service and puts the data into the interface table in Oracle Applications for Customer.
- **importCustomer** imports data to the production table from the interface table. It calls the execCustomerConcProg service to execute the corresponding concurrent program that inserts data into the production table and to generate the error/acknowledgement message.

- `execCustomerConcProg` invokes the stored procedure `WM_CUSTOMER_IMP_HANDLER_PKG.WM_HANDLE_CUSTOMER`. The procedure calls the corresponding concurrent subroutine to execute the data import process for the Customer into Oracle Applications. This service produces Status ID, Request ID, Execution Status Message for normal concurrent program completion, and any database Stored Procedure error message if an exception occurs in the Stored Procedure execution).

For more details on receive transactions, see [“Overview of Receive Service Transaction Processing” on page 31](#).

Business Document Structure

This service uses the business document `CustomerBizDoc`. Its structure is as follows:

- 1.0 CUSTOMERS
 - 1.1. SITES
 - 1.1.1 SITE_USES
 - 1.1.1.1 SITE_PAYMENT_METHODS
 - 1.1.1.2 **SITE_BANK_ACCOUNTS
 - 1.1.1.3 **SITE_PROFILES
 - 1.1.1.3.1 SITE_PROFILE_AMOUNTS
 - 1.1.2 SITE_PHONES
 - 1.1.3 SITE_CONTACTS
 - 1.1.3.1** SITE_CONTACT_PHONES
 - 1.2. **PHONES
 - 1.3 **CONTACTS
 - 1.3.1 **CONTACT_PHONES
 - 1.4 **BANK_ACCOUNTS
 - 1.5**PAYMENT_METHODS
 - 1.6 **CUSTOMER_PROFILES
 - 1.6.1 **PROFILE_AMOUNTS

**Some services use the same field names and column mappings as follows:

For These Documents...	Use the following table:
<ul style="list-style-type: none"> ■ 1.1.3.1 SITE_CONTACT_PHONES ■ 1.2 PHONES ■ 1.3.1 CONTACT_PHONES 	“1.1.2 SITE_PHONES (Maps to (Maps to RA_CONTACT_PHONES_INT_ALL)” on page 113.
<ul style="list-style-type: none"> ■ 1.3 CONTACTS 	“1.1.3 SITE_CONTACTS (Maps to RA_CONTACT_PHONES_INT_ALL)” on page 115.
<ul style="list-style-type: none"> ■ 1.4 BANK_ACCOUNTS 	“1.1.1.2 SITE_BANK_ACCOUNTS (Maps to RA_CUSTOMER_BANKS_INT_ALL)” on page 109.
<ul style="list-style-type: none"> ■ 1.5 PAYMENT_METHODS 	“1.1.1.1 SITE_PAYMENT_METHODS (Maps to RA_CUST_PAY_METHOD_INT_ALL)” on page 109.
<ul style="list-style-type: none"> ■ 1.6 CUSTOMER_PROFILES 	“1.1.1.3 SITE_PROFILES (Maps to RA_CUSTOMER_PROFILES_INT_ALL)” on page 111.
<ul style="list-style-type: none"> ■ 1.6.1 PROFILE_AMOUNTS 	“1.1.1.3.1 SITE_PROFILE_AMOUNTS (Maps to RA_CUSTOMER_PROFILES_INT_ALL)” on page 113.

1.0 CUSTOMERS (Maps to RA_CUSTOMERS_INTERFACE_ALL)

Field Name	Maps to Column	Description
CUSTOMER_NAME	CUSTOMER_NAME	Required. Name of the customer.
CUSTOMER_NUMBER	CUSTOMER_NUMBER	Customer number
CUSTOMER_KEY	CUSTOMER_KEY	Derived key created by Oracle sales and marketing to facilitate querying
CUSTOMER_STATUS	CUSTOMER_STATUS	Customer status flag.
ORIG_SYSTEM_REFERENCE	ORIG_SYSTEM_CUSTOMER_REFERENCE	Required Unique customer identifier from foreign system
CUSTOMER_CATEGORY_CODE	CUSTOMER_CATEGORY_CODE	User-definable category.
CUSTOMER_CLASS_CODE	CUSTOMER_CLASS_CODE	Customer class identifier
CUSTOMER_TYPE	CUSTOMER_TYPE	Receivables lookup code for CUSTOMER_TYPE. Use: I for Internal customers R for External customers
TAX_REFERENCE	TAX_REFERENCE	Taxpayer identification number.
TAX_CODE	TAX_CODE	Tax code for this customer.

Field Name	Maps to Column	Description
SHIP_VIA	CUST_SHIP_VIA_CODE	Name of shipping firm.
TAXPAYER_ID	JGZZ_FISCAL_CODE	Fiscal code for some European countries.
LANGUAGE	LANGUAGE	Language.
INSERT_UPDATE_FLAG	INSERT_UPDATE_FLAG	Required. Flag to indicate whether the Customer record is being inserted, or an existing record is being updated
ORGANIZATION_NAME	ORG_ID	Organization name.

1.1 SITES (Maps RA_CUSTOMERS_INTERFACE_ALL)

Field Name	Maps to Column	Description
ORIG_SYSTEM_REFERENCE	ORIG_SYSTEM_ADDRESS_REF	Address identifier from foreign system.
ADDRESS1	ADDRESS1	First line for address.
ADDRESS2	ADDRESS2	Second line for address.
ADDRESS3	ADDRESS3	Third line for address.
ADDRESS4	ADDRESS4	Fourth line for address.
CITY	CITY	City.
COUNTY	COUNTY	COUNTY.
STATE	STATE	STATE.
PROVINCE	PROVINCE	PROVINCE.
COUNTRY	COUNTRY	COUNTRY.
POSTAL_CODE	POSTAL_CODE	POSTAL_CODE.
ORGANIZATION_NAME	ORG_ID	Organization name.

1.1.1 SITE_USES (Maps to RA_CUSTOMERS_INTERFACE_ALL)

Field Name	Maps to Column	Description
SITE_USE_ID		
SITE_USE_CODE	SITE_USE_CODE	Business purpose.
PRIMARY_FLAG	PRIMARY_SITE_USE_FLAG	Y or N flag indicates whether site is primary.

Field Name	Maps to Column	Description
BILL_TO_LOCATION	BILL_TO_ORIG_ADDRESS_REF	Site use identifier.
SHIP_VIA	SITE_SHIP_VIA_CODE	Name of the preferred shipping Company.
TAX_REFERENCE	SITE_USE_TAX_REFERENCE	TAX_REFERENCE.
TAX_CODE	SITE_USE_TAX_CODE	TAX_CODE.
DEMAND_CLASS_CODE	DEMAND_CLASS_CODE	DEMAND_CLASS_CODE.
LOCATION	LOCATION	Location.

1.1.1.1 SITE_PAYMENT_METHODS (Maps to RA_CUST_PAY_METHOD_INT_ALL)

Field Name	Maps to Column	Description
PRIMARY_FLAG	PRIMARY_FLAG	Required. Indicates whether customer receipt method is primary (Yes) or not (No)..
RECEIPT_METHOD_NAME	RECEIPT_METHOD_NAME	Required. Receipt method name.
START_DATE	START_DATE	Required. Start date of the customer receipt methods.
END_DATE	END_DATE	End date of the customer receipt methods.

1.1.1.2 SITE_BANK_ACCOUNTS (Maps to RA_CUSTOMER_BANKS_INT_ALL)

Field Name	Maps to Column	Description
START_DATE	START_DATE	Required. Start date that the bank account uses.
END_DATE	END_DATE	Required. End Date that the Bank Account uses.
PRIMARY_FLAG	PRIMARY_FLAG	Required. Indicates whether the bank account use is primary (Yes) or not (No).
BANK_NUMBER	BANK_NUMBER	Bank number.
BANK_NAME	BANK_NAME	Bank name.
BANK_BRANCH_NAME	BANK_BRANCH_NAME	Bank branch name.
BANK_BRANCH_NUM	BANK_NUM	Branch number.

Field Name	Maps to Column	Description
BANK_BRANCH_DESCRIPTION	BANK_BRANCH_DESCRIPTION	Description.
BANK_ACCOUNT_NAME	BANK_ACCOUNT_NAME	Bank account name.
BANK_ACCOUNT_NUM	BANK_ACCOUNT_NUM	Bank account number.
BANK_ACCOUNT_DESCRIPTION	BANK_ACCOUNT_DESCRIPTION	Description.
BANK_CURRENCY_CODE	BANK_CURRENCY_CODE	Currency code.
BANK_ACCOUNT_INACTIVE_DATE	BANK_ACCOUNT_INACTIVE_DATE	Inactive date.
BANK_BRANCH_ADDRESS1	BANK_BRANCH_ADDRESS1	First address line.
BANK_BRANCH_ADDRESS2	BANK_BRANCH_ADDRESS2	Second address line.
BANK_BRANCH_ADDRESS3	BANK_BRANCH_ADDRESS3	Third address line.
BANK_BRANCH_ADDRESS4	BANK_BRANCH_ADDRESS4	Fourth address line.
BANK_BRANCH_CITY	BANK_BRANCH_CITY	CITY.
BANK_BRANCH_STATE	BANK_BRANCH_STATE	STATE.
BANK_BRANCH_ZIP	BANK_BRANCH_ZIP	ZIP.
BANK_BRANCH_PROVINCE	BANK_BRANCH_PROVINCE	PROVINCE.
BANK_BRANCH_COUNTRY	BANK_BRANCH_COUNTRY	COUNTRY.
BANK_BRANCH_AREA_CODE	BANK_BRANCH_AREA_CODE	AREA_CODE.
BANK_BRANCH_PHONE	BANK_BRANCH_PHONE	PHONE.
BANK_BRANCH_COUNTY	BANK_BRANCH_COUNTY	COUNTY.
BANK_BRANCH_EFT_USER_NUMBER	BANK_BRANCH_EFT_USER_NUMBER	The number that identifies you as a user of electronic funds transfer services to your bank or clearing organization.
BANK_ACCOUNT_CHECK_DIGITS	BANK_ACCOUNT_CHECK_DIGITS	Holds any check digits that result from bank account number validation in FBS.

1.1.1.3 SITE_PROFILES (Maps to RA_CUSTOMER_PROFILES_INT_ALL)

Field Name	Maps to Column	Description
PROFILE_CLASS_NAME	CUSTOMER_PROFILE_CLASS_NAME	Profile class name.
PROFILE_CLASS_DESCRIPTION		Profile class description.
COLLECTOR_ID		Collector identifier.
COLLECTOR_NAME	COLLECTOR_NAME	Name of collector.
CREDIT_CHECKING	CREDIT_CHECKING	Indicates whether a credit check is to be carried out.
TOLERANCE	TOLERANCE	Percentage over credit limit that this customer can exceed before action is taken.
DISCOUNT_TERMS	DISCOUNT_TERMS	Indicates whether to allow discount terms. Default value is Y.
DUNNING_LETTERS	DUNNING_LETTERS	Indicates whether to send dunning letters to this customer when invoices, debit memos or charge backs become past due.
INTEREST_CHARGES	INTEREST_CHARGES	Indicates whether to charge this customer interest.
STATEMENTS	STATEMENTS	Indicates whether to send this customer statement.
CREDIT_BALANCE_STATEMENTS	CREDIT_BALANCE_STATEMENTS	Indicates whether to send statements that have a credit balance.
CREDIT_HOLD	CREDIT_HOLD	Required. Indicates whether to put a hold on this customer's credit.
CREDIT_RATING	CREDIT_RATING	Credit rating.
CREDIT_RATING_MEANING	CREDIT_RATING_MEANING	Credit rating meaning.
RISK_CODE	RISK_CODE	Risk code.
RISK_MEANING	RISK_CODE_MEANING	Risk code meaning.
STANDARD_TERMS	STANDARD_TERMS	User-defined payment.
OVERRIDE_TERMS	OVERRIDE_TERMS	Indicates whether to allow override of standard.

Field Name	Maps to Column	Description
DUNNING_LETTER_SET_NAME	DUNNING_LETTER_SET_NAME	Dunning letter set name.
INTEREST_PERIOD_DAYS	INTEREST_PERIOD_DAYS	Number of days to which the interest rate refers.
PAYMENT_GRACE_DAYS	PAYMENT_GRACE_DAYS	Maximum number of overdue days allowed before action.
DISCOUNT_GRACE_DAYS	DISCOUNT_GRACE_DAYS	Number of days after discount term date during which this customer can take discounts.
STATEMENT_CYCLE_NAME	STATEMENT_CYCLE_NAME	Statement cycle name.
ACCOUNT_STATUS	ACCOUNT_STATUS	User defined account status.
ACCOUNT_STATUS_MEANING		
AUTOCASH_HIERARCHY_NAME	AUTOCASH_HIERARCHY_NAME	Autocash hierarchy name.
AUTO_REC_INCL_DISPUTED_FLAG	AUTO_REC_INCL_DISPUTED_FLAG	Indicates whether to include disputed transactions.
TAX_PRINTING_OPTION	TAX_PRINTING_OPTION	Tax printing option.
CHARGE_ON_FINANCE_CHARGE_FLAG	CHARGE_ON_FINANCE_CHARGE_FLAG	Indicates whether to compound finance charges for this customer or site.
GROUPING_RULE_NAME	GROUPING_RULE_NAME	Grouping rule name.
PERCENT_COLLECTABLE	PERCENT_COLLECTABLE	Percentage of the customer's account balance that you expect to collect on a regular basis.
CLEARING_DAYS	CLEARING_DAYS	Number of clearing days before receipts that belong to this customer or site. This overrides the value of the payment method / bank account.
INSERT_UPDATE_FLAG	INSERT_UPDATE_FLAG	Required. Flag to indicate an Insert or Update of the Profile record.

1.1.1.3.1 SITE_PROFILE_AMOUNTS (Maps to RA_CUSTOMER_PROFILES_INT_ALL)

Field Name	Maps to Column	Description
CURRENCY_CODE	CURRENCY_CODE	Required. Code defined for a currency.
TRX_CREDIT_LIMIT	TRX_CREDIT_LIMIT	Required. Credit limit for an order.
OVERALL_CREDIT_LIMIT	OVERALL_CREDIT_LIMIT	Required. Overall credit limit.
MIN_DUNNING_AMOUNT	MIN_DUNNING_AMOUNT	Minimum total a dunning letter should have before the letter is printed.
MIN_DUNNING_INVOICE_AMOUNT	MIN_DUNNING_INVOICE_AMOUNT	The balance due on a payment schedule must be at least this value before it can be dunned.
MAX_INTEREST_CHARGE	MAX_INTEREST_CHARGE	Maximum interest to be charged per invoice for a currency.
MIN_STATEMENT_AMOUNT	MIN_STATEMENT_AMOUNT	Minimum total a statement should have before the statement is printed.
AUTO_REC_MIN_RECEIPT_AMOUNT	AUTO_REC_MIN_RECEIPT_AMOUNT	Limits the minimum receipt amount for a currency.
INTEREST_RATE	INTEREST_RATE	The interest rate to be charged to this customer account or site for invoices in this currency.
MIN_FC_BALANCE_AMOUNT	MIN_FC_BALANCE_AMOUNT	Minimum balance that a customer account or customer account site should have before any finance charges can be charged to invoices.
MIN_FC_INVOICE_AMOUNT	MIN_FC_INVOICE_AMOUNT	Minimum balance on an invoice before any finance charges can be computed for it.
INSERT_UPDATE_FLAG	INSERT_UPDATE_FLAG	Required. Flag to indicate Insert or Update of the Profile amount record.

1.1.2 SITE_PHONES (Maps to (Maps to RA_CONTACT_PHONES_INT_ALL)

Field Name	Maps to Column	Description
PHONE_NUMBER	TELEPHONE	Required. Telephone number formatted in the local format.
STATUS		Active or inactive status.

Field Name	Maps to Column	Description
PHONE_TYPE	TELEPHONE_TYPE	Required. Lookup code for the type of phone line, for example, general, fax, inbound, outbound.
AREA_CODE	TELEPHONE_AREA_CODE	Area code within a country code.
EXTENSION	TELEPHONE_EXTENSION	Additional number used by an internal telephone system after the internal telephone system is contacted.
PRIMARY_FLAG		A value of Y indicates this is the primary contact point of this contact point type for referenced party, site, or location.
ORIG_SYSTEM_REFERENCE	ORIG_SYSTEM_TELEPHONE_REF	Required. Identifier for this record from foreign system.
INSERT_UPDATE_FLAG	INSERT_UPDATE_FLAG	Required. Flag to indicate insert or update of the phone record.

1.1.3 SITE_CONTACTS (Maps to RA_CONTACT_PHONES_INT_ALL)

Field Name	Maps to Column	Description
TITLE	CONTACT_TITLE	Title.
FIRST_NAME	CONTACT_FIRST_NAME	Person first name.
LAST_NAME	CONTACT_LAST_NAME	Required. Person last name.
JOB_TITLE	CONTACT_JOB_TITLE	Job title.
MAIL_STOP	MAIL_STOP	Mail stop.
ORIG_SYSTEM_REFERENCE	ORIG_SYSTEM_CONTACT_REF	Required. Reference to identify foreign system.
CONTACT_KEY	CONTACT_KEY	Contact key.
EMAIL_ADDRESS	EMAIL_ADDRESS	Email address
INSERT_UPDATE_FLAG	INSERT_UPDATE_FLAG	Required. Flag to indicate insert or update of the contact record.

Receive FA Budget Service

The name of this service is:

WmOAFIN107SC.fixedAssets107SC.intoOA.FABudget:receiveFABudget

This service monitors the FA Budget insert process. It executes flow services internally to map the incoming document to the interface table, and inserts data into the interface table.

The Integration Server flow does not execute the concurrent program for the actual upload of Budgets into Oracle Assets production tables because:

- You define the Concurrent Program for uploading budgets in Oracle Assets, but the standard configuration of the concurrent program does not allow the submission from SRS (Standard Request Submission). Instead, you use a separate Upload Capital Budget (that is, Assets: Budgets to Upload) window where you specify the book and initiate the upload.
- Uploading Budgets for a book requires you to delete that existing budget for the same book from Oracle Assets. You must use the Upload Capital Budget window to do this. This window gives the option to delete any existing Budget and upload a new one.

Database Scripts

This service does not require any database scripts.

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definition:

- `setFABudgetTxn107SC.txp`

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow executes as follows:

- `specifyDefaultSettings` specifies the default parameter settings required for service execution. You should change these settings accordingly.
- `bizDocMapping` maps the incoming business document structure to the required Oracle Applications data structures (interface tables).
- `setFABudgetTxn` inserts data into the interface table.

For more details on receive transactions, see [“Overview of Receive Service Transaction Processing” on page 31](#).

Business Document Structure

This service uses the business document `FABudgetBizDoc`. Its structure is as follows:

- `FA_BUDGET`

FA_BUDGET (Maps to FA_BUDGET_INTERFACE)

Field Name	Maps to Column	Description
BOOK_TYPE_CODE	BOOK_TYPE_CODE	Name of the Fixed Asset Book.
PERIOD1_AMOUNT	PERIOD1_AMOUNT	The budget tax amount you allocate to an asset category and cost center for one period in the fiscal year. You can allocate a budget book annual budget up to 12 periods.
PERIOD2_AMOUNT	PERIOD2_AMOUNT	See PERIOD1_AMOUNT.
PERIOD3_AMOUNT	PERIOD3_AMOUNT	See PERIOD1_AMOUNT.
PERIOD4_AMOUNT	PERIOD4_AMOUNT	See PERIOD1_AMOUNT.
PERIOD5_AMOUNT	PERIOD5_AMOUNT	See PERIOD1_AMOUNT.
PERIOD6_AMOUNT	PERIOD6_AMOUNT	See PERIOD1_AMOUNT.

Field Name	Maps to Column	Description
PERIOD7_AMOUNT	PERIOD7_AMOUNT	See PERIOD1_AMOUNT.
PERIOD8_AMOUNT	PERIOD8_AMOUNT	See PERIOD1_AMOUNT.
PERIOD9_AMOUNT	PERIOD9_AMOUNT	See PERIOD1_AMOUNT.
PERIOD10_AMOUNT	PERIOD10_AMOUNT	See PERIOD1_AMOUNT.
PERIOD11_AMOUNT	PERIOD11_AMOUNT	See PERIOD1_AMOUNT.
PERIOD12_AMOUNT	PERIOD12_AMOUNT	See PERIOD1_AMOUNT.
ACCT_SEGMENT1	ACCT_SEGMENT1	Segments of the Accounting flexfield structure.. The number of segments filled depends on the Accounting flexfield structure for the Set of Books
ACCT_SEGMENT2	ACCT_SEGMENT2	See ACCT_SEGMENT1.
ACCT_SEGMENT3	ACCT_SEGMENT3	See ACCT_SEGMENT1.
ACCT_SEGMENT4	ACCT_SEGMENT4	See ACCT_SEGMENT1.
ACCT_SEGMENT5	ACCT_SEGMENT5	See ACCT_SEGMENT1.
ACCT_SEGMENT6	ACCT_SEGMENT6	See ACCT_SEGMENT1.
ACCT_SEGMENT7	ACCT_SEGMENT7	See ACCT_SEGMENT1.
ACCT_SEGMENT8	ACCT_SEGMENT8	See ACCT_SEGMENT1.
ACCT_SEGMENT9	ACCT_SEGMENT9	See ACCT_SEGMENT1.
ACCT_SEGMENT10	ACCT_SEGMENT10	See ACCT_SEGMENT1.
ACCT_SEGMENT11	ACCT_SEGMENT11	See ACCT_SEGMENT1.
ACCT_SEGMENT12	ACCT_SEGMENT12	See ACCT_SEGMENT1.
ACCT_SEGMENT13	ACCT_SEGMENT13	See ACCT_SEGMENT1.
ACCT_SEGMENT14	ACCT_SEGMENT14	See ACCT_SEGMENT1.
ACCT_SEGMENT15	ACCT_SEGMENT15	See ACCT_SEGMENT1.
ACCT_SEGMENT16	ACCT_SEGMENT16	See ACCT_SEGMENT1.
ACCT_SEGMENT17	ACCT_SEGMENT17	See ACCT_SEGMENT1.
ACCT_SEGMENT18	ACCT_SEGMENT18	See ACCT_SEGMENT1.
ACCT_SEGMENT19	ACCT_SEGMENT19	See ACCT_SEGMENT1.
ACCT_SEGMENT20	ACCT_SEGMENT20	See ACCT_SEGMENT1.
ACCT_SEGMENT21	ACCT_SEGMENT21	See ACCT_SEGMENT1.
ACCT_SEGMENT22	ACCT_SEGMENT22	See ACCT_SEGMENT1.
ACCT_SEGMENT23	ACCT_SEGMENT23	See ACCT_SEGMENT1.

Field Name	Maps to Column	Description
ACCT_SEGMENT24	ACCT_SEGMENT24	See ACCT_SEGMENT1.
ACCT_SEGMENT25	ACCT_SEGMENT25	See ACCT_SEGMENT1.
ACCT_SEGMENT26	ACCT_SEGMENT26	See ACCT_SEGMENT1.
ACCT_SEGMENT27	ACCT_SEGMENT27	See ACCT_SEGMENT1.
ACCT_SEGMENT28	ACCT_SEGMENT28	See ACCT_SEGMENT1.
ACCT_SEGMENT29	ACCT_SEGMENT29	See ACCT_SEGMENT1.
ACCT_SEGMENT30	ACCT_SEGMENT30	See ACCT_SEGMENT1.
CAT_SEGMENT1	CAT_SEGMENT1	Segments of the category flexfield structure. The number of segments entered depends on the category flexfield.
CAT_SEGMENT2	CAT_SEGMENT2	See CAT_SEGMENT1.
CAT_SEGMENT3	CAT_SEGMENT3	See CAT_SEGMENT1.
CAT_SEGMENT4	CAT_SEGMENT4	See CAT_SEGMENT1.
CAT_SEGMENT5	CAT_SEGMENT5	See CAT_SEGMENT1.
CAT_SEGMENT6	CAT_SEGMENT6	See CAT_SEGMENT1.
CAT_SEGMENT7	CAT_SEGMENT7	See CAT_SEGMENT1.

Receive General Ledger Budget Service

The name of this service is:

WmOAFIN107SC.generalLedger107SC.intoOA.GLBudget:receiveGLBudget

This service monitors the FA Budget insert process.

The Integration Server flow does not execute the concurrent program for actual upload of Budgets into Oracle Assets production tables because:

- The General Ledger defines the Concurrent Program for uploading Budgets, but the standard configuration of the concurrent program does not allow the submission from SRS (Standard Request Submission). Instead, you use a separate Upload Budgets (General Ledger: Budgets ->Enter ->Upload).window to specify and update the Budget and the Budget Organization.
- The Concurrent Program definition in Oracle Applications General Ledger does not specify the required parameters. The General Ledger User Guide also does not provide information on the parameters for the Concurrent Program.

- Refer to Metalink Note: 17794.1 titled *Troubleshooting the Budget Upload Process*, which provides the parameters required while executing the Concurrent Program Executable from command line. Using this information, you can execute the Concurrent Program from the flow. However, because the errors encountered during upload do not log into any Interface Error Table, the flow will be unable to report the errors. When the upload initiates from the Upload Budgets window in General Ledger, it creates an error report online. This is the only means to detect the errors during upload.
- Budget Upload can be password-protected. Executing Concurrent Program from the flow will override or ignore this functionality.

Database Scripts

This service does not require any database scripts.

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definitions:

- setGLBudgetTxn107SC.txp
- GLBudgetTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow executes as follows:

- **specifyDefaultSettings** specifies the default parameter settings required for service execution. You should change these settings accordingly.
- **bizDocMapping** maps the incoming business document structure to the required Oracle Applications data structures (interface tables). It loops over the parent and the child to extract data to flat format.
 - **getSOBVersionEntityId** maps the SET_OF_BOOKS_ID, BUDGET_VERSION_ID and BUDGET_ENTITY_ID based on P_BUDGET_NAME and P_BUDGET_ORGANIZATION passed to it. This service is in bizDocMapping.
 - **getGLCodeCombinationId** maps the CODE_COMBINATION_ID based on the GL Accounting Segments and SET_OF_BOOKS_NAME.
- **setGLBudgetTxn** inserts data into the interface table. It extracts data from the Idata structure resulted in the bizDocMapping service and puts the data into the interface table in Oracle Applications for GL Budgets Open Interfaces.

For more details on receive transactions, see [“Overview of Receive Service Transaction Processing” on page 31](#).

Business Document Structure

This service uses the business document GLBudgetBizDoc. Its structure is as follows:

■ GL_BUDGET

GL_BUDGET (Maps to GL_BUDGET_INTERFACE)

Field Name	Maps to Column	Description
BUDGET_NAME	BUDGET_NAME	Required. Name of the Budget in the Define Budget window.
BUDGET_ENTITY_NAME	BUDGET_ENTITY_NAME	Required. Budget Organization Name in the Define Budget Organization window.
FISCAL_YEAR	FISCAL_YEAR	Required. Fiscal year for budgeting.
CURRENCY_CODE	CURRENCY_CODE	Required. Budget currency code.
UPDATE_LOGIC_TYPE	UPDATE_LOGIC_TYPE	Use R to indicate that the amount has to be replaced; use A to indicate that the amount will be added to the existing amount for the budget.
SET_OF_BOOKS_NAME	SET_OF_BOOKS_ID	Required. Derive from GL_SETS_OF_BOOKS.
PERIOD1_AMOUNT through PERIOD60_AMOUNT	PERIOD1_AMOUNT through PERIOD60_AMOUNT	Amounts for a maximum of 60 periods in a fiscal year.
SEGMENT1 through SEGMENT30	SEGMENT1 through SEGMENT30	Segments in the accounting flexfield structure.

Receive Journal Service

The name of this service is:

WmOAFIN107SC.generalLedger107SC.intoOA.journal.receiveJournal

This service inserts Journals into the General Ledger production tables. You can import Journals such as Accounting, Encumbrance, and Multi-currency.

Database Scripts

This service uses the following database scripts:

Database Script	Description
wm_install_into_journal.sql	Runs all the scripts listed below, except the uninstall script.
wm_into_journal_pkg.sql	Installs WM_JOURNAL_IMP_HANDLER_PKG, WM_HANDLE_JOURNAL, which calls the WM_CONC_REQUEST.WM_REQUEST_SUBMIT procedure to submit the journal concurrent process.
wm_drop_into_journal.sql	Uninstalls all components created by wm_install_into_journal.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definitions:

- setJournalTxn107SC.txp
- JournalTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow executes as follows:

- **specifyDefaultSettings** specifies the default parameter settings required for service execution. You should change these settings accordingly.
- **setConcProgParams** map sets the default values of concurrent program parameters:
- **bizDocMapping** maps the incoming business document structure to the required Oracle Applications data structures (interface tables). This service uses the following services internally to provide required data feed during the mapping.
 - **getBudgetVersionID** is a transformer that maps the business doc Idata structure to the interface table Idata structure. It takes BUDGET_NAME as the input parameter and queries the table GL_BUDGET_VERSIONS and gets the BUDGET_VERSION_ID corresponding to the BUDGET_NAME.
 - **getCodeCombinationID** is a transformer that maps the business doc Idata structure to the interface table Idata structure. It takes CONCATENATED_SEGMENTS and SET_OF_BOOKS_NAME as the input parameters and queries the tables GL_CODE_COMBINATIONS_KFV and GL_SETS_OF_BOOKS to get the CODE_COMBINATION_ID.
 - **getSOBId** is a transformer that maps the business doc Idata structure to the interface table Idata structure. It takes set of books short name as the input parameter and queries the table GL_SETS_OF_BOOKS and gets the SET_OF_BOOKS_ID.
 - **getEncumTypeId** is a transformer that maps the business doc Idata structure to the interface table Idata structure. It takes ENCUMBRANCE_TYPE as the input parameter and queries the table GL_ENCUMBRANCE_TYPES to get the ENCUMBRANCE_TYPE_ID.
 - **convertToDateObject** converts a date string to the date object.
 - **getOASystemDate** maps the current date as a date-object.
 - **getOracleAppsUserId** is a transformer for mapping the business document Idata structure to the interface table Idata structure. It takes ORACLE_APPS_USER_NAME as the input parameter from the business document, and queries the table FND_USER to get the USER_ID. The USER_ID information is required for insertion into the interface tables.
- **setJournalTxn** inserts data into the interface table. It extracts data from the Idata structure resulted in the bizDocMapping service and puts the data into the interface table in Oracle Applications for GL.

- **importJournal** imports data to the production table from the interface table. It then calls the **execJournalConcProg**, **checkInterfaceTabStatus**, and **getJournalImport_ERR** services to execute the corresponding concurrent program that inserts data into the production table to generate the error or acknowledgement message. If the status of the execution is **SUCCESS** (returned by the service **execJournalConcProg**), it checks for the record having the returned request ID in interface table and indicates an error during import. In this case, this service calls **getJournalImport_ERR** to retrieve the errors. If no record is found, it comes out of the flow indicating success of the data import process. If the status of the execution is **FAILED** it returns appropriate error message and comes out of execution.
- **execJournalConcProg** invokes the **WM_JOURNAL_IMP_HANDLER_PKG.WM_HANDLE_JOURNAL** stored procedure. This procedure calls corresponding concurrent subroutine to execute the data import process for Journal Entry into Oracle Application. This service produces the status of the execution and the request ID generated for the request posted in Oracle Applications for the concurrent program.
- **checkJournalImportStatus** checks the status of the execution of the Journal Import. It returns the number of rows in the interface table for a specific request ID, which were not successfully imported.
- **getJournalImport_ERR** gets the error message that occurs during the data import to the production table from interface table. Based on the parameter request ID it scans **GL_INTERFACE TABLE** to get the corresponding message matches to the request ID.

For more details on receive transactions, see [“Overview of Receive Service Transaction Processing” on page 31](#).

Business Document Structure

This service uses the business document **JournalBizDoc**. Its structure is as follows:

- GL

GL (Maps to GL_INTERFACE)

Field Name	Maps to Column	Description
SET_OF_BOOKS_NAME	SET_OF_BOOKS_ID	Required. Accounting books defining column. Mapping required from GL_SETS_OF_BOOKS table.
ACCOUNTING_DATE	ACCOUNTING_DATE	Required. Effective date of the transaction, maps directly.
CURRENCY_CODE	CURRENCY_CODE	Required. Currency Code, maps directly.
DATE_CREATED	DATE_CREATED	Required. System date. The DATE_CREATED from business document not used.
CREATED_BY	CREATED_BY	Required. The CREATED_BY from business document not used.
ACTUAL_FLAG	ACTUAL_FLAG	Required. Balance type (actual, budget, or encumbrance). This field maps directly.
CATEGORY_NAME	USER_JE_CATEGORY_NAME	Required. Journal entry category user defined name. This field maps directly.
SOURCE_NAME	USER_JE_SOURCE_NAME	Required. Journal entry source user defined name. This field maps directly.
CURRENCY_CONVERSION_DATE	CURRENCY_CONVERSION_DATE	Date of exchange rate. This field maps directly.
ENCUMBRANCE_TYPE	ENCUMBRANCE_TYPE_ID	Encumbrance type defining column. Mapping required from GL_ENCUMBRANCE_TYPE table.
BUDGET_NAME	BUDGET_VERSION_ID	Budget version defining column. Mapping required from GL_BUDGET_VERSIONS.
USER_CURRENCY_CONVERSION_TYPE	USER_CURRENCY_CONVERSION_TYPE	Currency conversion type. This field maps directly.
CURRENCY_CONVERSION_RATE	CURRENCY_CONVERSION_RATE	Foreign currency exchange rate. This field maps directly.
ENTERED_DR	ENTERED_DR	Transaction debit amount, entered currency; maps directly. Either ENTERED_DR or ENTERED_CR should be entered.

Field Name	Maps to Column	Description
ENTERED_CR	ENTERED_CR	Transaction credit amount, entered currency. This field maps directly. Valid value are: <ul style="list-style-type: none"> ■ ENTERED_DR ■ ENTERED_CR
ACCOUNTED_DR	ACCOUNTED_DR	Transaction debit amount, base currency. This field maps directly.
ACCOUNTED_CR	ACCOUNTED_CR	Transaction credit amount, base Currency. This field maps directly.
REFERENCE	REFERENCE1	Journal Import reference column which is the Batch name for the Journal. This field maps directly.
BATCH_NAME		Not used. Journal entry batch defining column.
PERIOD_NAME	PERIOD_NAME	Period Name. This field maps directly..
ACCOUNT_CODE	CODE_COMBINATION_ID	Combination defining column; Mapping required from GL_CODE_COMBINATIONS table.
STAT_AMOUNT	STAT_AMOUNT	Statistical amount; maps directly.
INVOICE_DATE	INVOICE_DATE	DATE Value-added tax descriptive flexfield column. This field maps directly.
TAX_CODE	TAX_CODE	Value-added tax descriptive flexfield column. This field maps directly.
INVOICE_IDENTIFIER	INVOICE_IDENTIFIER	Value added tax descriptive flexfield column. This field maps directly.
INVOICE_AMOUNT	INVOICE_AMOUNT	Value added tax descriptive flexfield column. This field maps directly.
USSGL_TRANSACTION_CODE	USSGL_TRANSACTION_CODE	Government transaction code. This field maps directly.
JGZZ_RECON_REF	JGZZ_RECON_REF	Global reconciliation reference. This field maps directly.

Receive Mass Additions Service

The name of this service is:

WmOAFIN107SC.fixedAssets107SC.intoOA.massAdditions:receiveMassAdditions

This service creates assets automatically from information in any other system. Since Assets are already integrated with Payables, you can use them to integrate with other payables systems. If multiple distributions exist for an asset, you must enter them using the Assignments window for Assets in Oracle Applications.

The posting of the Mass Additions program in Oracle Applications is a request set. Since this version of Oracle Applications does not provide any API to submit a Request set, this interface will only insert records into the interface table.

You must use the concurrent request outputs to view errors because there are no interface error tables for storing the error records.

Database Scripts

This service uses the following database scripts:

Database Script	Description
wminstall_into_massadditions.sql	Runs all the scripts listed below, except the uninstall script.
wm_into_massadditions_pkg.sql	Installs WM_FA_MASSADD_IMP_HANDLER_PKG. WM_HANDLE_FA_MASSADD, which calls the WM_CONC_REQUEST. WM_REQUEST_SUBMIT procedure to submit the mass additions concurrent process.
wm_into_massadditions_seq.sql	Creates the following components: <ul style="list-style-type: none">■ WM_FA_MASSADD_S which creates the unique MASS_ADDITION_ID.■ WM_FA_CREATEBATCH_S which creates the BATCH_ID.
wm_drop_into_massadditions.sql	Uninstalls all components created by wminstall_into_massadditions.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definitions:

- setMassAdditionsTxn107SC.txp
- MassAdditionsTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow executes as follows:

- **specifyDefaultSettings** specifies the default parameter settings required for service execution. You should change these settings accordingly.
- **bizDocMapping** maps the incoming Business Document structure to the required Oracle Applications data structures (interface tables).
 - **pickSequence** is a transformer that maps the business doc Idata structure to the interface table Idata structure. It picks up a new sequence number for BATCH_ID from the sequence WM_FA_CREATEBATCH_S.
 - **pickSequence** is a transformer that maps the business doc Idata structure to the interface table Idata structure. It picks up a new sequence number for MASS_ADDITION_ID from the sequence WM_FA_MASSADD_S
 - **getAssetCategoryId** is a transformer that maps the business doc Idata structure to the interface table Idata structure. It is used to select the Asset Category Id for the Asset Category Description.
 - **getCodeCombinationId** is used for the Payables Code Combination Id and Expense Code Combination Id services as a transformer that maps the business doc Idata structure to the interface table Idata structure. It selects the Payables Code Combination ID for the Payables Account Code Concatenated Segments information.
 - **getLocationCodeCombination** is a transformer that maps the business doc Idata structure to the interface table Idata structure. It is used to select the LOCATION_ID for the Concatenated Location information.
 - **getEmpIdFromFullName** is a transformer that maps the business doc Idata structure to the interface table Idata structure. It is used to select the ASSIGNED_TO Id for the ASSIGNED_TO_NAME.
 - **getAdditionalAssetId** is a transformer that maps the business doc Idata structure to the interface table Idata structure. It is used to select the PARENT_ASSET_ID for the PARENT_ASSET_NUMBER.

- `getAssetKeyCCId` is a transformer that maps the business doc IData structure to the interface table IData structure. It is used to select the ASSET_KEY_CCID which is CODE_COMBINATION_ID for the ASSET_KEY_ACCOUNT_CODE, Concatenated Segments information provided.
- `setMassAdditionsTxn` inserts data into the interface table. It extracts data from the IData structure resulted in the bizDocMapping service and puts the data into the interface table in Oracle Applications for FA Mass Additions Interface Table.

For more details on receive transactions, see [“Overview of Receive Service Transaction Processing” on page 31](#).

Business Document Structure

This service uses the business document MassAdditionsBizDoc. Its structure is as follows:

- MASS_ADDITIONS_HEADER

MASS_ADDITIONS_HEADER (Maps to FA_MASS_ADDITIONS)

Field Name	Maps to Column	Description
ACCOUNTING_DATE	ACCOUNTING_DATE	Required. Accounting date of the Invoice.
AP_DISTRIBUTION_LINE_NUMBER	AP_DISTRIBUTION_LINE_NUMBER	Line Number in Payables Distribution.
ASSET_CATEGORY_DESCRIPTION	ASSET_CATEGORY_ID	Required. Derived from the OA view FA_CATEGORIES_KFV. Refer to the transformer.
ASSET_KEY_ACCOUNT_CODE	ASSET_KEY_CCID	Validated against FA_ASSET_KEYWORDS_KFV. Refer to the transformer for information.
ASSET_NUMBER	ASSET_NUMBER	Used for Asset Numbering. If this field is null, an Asset Number is assigned
ASSET_TYPE	ASSET_TYPE	Required. Type of Asset. Valid values are CAPITALIZED, CIP, and EXPENSED.
ASSIGNED_TO_NAME	ASSIGNED_TO	Derives the Employee Id from FA_EMPLOYEES. Refer to the transformer for information.
BOOK_TYPE_CODE	BOOK_TYPE_CODE	Required. Choose a Book that has been defined in FA_BOOK_CONTROLS. This Book must be CORPORATE type only.
CREATE_BATCH_DATE	CREATE_BATCH_DATE	Date of creation

Field Name	Maps to Column	Description
CONTEXT	CONTEXT	Used to bring in the date when loaded into Assets
DATE_PLACED_IN_SERVICE	DATE_PLACED_IN_SERVICE	Required. Date the Asset was placed into service.
DEPRECIATE_FLAG	DEPRECIATE_FLAG	Indicates whether Oracle Assets should depreciate this asset.
DEPRN_RESERVE	DEPRN_RESERVE	Accumulated depreciation of this asset in General Ledger.
DESCRIPTION	DESCRIPTION	Required. Description of the Asset.
DIST_NAME	DIST_NAME	Required. Distribution Set Name.
EXPENSE_ACCOUNT_CODE SET_OF_BOOKS_NAME	EXPENSE_CODE_COMBINATION_ID	Required. GL Code to which the depreciation expense should be charged. Code Combination ID from GL_CODE_COMBINATIONS_KFV.
FEEDER_SYSTEM_NAME	FEEDER_SYSTEM_NAME	Name of the Source system for interfacing.
FIXED_ASSETS_COST	FIXED_ASSET_COST	Required. Cost of the Asset.
FIXED_ASSETS_UNITS	FIXED_ASSET_UNITS	Required. Number of Assets.
FULLY_RSVD_REVALS_COUNTER	FULLY_RSVD_REVALS_COUNTER	Number of times the Asset has been revalued.
INVOICE_NUMBER	INVOICE_NUMBER	Invoice number of the Payables Invoice
CONCATENATED_LOCATION	LOCATION_ID	Required. Location Id is derived from FA_LOCATIONS. The location of an asset. Refer to the transformer.
MANUFACTURER_NAME	MANUFACTURER_NAME	Name of the manufacturer
MERGE_INVOICE_NUMBER	MERGE_INVOICE_NUMBER	Set to Invoice Number for unmerged lines
MERGE_VENDOR_NUMBER	MERGE_VENDOR_NUMBER	Set to Vendor Number for unmerged lines
MODEL_NUMBER	MODEL_NUMBER	Model Number of the Asset
PARENT_ASSET_NUMBER	PARENT_ASSET_ID	If this Asset is component of another parent Asset. Derived from FA_ADDITIONS. Refer to Transformer
PAYABLES_ACCOUNT_CODE	PAYABLES_CODE_COMBINATION_ID	Required. Code Combination for the Asset Clearing Account assigned to the Asset Category. Refer to the transformer.

Field Name	Maps to Column	Description
PAYABLES_COST	PAYABLES_COST	Required. Original Cost of the Asset.
PAYABLES_UNITS	PAYABLES_UNITS	Required. Number of units that make up the Asset.
PO_NUMBER	PO_NUMBER	Purchase Order Number of the Asset if available
PRODUCTION_ CAPACITY_NUMBER	PRODUCTION_ CAPACITY	Capacity of units of a production asset.
REVAL_AMORTIZATION_ BASIS	REVAL_AMORTIZATION_ BASIS	Basis for amortization of revaluation reserve.
REVAL_RESERVE	REVAL_RESERVE	Revaluation Reserve of a revalued asset.
REVIEWER_COMMENTS	REVIEWER_COMMENTS	Details about the asset.
SALVAGE_VALUE	SALVAGE_VALUE	Salvage Value of the asset.
SERIAL_NUMBER	SERIAL_NUMBER	Serialnumber of the asset.
TAG_NUMBER	TAG_NUMBER	Tag number.
UNIT_OF_MEASURE	UNIT_OF_MEASURE	Unit of measure of a production asset. If this field is null, Oracle Assets uses the Unit of Measure from the Asset Category.
UNREVALUE_COST	UNREVALUED_COST	Cost without regard to any revaluation of a revalued asset.
YTD_DEPRN	YTD_DEPRN	Year to date depreciation.
YTD_REVAL_DEPRN_ EXPENSE	YTD_REVAL_DEPRN_ EXPENSE	Year to date depreciation expense due to revaluation.

Send Accounts Payable Invoice Service

The name of the serviced is:

WmOAFIN107SC.payables107SC.fromOA.APInvoice:sendAPInvoice

This service retrieves new or changed approved invoice records only. You set up Accounts Payable Invoices in Oracle Applications Payables to record invoice-related information.

There is no deletion of payments defined in Oracle Applications; therefore the document status of DELETE does not apply in this case. However, since Oracle Applications does not maintain the Invoice Approval History, the first approval for an invoice cannot be distinguished from subsequent invoices; therefore a business document is created with the document status of UPDATE. The document status of INSERT does not apply.

Database Scripts

This service uses the following database scripts:

Database Script	Description
wm_install_from_apinvoice.sql	Runs all the scripts listed below, except the uninstall script.
wm_from_apinvoice_vw.sql	Creates the following required core view components: <ul style="list-style-type: none"> ■ WM_AP_INVOICES_VW ■ WM_AP_INVOICES_LINES_VW ■ WM_AP_INVOICES_QRY_VW
wm_from_apinvoice_trg.sql	Creates the following trigger component to be used for writing to the WM_TRACKCHANGES table to indicate a new or updated document: <ul style="list-style-type: none"> ■ WM_AP_INVOICE_DIST_IU_TRG
wm_disable_from_apinvoice.sql.	Disables the triggers installed by wm_from_apinvoice_trg.sql.
wm_enable_from_apinvoice.sql	Re-enables the triggers installed by wm_from_apinvoice_trg.sql.
wm_drop_from_apinvoice.sql	Uninstalls all components created by wm_install_from_apinvoice.sql.

Supporting Transaction Definitions

This service uses the following transaction definitions:

- `getAPInvoiceTxn107SC.txp`

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow executes as follows:

- `specifyDefaultSettings` specifies the default parameter settings required for service execution. You should change these settings accordingly.
- `lockTxnCtrl` determines whether the `sendAPInvoice` service is ready for execution. That is, it determines whether there is no other instance of this service already in process.
 - If `lockTxnCtrl` returns `False`, it means that another instance of this service is already in process. The service exits, and waits for next scheduled execution.
 - If `lockTxnCtrl` returns `True`, it means that the service is ready to execute. The `APInvoice` row in the control table is locked and updated so that the status is changed to `INPROCESS` to prevent another `APInvoice` service from executing.
- `getAPInvoiceTxn` queries the Oracle Applications database for any `APInvoice` Transaction to be processed. The number of records returned depends on the parameter value specified in the above map.
- `processBizDoc` sends the business document to the recipient such as a trading partner by looping against each document. You must customize this step to receive a `SUCCESS` or an `ERROR` status of the document transfer along with the Error information. The Transfer Status and any Error Information are logged against each document.
- Loop against each document. Based on the Debug Mode specified during execution, it either purges or updates the records in the `WM_TRACKCHANGES` custom table.
 - If the Debug Mode is `TRUE`, the records in the `WM_TRACKCHANGES` table are updated and the `PROCESSED_FLAG` is set to `Y`. This ensures that the same sets of records are not picked up during next polling interval. The `updateTrackChanges` service updates the `PROCESSED_FLAG` in the `WM_TRACKCHANGES` table to `Y` and updates `PROCESSED_DATE` to `sysdate` so that same information is not picked up again during next polling instance.
 - If the Debug Mode is `FALSE`, the records in the `WM_TRACKCHANGES` table are deleted; the `purgeTrackChanges` service purges the records from the `WM_TRACKCHANGES` table.

- Loop against each document and based on the Transfer Status, insertTransferERRInfo inserts a new record in the WM_TRACKCHANGES table so that same document can be picked up during the next polling interval.
- unlockTxnCtrl releases the lock on the Custom Control table so that next polling instance of sendVendor service can begin.
- getLastError logs any errors in the above steps.
- unlockTxnCtrl service executes to release the lock on the Custom Control table.

For more details on receive transactions, see [“Overview of Receive Service Transaction Processing” on page 31](#).

Business Document Structure

This service uses the following business document structure:

- 1.0 AP_INVOICE
 - 1.1 AP_INVOICE_LINES

1.0 AP_INVOICE

Document Field	Oracle Applications Table/View Name	Column Name	Description
WEB_TRANSACTION_ID			
DOCUMENT_TYPE			Value is DOCUMENT_TYPE.
DOCUMENT_STATUS			Value is UPDATE or INSERT.
INVOICE_ID	AP_INVOICES_ALL	INVOICE_ID	Unique identifier for an Invoice.
INVOICE_NUM	AP_INVOICES_ALL	INVOICE_NUM	Invoice Number.
INVOICE_TYPE	AP_INVOICES_ALL	INVOICE_TYPE	Valid values are Standard and Credit Memo.
INVOICE_DATE	AP_INVOICES_ALL	INVOICE_DATE	Invoice date.
PO_NUMBER	AP_INVOICES_PKG_GET_PO_NUMBER	INVOICE_ID	Purchase Order number.
VENDOR_NUMBER	PO_VENDORS	SEGMENT1	Vendor Number.
VENDOR_NAME	PO_VENDORS	VENDOR_NAME	Name of vendor

Document Field	Oracle Applications Table/View Name	Column Name	Description
VENDOR_SITE_ADDRESS_LINE1	PO_VENDOR_SITES_ALL	ADDRESS_LINE1	Address of the Vendor Site. Used to derive the Vendor Site ID in the interface table
VENDOR_SITE_ADDRESS_LINE2	PO_VENDOR_SITES_ALL	ADDRESS_LINE2	See ADDRESS_LINE1.
VENDOR_SITE_ADDRESS_LINE3	PO_VENDOR_SITES_ALL	ADDRESS_LINE3	See ADDRESS_LINE1.
VENDOR_TOWN_OR_CITY	PO_VENDOR_SITES_ALL	CITY	See ADDRESS_LINE1.
VENDOR_COUNTY	PO_VENDOR_SITES_ALL	COUNTY	See ADDRESS_LINE1.
VENDOR_STATE	PO_VENDOR_SITES_ALL	STATE	See ADDRESS_LINE1.
VENDOR_POSTAL_CODE	PO_VENDOR_SITES_ALL	ZIP	See ADDRESS_LINE1.
VENDOR_COUNTRY	PO_VENDOR_SITES_ALL	COUNTRY	See ADDRESS_LINE1.
INVOICE_AMOUNT	AP_INVOICES_ALL	INVOICE_AMOUNT	Invoice Amount.
INVOICE_CURRENCY_CODE	AP_INVOICES_ALL	INVOICE_CURRENCY_CODE	Currency of Invoice.
EXCHANGE_RATE	AP_INVOICES_ALL	EXCHANGE_RATE	Exchange Rate for foreign currency Invoices.
EXCHANGE_RATE_TYPE	AP_INVOICES_ALL	EXCHANGE_RATE_TYPE	Rate Type, for example, Corporate or User.
TERMS_NAME	AP_INVOICES_ALL	TERMS_NAME	Payment Terms name
EXCHANGE_RATE_DATE	AP_INVOICES_ALL	EXCHANGE_RATE_DATE	Date on which the exchange rate has to be taken.
DESCRIPTION	AP_INVOICES_ALL	DESCRIPTION	Description of invoice.
AWT_GROUP_NAME	AP_AWT_GROUPS	NAME	Withholding tax name
SOURCE	AP_INVOICES_ALL	SOURCE	Source of the invoice
PAYMENT_CROSS_RATE	AP_INVOICES_ALL	PAYMENT_CROSS_RATE	Exchange rate between invoice and payment; usually value is 1 unless they are associated fixed-rate currencies.

Document Field	Oracle Applications Table/View Name	Column Name	Description
PAYMENT_CURRENCY_CODE	AP_INVOICES_ALL	PAYMENT_CURRENCY_CODE	Cross currency payment currency.
DOC_CATEGORY_CODE	AP_INVOICES_ALL	DOC_CATEGORY_CODE	This defaults to standard or credit depending on the Invoice type. You can assign a valid document category that exists in Accounts Payable.
VOUCHER_NUM	AP_INVOICES_ALL	VOUCHER_NUM	Voucher number
PAYMENT_METHOD_LOOKUP_CODE	AP_INVOICES_ALL	PAYMENT_METHOD_LOOKUP_CODE	Name of payment method
PAY_GROUP_LOOKUP_CODE	AP_INVOICES_ALL	PAY_GROUP_LOOKUP_CODE	Name of pay group.
GOODS_RECEIVED_DATE	AP_INVOICES_ALL	GOODS_RECEIVED_DATE	Date invoice items received.
INVOICE_RECEIVED_DATE	AP_INVOICES_ALL	INVOICE_RECEIVED_DATE	Date invoice received.
LIABILITY_ACCOUNT	GL_CODE_COMBINATIONS_KFV	CONCATENATED_SEGMENTS	Accounts Payable liability GL Code Combination Id is derived from the accounts code combination
USSGL_TRANSACTION_CODE	AP_INVOICES_ALL	USSGL_TRANSACTION_CODE	Default transaction code for creating US Standard General Ledger journal entries
EXCLUSIVE_PAYMENT	AP_INVOICES_ALL	EXCLUSIVE_PAYMENT	Exclusive payment flag
ORGANIZATION_NAME	HR_ALL_ORGANIZATION_UNITS_TL	NAME	Organization name is used to derive the ORG_ID from organizations definitions
AMOUNT_APPLICABLE_TO_DISCOUNT	AP_INVOICES_ALL	AMOUNT_APPLICABLE_TO_DISCOUNT	Amount of invoice applicable to a discount

1.1 AP_INVOICE_LINES

Document Field	Oracle Applications Table/View Name	Column Name	Description
INVOICE_ID	AP_INVOICE_DISTRIBUTIONS_ALL	INVOICE_ID	Unique identifier for an Invoice.
LINE_NUMBER	AP_INVOICE_DISTRIBUTIONS_ALL	LINE_NUMBER	Invoice line number
LINE_TYPE	AP_INVOICE_DISTRIBUTIONS_ALL	LINE_TYPE	Type of invoice line, such as item, freight, tax, or miscellaneous.
AMOUNT	AP_INVOICE_DISTRIBUTIONS_ALL	AMOUNT	Line amount.
ACCOUNTING_DATE	AP_INVOICE_DISTRIBUTIONS_ALL	ACCOUNTING_DATE	Accounting date.
DESCRIPTION	AP_INVOICE_DISTRIBUTIONS_ALL	DESCRIPTION	Description.
AMOUNT_INCLUDES_TAX_FLAG	AP_INVOICE_DISTRIBUTIONS_ALL	AMOUNT_INCLUDES_TAX_FLAG	Whether the line amount includes tax.
PO_NUMBER	PO_HEADERS_ALL	SEGMENT1	Purchase order number used for PO matching.
PO_LINE_NUMBER	PO_LINES_ALL	LINE_NUM	Purchase order line number used for PO matching.
PO_SHIPMENT_NUM	PO_LINE_LOCATIONS_ALL	SHIPMENT_NUM	Purchase order shipment number used for PO matching.
PO_DISTRIBUTION_NUM	PO_DISTRIBUTIONS_ALL	DISTRIBUTION_NUM	Purchase order distribution line number used for PO matching.
PO_UNIT_OF_MEASURE	PO_LINES_ALL	UNIT_MEAS_LOOKUP_CODE	Unit of measure on purchase order line.
ITEM_DESCRIPTION	MTL_SYSTEM_ITEMS	DESCRIPTION	Item description.
QUANTITY_INVOICED	AP_INVOICE_DISTRIBUTIONS_ALL	QUANTITY_INVOICED	Quantity invoiced against purchase order shipment.
SHIP_TO_LOC_ADDRESS_LINE1	HR_LOCATIONS	ADDRESS_LINE1	Used to derive the Location ID based on the ship to location address

Document Field	Oracle Applications Table/View Name	Column Name	Description
SHIP_TO_LOC_ADDRESS_LINE2	HR_LOCATIONS	ADDRESS_LINE2	See ADDRESS_LINE1.
SHIP_TO_LOC_ADDRESS_LINE3	HR_LOCATIONS	ADDRESS_LINE3	ADDRESS_LINE1.
SHIP_TO_LOC_TOWN_OR_CITY	HR_LOCATIONS	TOWN_OR_CITY	Unit price for purchase order matched invoice items. Name of GL set of Books. Account flexfield for account associated with distribution line
SHIP_TO_LOC_COUNTY	HR_LOCATIONS	REGION_1	See TOWN_OR_CITY.
SHIP_TO_LOC_STATE	HR_LOCATIONS	REGION_2	See TOWN_OR_CITY.
SHIP_TO_LOC_POSTAL_CODE	HR_LOCATIONS	POSTAL_CODE	See TOWN_OR_CITY.
SHIP_TO_LOC_COUNTRY	HR_LOCATIONS	COUNTRY	See TOWN_OR_CITY.
UNIT_PRICE	AP_INVOICE_DISTRIBUTIONS_ALL	UNIT_PRICE	See TOWN_OR_CITY.
DIST_CODE_CONCATENATED	GL_CODE_COMBINATIONS_KFV	CONCATENATED_SEGMENTS	See TOWN_OR_CITY.
AWT_GROUP_NAME	AP_AWT_GROUPS	NAME	Withholding tax group name.
RELEASE_NUM	PO_HEADERS_ALL	REVISION_NUM	Blanket Purchase Order release number used for PO matching.
PROJECT_NAME	PA_PROJECTS_ALL	SEGMENT1	Project name validated against PA_PROJECTS_ALL.NAME.
TASK	PA_TASKS	TASK_NAME	Project Task Name validated against PA_TASKS.
EXPENDITURE_TYPE	AP_INVOICE_DISTRIBUTIONS_ALL	EXPENDITURE_TYPE	Project Expenditure type.
EXPENDITURE_ITEM_DATE	AP_INVOICE_DISTRIBUTIONS_ALL	EXPENDITURE_ITEM_DATE	Project expenditure item date.

Document Field	Oracle Applications Table/View Name	Column Name	Description
EXPENDITURE_ORGANIZATION_NAME	HR_ALL_ORGANIZATION_UNITS_TL	NAME	Project organization name.
PROJECT_ACCOUNTING_CONTEXT	AP_INVOICE_DISTRIBUTIONS_ALL	PROJECT_ACCOUNTING_CONTEXT	Project accounting context.
PA_ADDITION_FLAG	AP_INVOICE_DISTRIBUTIONS_ALL	PA_ADDITION_FLAG	Whether the invoice is for a project.
PA_QUANTITY	AP_INVOICE_DISTRIBUTIONS_ALL	PA_QUANTITY	Quantity.
USSGL_TRANSACTION_CODE	AP_INVOICE_DISTRIBUTIONS_ALL	USSGL_TRANSACTION_CODE	USSGL transaction code for creating US Standard General Ledger journal entries.
STAT_AMOUNT	AP_INVOICE_DISTRIBUTIONS_ALL	STAT_AMOUNT	Amount associated with distribution line for measuring statistical quantities.
TYPE_1099	AP_INVOICE_DISTRIBUTIONS_ALL	TYPE_1099	1099 type.
INCOME_TAX_REGION_LONG_NAME	AP_INCOME_TAX_REGIONS	REGION_LONG_NAME	Reporting region for distribution line for 1099 supplier.
PA_REFERENCE1	AP_INVOICE_DISTRIBUTIONS_ALL	PA_REFERENCE1	PA-related reference.
PA_REFERENCE2	AP_INVOICE_DISTRIBUTIONS_ALL	PA_REFERENCE2	PA-related reference.
ASSET_TRACKING_FLAG	AP_INVOICE_DISTRIBUTIONS_ALL	ASSET_TRACKING_FLAG	Indicates whether tracked as an asset.
PRICE_CORRECTION_FLAG	AP_INVOICE_DISTRIBUTIONS_ALL	PRICE_CORRECTION_FLAG	Indicates whether price adjustment was done.

Send Accounts Payable Payment Service

The name of this service is:

WmOAFIN107SC.payables107SC.fromOA.APPayment:sendAPPayment

This service retrieves Accounts Payable Payments. You set up Accounts Payable Payments in Oracle Applications Payables to record payment-related information.

The following business rules apply when retrieving Accounts Payable Payment data from Oracle Applications:

- Picks up new or changed payment records only.
- Picks up confirmed payments only.
- No deletion of payments defined in Oracle Applications. Therefore, a business document with document status of INSERT or UPDATE will be created. The document status of DELETE does not apply in this case.

Database Scripts

This service uses the following database scripts:

Database Script	Description
wm_install_from_appayment.sql	Runs all the scripts listed below, except the uninstall script.
wm_from_appayment_vw.sql	Creates the following required view components for AP Payments outbound transactions: <ul style="list-style-type: none"> ■ WM_AP_CHECKS_VW ■ WM_AP_INVOICE_PAYMENTS_VW ■ WM_AP_CHECKS_QRY_VW
wm_from_appayment_trg.sql	Creates the following trigger component to be used for writing to the WM_TRACKCHANGES table to indicate a new or updated document: <ul style="list-style-type: none"> ■ WM_AP_PAYMENTS_IU_TRG
wm_disable_from_appayment.sql	Disables the triggers installed by wm_from_appayment_trg.sql.

Database Script	Description
wm_enable_from_appayment.sql	Re-enables the triggers installed by wm_from_appayment_trg.sql.
wm_drop_from_appayment.sql	Uninstalls all components created by wm_install_from_appayment.sql.

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definition:

- `getAPPaymentTxn107SC.txp`

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow executes as follows:

- `specifyDefaultSettings` specifies the default parameter settings required for service execution. You should change these settings accordingly.
- `lockTxnCtrl` determines whether the `sendAPPayment` service is ready for execution. That is, it determines whether there is no other instance of this service already in process.
 - If `lockTxnCtrl` returns `False`, it means that another instance of this service is already in process. The service exits, and waits for next scheduled execution.
 - If `lockTxnCtrl` returns `true`, it means that the service is ready to execute. The `APPayment` row in the control table is locked and updated so that the status is changed to `INPROCESS` to prevent any other `APPayment` service from executing.
- `getAPPaymentTxn` queries the Oracle Applications database for any `APPayment` Transaction to be processed. The number of records returned depends on the parameter value specified in the above map.
- `processBizDoc` sends the business document to the recipient such as a trading partner by looping against each document. You must customize this step to receive a `SUCCESS` or an `ERROR` status of the document transfer along with the Error information. The Transfer Status and any Error Information are logged against each document.

- Loop against each document. Based on the Debug Mode specified during execution, it either purges or updates the records in the WM_TRACKCHANGES custom table.
 - If the Debug Mode is TRUE, the records in the WM_TRACKCHANGES table are updated and the PROCESSED_FLAG is set to Y. This ensures that same sets of records are not picked up during next polling interval. The `updateTrackChanges` service updates the PROCESSED_FLAG in the WM_TRACKCHANGES table to Y and updates PROCESSED_DATE to sysdate so that same information is not picked up again during next polling instance.
 - If the Debug Mode is FALSE, the records in the WM_TRACKCHANGES table are deleted. The `purgeTrackChanges` service purges the records from the WM_TRACKCHANGES table.
- Loop against each document. Based on the Transfer Status, `insertTransferERRInfo` inserts a new record in the WM_TRACKCHANGES table so that same document can be picked up during the next polling interval.
- `unlockTxnCtrl` releases the lock on the Custom Control table so that next polling instance of `sendVendor` service can begin.
- `getLastError` logs any errors occurring in the above steps.
- `unlockTxnCtrl` releases the lock on the Custom Control table.

For more details on receive transactions, see [“Overview of Receive Service Transaction Processing” on page 31](#).

Business Document Structure

This service uses the following business document structure:

- 1.0 AP_CHECKS
 - 1.1 AP_INVOICE_PAYMENTS

1.0 AP_CHECKS

Document Field	Oracle Applications Table/View Name	Column Name	Description
WEB_TRANSACTION_ID			
DOCUMENT_TYPE			The value of this field will be APPAYMENT.
DOCUMENT_STATUS			Valid values are UPDATE, INSERT, or QUERY.

Document Field	Oracle Applications Table/View Name	Column Name	Description
CHECK_ID	AP_CHECKS_ALL	CHECK_ID	Unique payment identifier.
ORGANIZATION_NAME	HR_ALL_ORGANIZATION_UNITS_TL	NAME	Organization name.
CHECK_NUMBER	AP_CHECKS_ALL	CHECK_NUMBER	Payment number.
CURRENCY_CODE	AP_CHECKS_ALL	CURRENCY_CODE	Currency code.
AMOUNT	AP_CHECKS_ALL	AMOUNT	Payment amount.
CHECK_DATE	AP_CHECKS_ALL	CHECK_DATE	Payment date.
BATCH_NAME	AP_CHECKS_ALL	CHECKRUN_NAME	Payment batch name.
CHECK_VOUCHER_NUM	AP_CHECKS_ALL	CHECK_VOUCHER_NUM	Payment voucher number.
CHECK_STATUS	AP_LOOKUP_CODES	DISPLAYED_FIELD	Payment Status.
CLEARED_AMOUNT	AP_CHECKS_ALL	CLEARED_AMOUNT	Payment cleared amount.
CLEARED_BASE_AMOUNT	AP_CHECKS_ALL	CLEARED_BASE_AMOUNT	Payment cleared base amount.
CLEARED_DATE	AP_CHECKS_ALL	CLEARED_DATE	Payment cleared date.
CLEARED_EXCHANGE_DATE	AP_CHECKS_ALL	CLEARED_EXCHANGE_DATE	Payment cleared exchange date.
CLEARED_EXCHANGE_RATE	AP_CHECKS_ALL	CLEARED_EXCHANGE_RATE	Payment cleared exchange rate.
CLEARED_EXCHANGE_RATE_TYPE	AP_CHECKS_ALL	CLEARED_EXCHANGE_RATE_TYPE	Payment cleared exchange rate type.
VOID_DATE	AP_CHECKS_ALL	VOID_DATE	Payment void date.
VENDOR_NAME	AP_CHECKS_ALL	VENDOR_NAME	Supplier name.
VENDOR_ADDRESS_LINE1	PO_VENDOR_SITES_ALL	ADDRESS_LINE1	Supplier address.
VENDOR_ADDRESS_LINE2	PO_VENDOR_SITES_ALL	ADDRESS_LINE2	
VENDOR_ADDRESS_LINE3	PO_VENDOR_SITES_ALL	ADDRESS_LINE3	

Document Field	Oracle Applications Table/View Name	Column Name	Description
VENDOR_CITY	PO_VENDOR_SITES_ALL	CITY	
VENDOR_COUNTY	PO_VENDOR_SITES_ALL	COUNTY	
VENDOR_STATE	PO_VENDOR_SITES_ALL	STATE	
VENDOR_ZIP	PO_VENDOR_SITES_ALL	ZIP	
VENDOR_COUNTRY	PO_VENDOR_SITES_ALL	COUNTRY	
BANK_NAME	AP_BANK_BRANCHES	BANK_NAME	Payment Bank Name.
CURRENT_BANK_ACCOUNT_NAME	AP_BANK_ACCOUNTS_ALL	BANK_ACCOUNT_NAME	Payment Current Bank Account Name.
BANK_CURRENCY_CODE	AP_BANK_ACCOUNTS_ALL	CURRENCY_CODE	Payment Bank Currency Code.
CHECK_STOCK_NAME	AP_CHECK_STOCKS_ALL	NAME	Payment Stock Name.
PAYMENT_TYPE	AP_LOOKUP_CODES	DISPLAYED_FIELD	Payment Type.
PAYMENT_METHOD	AP_LOOKUP_CODES	DISPLAYED_FIELD	Payment method.
BANK_ACCOUNT_NAME	AP_CHECKS_ALL	BANK_ACCOUNT_NAME	Payment bank account name.
BANK_ACCOUNT_NUM	AP_CHECKS_ALL	BANK_ACCOUNT_NUM	Payment bank account number.
BANK_ACCOUNT_TYPE	AP_CHECKS_ALL	BANK_ACCOUNT_TYPE	Payment bank account type.
BANK_NUM	AP_CHECKS_ALL	BANK_NUM	Payment bank number.
BASE_AMOUNT	AP_CHECKS_ALL	BASE_AMOUNT	Payment base amount
PAYMENT_ADDRESS_STYLE	AP_CHECKS_ALL	ADDRESS_STYLE	Payment address.
PAYMENT_ADDRESS_LINE1	AP_CHECKS_ALL	ADDRESS_LINE1	

Document Field	Oracle Applications Table/View Name	Column Name	Description
PAYMENT_ADDRESS_LINE2	AP_CHECKS_ALL	ADDRESS_LINE2	
PAYMENT_ADDRESS_LINE3	AP_CHECKS_ALL	ADDRESS_LINE3	
PAYMENT_ADDRESS_LINE4	AP_CHECKS_ALL	ADDRESS_LINE4	
PAYMENT_CITY	AP_CHECKS_ALL	CITY	
PAYMENT_COUNTY	AP_CHECKS_ALL	COUNTY	
PAYMENT_STATE	AP_CHECKS_ALL	STATE	
PAYMENT_ZIP	AP_CHECKS_ALL	ZIP	
PAYMENT_COUNTRY	AP_CHECKS_ALL	COUNTRY	
DOC_SEQUENCE_NAME	FND_DOCUMENT_SEQUENCES	NAME	Payment document sequence name.
DOC_CATEGORY_NAME	FND_DOC_SEQUENCE_CATEGORIES	NAME	Payment document category name.
EXCHANGE_DATE	AP_CHECKS_ALL	EXCHANGE_DATE	Date exchange rate is effective. Usually the accounting date of the transaction.
EXCHANGE_RATE	AP_CHECKS_ALL	EXCHANGE_RATE	Exchange rate for foreign currency payment.
EXCHANGE_RATE_TYPE	AP_CHECKS_ALL	EXCHANGE_RATE_TYPE	Exchange rate for foreign currency payment.
FUTURE_PAY_DUE_DATE	AP_CHECKS_ALL	FUTURE_PAY_DUE_DATE	Negotiable date for future dated payment.
RELEASED_DATE	AP_CHECKS_ALL	RELEASED_DATE	Date and time user released stop payment.
RELEASED_BY	AP_CHECKS_ALL	RELEASED_BY	Name of person who released the payment.
STOPPED_DATE	AP_CHECKS_ALL	STOPPED_DATE	Date and time user released stop payment.
STOPPED_BY	AP_CHECKS_ALL	STOPPED_BY	Name of person who stopped the payment.

Document Field	Oracle Applications Table/View Name	Column Name	Description
TREASURY_PAY_DATE	AP_CHECKS_ALL	TREASURY_PAY_DATE	Date payment processed through internal clearing organization.
TREASURY_PAY_NUMBER	AP_CHECKS_ALL	TREASURY_PAY_NUMBER	Number assigned to payment processed through internal clearing organization.
USSGL_TRANSACTION_CODE	AP_CHECKS_ALL	USSGL_TRANSACTION_CODE	Transaction code for creating US Standard General Ledger journal entries.
USSGL_TRX_CODE_CONTEXT	AP_CHECKS_ALL	USSGL_TRX_CODE_CONTEXT	USSGL transaction code descriptive flexfield context column.
SET_OF_BOOKS_NAME	GL_SETS_OF_BOOKS	NAME	Name of GL set of books.
CHECK_FORMAT_NAME	AP_CHECK_FORMATS	NAME	Check format name.
TERRITORY_SHORT_NAME	FND_TERRITORIES_VL	TERRITORY_SHORT_NAME	Name of territory.
USER_RATE_TYPE	GL_DAILY_CONVERSION_TYPES	USER_CONVERSION_TYPE	Rate type entered by user.
POSITIVE_PAY_STATUS_CODE	AP_CHECKS_ALL	POSITIVE_PAY_STATUS_CODE	Set by and used by positive pay. Report to select records.
TRANSFER_PRIORITY	AP_CHECKS_ALL	TRANSFER_PRIORITY	Transfer priority.
EXTERNAL_BANK_ACCOUNT_NAME	AP_BANK_ACCOUNTS_ALL	BANK_ACCOUNT_NAME	External bank account name.
FUTURE_DATED_PAYMENT_FLAG	AP_CHECKS_ALL	FUTURE_PAY_DUE_DATE	Indicates whether future date payment.

1.1 AP_INVOICE_PAYMENTS

Document Field	Oracle Applications Table/View Name	Column Name	Description
CHECK_ID	AP_INVOICE_PAYMENTS_ALL	CHECK_ID	Unique Payment identifier.
INVOICE_NUM	AP_INVOICES_ALL	INVOICE_NUM	Payment invoice number.
AMOUNT	AP_INVOICE_PAYMENTS_ALL	AMOUNT	Payment invoiced amount.
ACCOUNTING_DATE	AP_INVOICE_PAYMENTS_ALL	ACCOUNTING_DATE	Accounting date.
DESCRIPTION	AP_INVOICES_ALL	DESCRIPTION	Payment invoice description.
CHECK_DATE	AP_CHECKS_ALL	CHECK_DATE	Payment Date.
INVOICE_DATE	AP_INVOICES_ALL	INVOICE_DATE	Invoice date.
ACCRUAL_POSTED_FLAG	AP_INVOICE_PAYMENTS_ALL	ACCRUAL_POSTED_FLAG	Accrual posted flag.
ACCTS_PAY_ACCOUNT_NUMBER	GL_CODE_COMBINATIONS_KFV	CONCATENATED_SEGMENTS	Flag that indicates whether the payment has been accounted in accrual set of books.
ASSETS_ADDITION_FLAG	AP_INVOICE_PAYMENTS_ALL	ASSETS_ADDITION_FLAG	Flag that indicates whether discounts on invoice payment have been transferred into Oracle Assets.
ASSET_ACCOUNT_NUMBER	GL_CODE_COMBINATIONS_KFV	CONCATENATED_SEGMENTS	Account Number for assets.
BANK_ACCOUNT_NUM	AP_INVOICE_PAYMENTS_ALL	BANK_ACCOUNT_NUM	Bank Account Number.
BANK_ACCOUNT_TYPE	AP_INVOICE_PAYMENTS_ALL	BANK_ACCOUNT_TYPE	Bank Account type.
BANK_NUM	AP_INVOICE_PAYMENTS_ALL	BANK_NUM	Bank number.
CASH_POSTED_FLAG	AP_INVOICE_PAYMENTS_ALL	CASH_POSTED_FLAG	Cash posted flag.
DISCOUNT_LOST	AP_INVOICE_PAYMENTS_ALL	DISCOUNT_LOST	Amount of discount lost.

Document Field	Oracle Applications Table/View Name	Column Name	Description
DISCOUNT_TAKEN	AP_INVOICE_PAYMENTS_ALL	DISCOUNT_TAKEN	Amount of discount taken.
EXCHANGE_DATE	AP_INVOICE_PAYMENTS_ALL	EXCHANGE_DATE	Date for which exchange rate is obtained from daily rates table.
EXCHANGE_RATE	AP_INVOICE_PAYMENTS_ALL	EXCHANGE_RATE	Exchange rate for a foreign currency payment.
EXCHANGE_RATE_TYPE	AP_INVOICE_PAYMENTS_ALL	EXCHANGE_RATE_TYPE	Exchange rate type for a foreign currency payment.
GAIN_ACCOUNT_NUMBER	GL_CODE_COMBINATIONS_KFV	CONCATENATED_SEGMENTS	Account to which realized exchange rate gains are posted.
INVOICE_BASE_AMOUNT	AP_INVOICE_PAYMENTS_ALL		Payment amount in functional currency at invoice's exchange rate, only used for foreign currency invoice payments.
LOSS_ACCOUNT_NUMBER	GL_CODE_COMBINATIONS_KFV	CONCATENATED_SEGMENTS	Account to which realized exchange rate losses are posted.
PAYMENT_BASE_AMOUNT	AP_INVOICE_PAYMENTS_ALL	PAYMENT_BASE_AMOUNT	Payment amount at payment's exchange rate, only used for foreign currency invoice payments.
PAYMENT_NUM	AP_INVOICE_PAYMENTS_ALL	PAYMENT_NUM	Payment number.
PERIOD_NAME	AP_INVOICE_PAYMENTS_ALL	PERIOD_NAME	Period name.
POSTED_FLAG	AP_INVOICE_PAYMENTS_ALL	POSTED_FLAG	Flag that indicates if the payment has been accounted (Y or N).
SET_OF_BOOKS_NAME	GL_SETS_OF_BOOKS	NAME	GL set of books.
INVOICE_PAYMENT_TYPE	AP_INVOICE_PAYMENTS_ALL	INVOICE_PAYMENT_TYPE	Creation method of a payment, used by Positive Pay feature (Single or Batch).

Document Field	Oracle Applications Table/View Name	Column Name	Description
OTHER_INVOICE_NUM	AP_INVOICES_ALL	OTHER_INVOICE_NUM	Related invoice number.
CHECK_NUMBER	AP_CHECKS_ALL	CHECK_NUMBER	Check Number.
CHECK_AMOUNT	AP_CHECKS_ALL	AMOUNT	Check Amount.
CHECK_TYPE	AP_LOOKUP_CODES	DISPLAYED_FIELD	Check Type.
CHECK_STATUS	AP_LOOKUP_CODES	DISPLAYED_FIELD	Check Status.
PAYMENT_TYPE	AP_LOOKUP_CODES	DISPLAYED_FIELD	Payment type.
INVOICE_AMOUNT	AP_INVOICES_ALL	INVOICE_AMOUNT	Invoice Amount.
AMOUNT_PAID	AP_INVOICES_ALL	AMOUNT_PAID	Amount paid.
EXTERNAL_BANK_ACCOUNT_NUMBER	AP_BANK_ACCOUNTS_ALL	BANK_ACCOUNT_NUM	External bank account number.

Send Accounts Receivable Transactions Service

The name of this service is:

WmOAFIN107SC.receivables107SC.fromOA.ARTransaction:sendARTransaction

This service retrieves the Accounts Receivable Transactions.

The following business rules apply for retrieving AR Transaction data from Oracle Applications:

- Picks up new and changed Accounts Receivable Transaction records only.
- Picks up complete Transactions only.
- No deletion of Transactions defined in Oracle Applications. Therefore, a business document with document status of INSERT or UPDATE will be created. The document status of DELETE does not apply in this case.

Database Scripts

This service uses the following database scripts:

Database Script	Description
wm_install_from_artrans.sql	Runs all the scripts listed below, except the uninstall script.
wm_from_artrans_vw.sql	Creates the following required view components: <ul style="list-style-type: none"> ■ WM_RA_CUSTOMER_TRX_VW ■ WM_RA_CUST_TRX_LINE_GL_DIST_VW ■ WM_RA_CUST_TRX_FREIGHT_VW ■ WM_AR_NOTES_VW ■ WM_RA_CUST_TRX_CHARGES_VW ■ WM_RA_CUSTOMER_TRX_LINES_VW ■ WM_RA_CUST_TRX_FREIGHTLINES_VW ■ WM_RA_CUST_TRX_TAXLINES_VW ■ WM_RA_SALESREPS_LINES_VW ■ WM_RA_SALESREPS_DEFAULT_VW ■ WM_RA_CUSTOMER_TRX_QRY_VW
wm_from_artrans_trg.sql	Creates the following trigger component to be used for writing to the WM_TRACKCHANGES table to indicate a new or updated document: <ul style="list-style-type: none"> ■ WM_RA_CUSTOMER_TRX_ALL_IU_TRG
wm_from_artrans_trg.sql	Creates the following trigger component to be used for writing to the WM_TRACKCHANGES table to indicate a new or updated document: <ul style="list-style-type: none"> ■ WM_RA_CUSTOMER_TRX_ALL_IU_TRG
wm_disable_from_artrans.sql	Disables the triggers installed by wm_from_artrans_trg.sql.
wm_enable_from_artrans.sql	Re-enables the triggers installed by wm_from_artrans_trg.sql.
wm_drop_from_artrans.sql	Uninstalls all components created by wm_install_from_artrans.sql.

Supporting Transaction Definition

This service was built from the configured services defined in the following transaction definition:

- `getARTransactionTxn107SC.txp`

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow executes as follows:

- `specifyDefaultSettings` specifies the default parameter settings required for service execution. You should change these settings accordingly.
- `lockTxnCtrl` determines whether the `sendARTransaction` service is ready for execution. That is, it determines whether there is no other instance of this service already in process.
 - If `lockTxnCtrl` returns `False`, it means that another instance of this service is already in process. The service exits, and waits for the next scheduled execution.
 - If `lockTxnCtrl` returns `True`, it means that the service is ready to execute. The AR Transaction row in the control table is locked and updated so that the status is changed to be `INPROCESS`. This prevents any other AR Transaction service from executing.
- `getARTransactionTxn` service queries the Oracle Applications database for any AR Transaction. The number of records returned depends on the parameter value specified in `noOfRowsToFetch` parameter in the `specifyDefaultSettings` map.
- `processBizDoc` sends the business document to a trading partner. You must customize this step to receive a Success or an Error status of the document transfer. The document transfer status information should be sent back to the calling service (`sendARTransaction`). The document transfer status passed back should have the document identifiers along with the transfer status of `SUCCESS` or `ERROR` and Error information for the particular document which has not been transferred successfully to the trading partner.
- Loop against each document. Based on the Debug Mode specified during execution, it either purges or updates the records in the `WM_TRACKCHANGES` custom table.
 - If the Debug Mode is `TRUE`, the records in the `WM_TRACKCHANGES` table are updated and the `PROCESSED_FLAG` is set to `Y`. This ensures that same sets of records are not picked up during next polling interval. The `updateTrackChanges` service updates the `PROCESSED_FLAG` in the `WM_TRACKCHANGES` table to `Y` and updates `PROCESSED_DATE` to `sysdate` so that same information is not picked up again during next polling instance.

- If the Debug Mode is FALSE, the records in the WM_TRACKCHANGES table are deleted. The `purgeTrackChanges` service purges the records from the WM_TRACKCHANGES table.
- Loop against each document and based on the Transfer Status, `insertTransferERRInfo` inserts a new record in the WM_TRACKCHANGES table so that same document can be picked up during the next polling interval.
- `unlockTxnCtrl` releases the lock on the Custom Control table so that next polling instance of `sendVendor` service can begin.
- `getLastError` logs any flow service execution errors in the above steps.

For more details on send transactions, see [“Overview of Send Service Transaction Processing” on page 38](#).

Business Document Structure

This service uses the following business document structure:

- 1.0 TRANSACTIONS
 - 1.1 TRANSACTION_LINES
 - 1.1.1 TAX_LINES
 - 1.1.2 FREIGHT_LINES
 - 1.1.3 SALES_CREDITS
 - 1.1.4 GL_DISTRIBUTIONS
 - 1.2 CHARGES
 - ***1.2.1 GL_DISTRIBUTIONS
 - 1.3 FREIGHT_LINES
 - ***1.3.1 GL_DISTRIBUTIONS
 - 1.4 **DEFAULT_SALES_CREDIT
 - 1.5 **NOTES
 - 1.6*** GL_DISTRIBUTIONS



Note: ** Default Sales Credit and Notes are not currently used.



Note: *** All GL Distributions use the same table. See [“1.1.4 GL_DISTRIBUTIONS” on page 169](#).

1.0 TRANSACTIONS

Document Field	Oracle Applications Table/View Name	Column Name	Description
WEB_TRANSACTION_ID			
DOCUMENT_TYPE			The value is ARTRANSACTION.
DOCUMENT_STATUS			The value is UPDATE or INSERT.
CUSTOMER_TRX_ID	RA_CUSTOMER_TRX_ALL	CUSTOMER_TRX_ID	Customer transaction ID.
AGREEMENT_NAME	RA_CUSTOMER_TRX_ALL	AGREEMENT_ID	The name of the customer agreement for this transaction.
COMMENTS	RA_CUSTOMER_TRX_ALL	COMMENTS	Comments about this transaction.
CONS_BILLING_NUM	AR_CONS_INV_ALL	CONS_BILLING_NUM	The number for this consolidated bill. A consolidated bill number is used for grouping a set of invoices under one bill.
CONVERSION_DATE	RA_CUSTOMER_TRX_ALL	CONVERSION_DATE	The exchange rate date for this transaction.
CONVERSION_RATE	RA_CUSTOMER_TRX_ALL	CONVERSION_RATE	The exchange rate for this transaction.
CONVERSION_TYPE	RA_CUSTOMER_TRX_ALL	CONVERSION_TYPE	The exchange rate type for this transaction.
CREDIT_METHOD_FOR_RULES	RA_CUSTOMER_TRX_ALL	CREDIT_METHOD_FOR_RULES	The credit method for crediting a transaction, which uses an accounting rule.
CREDIT_METHOD_FOR_INSTALLMENTS	RA_CUSTOMER_TRX_ALL	CREDIT_METHOD_FOR_INSTALLMENTS	The credit method for crediting a transaction that uses split payment terms.
INVOICE_CURRENCY_CODE	RA_CUSTOMER_TRX_ALL	INVOICE_CURRENCY_CODE	The currency code for this transaction.

Document Field	Oracle Applications Table/View Name	Column Name	Description
CUSTOMER_BANK_ACCOUNT_NAME	RA_CUSTOMER_TRX_ALL	CUSTOMER_BANK_ACCOUNT_ID	The Bill-to customer bank account name for this transaction.
TRX_TYPE_NAME	RA_CUSTOMER_TRX_ALL	CUST_TRX_TYPE_ID	The Transaction Type name for this transaction.
DOCUMENT_NUMBER	RA_CUSTOMER_TRX_ALL	TRX_NUMBER	The Document Number for this transaction.
GL_DATE	RA_CUST_TRX_LINE_GL_DIST_ALL	GL_DATE	The General Ledger date for this transaction. The GL date determines the accounting period that you record this transaction to your general ledger.
HEADER_ATTRIBUTE_UPDATE	RA_CUSTOMER_TRX_ALL	HEADER_ATTRIBUTE_CATEGORY	Descriptive flexfield attribute information for the transaction information flexfield. Descriptive flexfield attributes let you store additional columns, the contents of which you define.
HEADER_ATTRIBUTE1		HEADER_ATTRIBUTE1	
HEADER_ATTRIBUTE2		HEADER_ATTRIBUTE2	
HEADER_ATTRIBUTE3		HEADER_ATTRIBUTE3	
HEADER_ATTRIBUTE4		HEADER_ATTRIBUTE4	
HEADER_ATTRIBUTE5		HEADER_ATTRIBUTE5	
HEADER_ATTRIBUTE6		HEADER_ATTRIBUTE6	
HEADER_ATTRIBUTE7		HEADER_ATTRIBUTE7	
HEADER_ATTRIBUTE8		HEADER_ATTRIBUTE8	

Document Field	Oracle Applications Table/View Name	Column Name	Description
HEADER_ATTRIBUTE9		HEADER_ATTRIBUTE9	
HEADER_ATTRIBUTE10		HEADER_ATTRIBUTE10	
HEADER_ATTRIBUTE11		HEADER_ATTRIBUTE11	
HEADER_ATTRIBUTE12		HEADER_ATTRIBUTE12	
HEADER_ATTRIBUTE13		HEADER_ATTRIBUTE13	
HEADER_ATTRIBUTE14		HEADER_ATTRIBUTE14	
HEADER_ATTRIBUTE15		HEADER_ATTRIBUTE15	
INTERNAL_NOTES	RA_CUSTOMER_TRX_ALL	INTERNAL_NOTES	Internal notes for this transaction.
INVOICING_RULE_NAME	RA_CUSTOMER_TRX_ALL	INVOICING_RULE_ID	The invoicing rule name for this transaction.
ORIG_SYSTEM_BILL_ADDRESS_REF	RA_CUSTOMER_TRX_ALL	ORIG_SYSTEM_BILL_ADDRESS_REF	The Bill-To customer address reference.
ORIG_SYSTEM_BILL_CONTACT_REF	RA_CUSTOMER_TRX_ALL	ORIG_SYSTEM_BILL_CONTACT_REF	The Bill-to contact reference.
ORIG_SYSTEM_BILL_CUSTOMER_REF	RA_CUSTOMER_TRX_ALL	ORIG_SYSTEM_BILL_CUSTOMER_REF	Bill-to customer.
ORIG_SYSTEM_SHIP_ADDRESS_REF	RA_CUSTOMER_TRX_ALL	ORIG_SYSTEM_SHIP_ADDRESS_REF	Ship-to customer address.
ORIG_SYSTEM_SHIP_CONTACT_REF	RA_CUSTOMER_TRX_ALL	ORIG_SYSTEM_SHIP_CONTACT_REF	Ship-to contact.
ORIG_SYSTEM_SHIP_CUSTOMER_REF	RA_CUSTOMER_TRX_ALL	ORIG_SYSTEM_SHIP_CUSTOMER_REF	Ship-to customer.

Document Field	Oracle Applications Table/View Name	Column Name	Description
ORIG_SYSTEM_SOLD_CUSTOMER_REF	RA_CUSTOMER_TRX_ALL	ORIG_SYSTEM_SOLD_CUSTOMER_REF	Sold-to customer.
ORIG_SYSTEM_BATCH_NAME	RA_CUSTOMER_TRX_ALL	ORIG_SYSTEM_BATCH_NAME	The batch name for this transaction.
PRIMARY_SALESREP_NUMBER	RA_CUSTOMER_TRX_ALL	PRIMARY_SALESREP_ID	The primary salesperson number for this transaction.
PRINTING_OPTION	RA_CUSTOMER_TRX_ALL	PRINTING_OPTION	The printing option for this transaction.
PURCHASE_ORDER	RA_CUSTOMER_TRX_ALL	PURCHASE_ORDER	The purchase order number for this transaction.
PURCHASE_ORDER_REVISION	RA_CUSTOMER_TRX_ALL	PURCHASE_ORDER_REVISION	The purchase order revision for this transaction.
PURCHASE_ORDER_DATE	RA_CUSTOMER_TRX_ALL	PURCHASE_ORDER_DATE	The date of the purchase order for this transaction.
REASON_CODE	RA_CUSTOMER_TRX_ALL	REASON_CODE	The reason code for this transaction.
RECEIPT_METHOD_NAME	RA_CUSTOMER_TRX_ALL	RECEIPT_METHOD_ID	The name of the payment method for this transaction.
RELATED_CUSTOMER_TRX_NUMBER	RA_CUSTOMER_TRX_ALL	RELATED_CUSTOMER_TRX_ID	The document number to which this transaction is related.
SET_OF_BOOKS_NAME	RA_CUSTOMER_TRX_ALL	SET_OF_BOOKS_ID	Maps to the Set of Books ID for this transaction.
TERRITORY	RA_CUSTOMER_TRX_ALL	TERRITORY_ID	Maps to the Territory ID for this transaction.
TERM_NAME	RA_CUSTOMER_TRX_ALL	TERM_ID	The name of the Payment Term for this transaction.
TRX_DATE	RA_CUSTOMER_TRX_ALL	TRX_DATE	The Transaction Date for this transaction.
TRX_NUMBER	RA_CUSTOMER_TRX_ALL	TRX_NUMBER	The number for this transaction.

Document Field	Oracle Applications Table/View Name	Column Name	Description
ATTRIBUTE_CATEGORY	RA_CUSTOMER_TRX_ALL	ATTRIBUTE_CATEGORY	The Invoice Line Information flexfield attribute information for this transaction. Descriptive flexfield attributes allow you to store additional columns, the contents of which you define.
ATTRIBUTE1		ATTRIBUTE1	
ATTRIBUTE8		ATTRIBUTE8	
ATTRIBUTE2		ATTRIBUTE2	
ATTRIBUTE3		ATTRIBUTE3	
ATTRIBUTE4		ATTRIBUTE4	
ATTRIBUTE5		ATTRIBUTE5	
ATTRIBUTE6		ATTRIBUTE6	
ATTRIBUTE7		ATTRIBUTE7	
ATTRIBUTE9		ATTRIBUTE9	
ATTRIBUTE10		ATTRIBUTE10	
ATTRIBUTE11		ATTRIBUTE11	
ATTRIBUTE12		ATTRIBUTE12	
ATTRIBUTE13		ATTRIBUTE13	
ATTRIBUTE14		ATTRIBUTE14	
ATTRIBUTE15		ATTRIBUTE15	
RELATED_BATCH_SOURCE_NAME	RA_CUSTOMER_TRX_ALL	BATCH_SOURCE_ID	The name of the Batch Source of the document to which this transaction is related.
BATCH_SOURCE_NAME	RA_CUSTOMER_TRX_ALL	BATCH_SOURCE_ID	The name of the Batch Source for this transaction. AutoInvoice uses your batch source to determine your transaction and batch numbering method and your AutoInvoice processing options.

Document Field	Oracle Applications Table/View Name	Column Name	Description
FOB_POINT	RA_CUSTOMER_TRX_ALL	FOB_POINT	The FOB point for this transaction.
SHIP_DATE_ACTUAL	RA_CUSTOMER_TRX_ALL	SHIP_DATE_ACTUAL	The shipment date for this transaction.
SHIP_VIA	RA_CUSTOMER_TRX_ALL	SHIP_VIA	The ship via code for this transaction.
WAYBILL_NUMBER	RA_CUSTOMER_TRX_ALL	WAYBILL_NUMBER	The waybill number for this transaction.
DEFAULT_USSGL_TRANSACTION_CODE	RA_CUSTOMER_TRX_ALL	DEFAULT_USSGL_TRANSACTION_CODE	The transaction code for this transaction.
DEFAULT_USSGL_TRX_CODE_CONTEXT	RA_CUSTOMER_TRX_ALL	DEFAULT_USSGL_TRX_CODE_CONTEXT	AutoInvoice does not currently use column.
ORGANIZATION_NAME	RA_CUSTOMER_TRX_ALL	ORG_ID	Currently not used. ORG_ID has been added for future Oracle Applications functionality.

1.1 TRANSACTION_LINES

Document Field	Oracle Applications Table/View Name	Column Name	Description
CUSTOMER_TRX_ID	RA_CUSTOMER_TRX_LINES_ALL	CUSTOMER_TRX_ID	Customer Transaction ID.
CUSTOMER_TRX_LINE_ID	RA_CUSTOMER_TRX_LINES_ALL	CUSTOMER_TRX_LINE_ID	Customer Transaction Line ID.
DESCRIPTION	RA_CUSTOMER_TRX_LINES_ALL	DESCRIPTION	Description for this transaction.
QUANTITY	RA_CUSTOMER_TRX_LINES_ALL	QUANTITY_INVOICED	Line Quantity.
QUANTITY_ORDERED	RA_CUSTOMER_TRX_LINES_ALL	QUANTITY_ORDERED	The original number of units ordered for this transaction.
UNIT_STANDARD_PRICE	RA_CUSTOMER_TRX_LINES_ALL	UNIT_STANDARD_PRICE	The standard price per unit for this transaction.

Document Field	Oracle Applications Table/View Name	Column Name	Description
UNIT_SELLING_PRICE	RA_CUSTOMER_TRX_LINES_ALL	UNIT_SELLING_PRICE	The selling price per unit for this transaction.
AMOUNT	RA_CUSTOMER_TRX_LINES_ALL	REVENUE_AMOUNT	The revenue amount for this transaction.
UOM_CODE	RA_CUSTOMER_TRX_LINES_ALL	UOM_CODE	The unit of measure code for this transaction.
UOM_NAME	RA_CUSTOMER_TRX_LINES_ALL	UOM_CODE	The unit of measure name for this transaction.
ACCOUNTING_RULE_DURATION	RA_CUSTOMER_TRX_LINES_ALL	ACCOUNTING_RULE_DURATION	The accounting rule duration for this transaction.
ACCOUNTING_RULE_NAME	RA_CUSTOMER_TRX_LINES_ALL	ACCOUNTING_RULE_ID	The accounting rule name for this transaction.
RULE_START_DATE	RA_CUSTOMER_TRX_LINES_ALL	RULE_START_DATE	The date that you want to start the accounting rule for this transaction.
LAST_PERIOD_TO_CREDIT	RA_CUSTOMER_TRX_LINES_ALL	LAST_PERIOD_TO_CREDIT	For unit credit memos, the last period number from which you want to start crediting.
INVENTORY_ITEM	RA_CUSTOMER_TRX_LINES_ALL	INVENTORY_ID	The concatenated Inventory flexfield, which is derived based on the INVENTORY_ID for this transaction.
MEMO_LINE_NAME	RA_CUSTOMER_TRX_LINES_ALL	MEMO_LINE_ID	The name of the standard memo line for this transaction.
TAX_EXEMPT_FLAG	RA_CUSTOMER_TRX_LINES_ALL	TAX_EXEMPT_FLAG	If LINE_TYPE = 'LINE', this column is optional. The value controls how a line is taxed.
TAX_EXEMPT_NUMBER	RA_CUSTOMER_TRX_LINES_ALL	TAX_EXEMPT_NUMBER	The Tax Exempt Number for this transaction.
TAX_EXEMPT_REASON_CODE	RA_CUSTOMER_TRX_LINES_ALL	TAX_EXEMPT_REASON_CODE	The Tax Exempt Reason code for this transaction.

Document Field	Oracle Applications Table/View Name	Column Name	Description
TAX_EXEMPT_REASON_CODE_MEANING	RA_CUSTOMER_TRX_LINES_ALL	TAX_EXEMPT_REASON_CODE	The Tax Exempt Reason code meaning for this transaction.
SALES_ORDER_SOURCE	RA_CUSTOMER_TRX_LINES_ALL	SALES_ORDER_SOURCE	The source of the Sales Order for this transaction.
SALES_ORDER	RA_CUSTOMER_TRX_LINES_ALL	SALES_ORDER	The Sales Order Number for this transaction.
SALES_ORDER_REVISION	RA_CUSTOMER_TRX_LINES_ALL	SALES_ORDER_REVISION	The Sales Order Revision for this transaction.
SALES_ORDER_LINE	RA_CUSTOMER_TRX_LINES_ALL	SALES_ORDER_LINE	The Sales Order Line Number for this transaction.
SALES_ORDER_DATE	RA_CUSTOMER_TRX_LINES_ALL	SALES_ORDER_DATE	The date of the Sales Order for this transaction.
WAREHOUSE_NAME	RA_CUSTOMER_TRX_LINES_ALL	WAREHOUSE_ID	This column identifies the Ship-from Location and can be used to control taxation.
INTERFACE_LINE_CONTEXT	RA_CUSTOMER_TRX_LINES_ALL	INTERFACE_LINE_CONTEXT	The line transaction flexfield for this transaction. The line transaction flexfield is a combination of attribute values that you use to uniquely identify this transaction line in your original system.
INTERFACE_LINE_ATTRIBUTE1		INTERFACE_LINE_ATTRIBUTE1	
INTERFACE_LINE_ATTRIBUTE2		INTERFACE_LINE_ATTRIBUTE2	
INTERFACE_LINE_ATTRIBUTE3		INTERFACE_LINE_ATTRIBUTE3	
INTERFACE_LINE_ATTRIBUTE4		INTERFACE_LINE_ATTRIBUTE4	
INTERFACE_LINE_ATTRIBUTE5		INTERFACE_LINE_ATTRIBUTE5	

Document Field	Oracle Applications Table/View Name	Column Name	Description
INTERFACE_LINE_ATTRIBUTE6		INTERFACE_LINE_ATTRIBUTE6	
INTERFACE_LINE_ATTRIBUTE7		INTERFACE_LINE_ATTRIBUTE7	
INTERFACE_LINE_ATTRIBUTE8		INTERFACE_LINE_ATTRIBUTE8	
INTERFACE_LINE_ATTRIBUTE9		INTERFACE_LINE_ATTRIBUTE9	
INTERFACE_LINE_ATTRIBUTE10		INTERFACE_LINE_ATTRIBUTE10	
INTERFACE_LINE_ATTRIBUTE11		INTERFACE_LINE_ATTRIBUTE11	
INTERFACE_LINE_ATTRIBUTE13		INTERFACE_LINE_ATTRIBUTE13	
INTERFACE_LINE_ATTRIBUTE15		INTERFACE_LINE_ATTRIBUTE15	
INTERFACE_LINE_ATTRIBUTE12		INTERFACE_LINE_ATTRIBUTE12	
INTERFACE_LINE_ATTRIBUTE14		INTERFACE_LINE_ATTRIBUTE14	
REFERENCE_LINE_CONTEXT	RA_CUSTOMER_TRX_LINES_ALL	PREVIOUS_CUSTOMER_TRX_LINE_ID/INITIAL_CUSTOMER_TRX_LINE_ID	The transaction flexfield of the transaction line you are crediting in these columns.
REFERENCE_LINE_ATTRIBUTE1			
REFERENCE_LINE_ATTRIBUTE2			
REFERENCE_LINE_ATTRIBUTE3			
REFERENCE_LINE_ATTRIBUTE4			
REFERENCE_LINE_ATTRIBUTE5			

Document Field	Oracle Applications Table/View Name	Column Name	Description
REFERENCE_LINE_ATTRIBUTE6			
REFERENCE_LINE_ATTRIBUTE7			
REFERENCE_LINE_ATTRIBUTE8			
REFERENCE_LINE_ATTRIBUTE9			
REFERENCE_LINE_ATTRIBUTE10			
REFERENCE_LINE_ATTRIBUTE11			
REFERENCE_LINE_ATTRIBUTE12			
REFERENCE_LINE_ATTRIBUTE13			
REFERENCE_LINE_ATTRIBUTE14			
REFERENCE_LINE_ATTRIBUTE15			
LINK_TO_LINE_CONTEXT	RA_CUSTOMER_TRX_LINES_ALL	LINK_TO_CUST_TRX_LINE_ID	The link to your transaction flexfield attribute values.
LINK_TO_LINE_ATTRIBUTE1			
LINK_TO_LINE_ATTRIBUTE2			
LINK_TO_LINE_ATTRIBUTE3			
LINK_TO_LINE_ATTRIBUTE4			
LINK_TO_LINE_ATTRIBUTE5			
LINK_TO_LINE_ATTRIBUTE6			

Document Field	Oracle Applications Table/View Name	Column Name	Description
LINK_TO_LINE_ATTRIBUTE7			
LINK_TO_LINE_ATTRIBUTE8			
LINK_TO_LINE_ATTRIBUTE9			
LINK_TO_LINE_ATTRIBUTE10			
LINK_TO_LINE_ATTRIBUTE11			
LINK_TO_LINE_ATTRIBUTE12			
LINK_TO_LINE_ATTRIBUTE13			
LINK_TO_LINE_ATTRIBUTE14			
LINK_TO_LINE_ATTRIBUTE15			

1.1.1 TAX_LINES

Document Field	Oracle Applications Table/View Name	Column Name	Description
CUSTOMER_TRX_ID	RA_CUSTOMER_TRX_LINES_ALL	CUSTOMER_TRX_ID	Customer Transaction ID.
CUSTOMER_TRX_LINE_ID	RA_CUSTOMER_TRX_LINES_ALL	CUSTOMER_TRX_LINE_ID	Customer Transaction Line ID.
LINK_TO_CUST_TRX_LINE_ID	RA_CUSTOMER_TRX_LINES_ALL	LINK_TO_CUST_TRX_LINE_ID	Link to Customer Transaction Line ID.
AMOUNT	RA_CUSTOMER_TRX_LINES_ALL	REVENUE_AMOUNT	The revenue amount for this transaction.
VAT_TAX_CODE	RA_CUSTOMER_TRX_LINES_ALL	SALES_TAX_ID	The Tax Code for this Tax Line.

Document Field	Oracle Applications Table/View Name	Column Name	Description
TAX_PRECEDENCE	RA_CUSTOMER_TRX_LINES_ALL	TAX_PRECEDENCE	The precedence number for this tax line. This column is used to compute tax compounding.
TAX_RATE	RA_CUSTOMER_TRX_LINES_ALL	TAX_RATE	The Tax Rate for this Tax Line.
INTERFACE_LINE_CONTEXT	RA_CUSTOMER_TRX_LINES_ALL	LINK_TO_CUST_TRX_LINE_ID	The Line Transaction flexfield for this transaction.
INTERFACE_LINE_ATTRIBUTE1			
INTERFACE_LINE_ATTRIBUTE2			
INTERFACE_LINE_ATTRIBUTE3			
INTERFACE_LINE_ATTRIBUTE4			
INTERFACE_LINE_ATTRIBUTE5			
INTERFACE_LINE_ATTRIBUTE6			
INTERFACE_LINE_ATTRIBUTE7			
INTERFACE_LINE_ATTRIBUTE8			
INTERFACE_LINE_ATTRIBUTE9			
INTERFACE_LINE_ATTRIBUTE10			
INTERFACE_LINE_ATTRIBUTE11			
INTERFACE_LINE_ATTRIBUTE12			
INTERFACE_LINE_ATTRIBUTE13			

Document Field	Oracle Applications Table/View Name	Column Name	Description
INTERFACE_LINE_ATTRIBUTE14			
INTERFACE_LINE_ATTRIBUTE15			
ATTRIBUTE_CATEGORY	RA_CUSTOMER_TRX_LINES_ALL	ATTRIBUTE_CATEGORY	The Invoice Line Information flexfield attribute information for this transaction. Descriptive flexfield attributes allow you to store additional columns, the contents of which you define.
ATTRIBUTE1		ATTRIBUTE1	
ATTRIBUTE2		ATTRIBUTE2	
ATTRIBUTE3		ATTRIBUTE3	
ATTRIBUTE4		ATTRIBUTE4	
ATTRIBUTE5		ATTRIBUTE5	
ATTRIBUTE6		ATTRIBUTE6	
ATTRIBUTE7		ATTRIBUTE7	
ATTRIBUTE8		ATTRIBUTE8	
ATTRIBUTE9		ATTRIBUTE9	
ATTRIBUTE10		ATTRIBUTE10	
ATTRIBUTE11		ATTRIBUTE11	
ATTRIBUTE12		ATTRIBUTE12	
ATTRIBUTE13		ATTRIBUTE13	
ATTRIBUTE14		ATTRIBUTE14	
ATTRIBUTE15		ATTRIBUTE15	

1.1.2 FREIGHT_LINES

Document Field	Oracle Applications Table/View Name	Column Name	Description
CUSTOMER_TRX_ID	RA_CUSTOMER_TRX_LINES_ALL	CUSTOMER_TRX_ID	Customer transaction ID.
CUSTOMER_TRX_LINE_ID	RA_CUSTOMER_TRX_LINES_ALL	CUSTOMER_TRX_LINE_ID	Customer Transaction Line ID.
LINK_TO_CUST_TRX_LINE_ID	RA_CUSTOMER_TRX_LINES_ALL	LINK_TO_CUST_TRX_LINE_ID	Link to Customer Transaction Line ID.
REVENUE_AMOUNT	RA_CUSTOMER_TRX_LINES_ALL	REVENUE_AMOUNT	The revenue amount for this transaction.
INTERFACE_LINE_CONTEXT	RA_CUSTOMER_TRX_LINES_ALL	LINK_TO_CUST_TRX_LINE_ID	The line transaction flexfield for this transaction.
INTERFACE_LINE_ATTRIBUTE1			
INTERFACE_LINE_ATTRIBUTE2			
INTERFACE_LINE_ATTRIBUTE3			
INTERFACE_LINE_ATTRIBUTE4			
INTERFACE_LINE_ATTRIBUTE5			
INTERFACE_LINE_ATTRIBUTE6			
INTERFACE_LINE_ATTRIBUTE7			
INTERFACE_LINE_ATTRIBUTE8			
INTERFACE_LINE_ATTRIBUTE9			
INTERFACE_LINE_ATTRIBUTE10			
INTERFACE_LINE_ATTRIBUTE11			
INTERFACE_LINE_ATTRIBUTE12			
INTERFACE_LINE_ATTRIBUTE13			
INTERFACE_LINE_ATTRIBUTE14			
INTERFACE_LINE_ATTRIBUTE15			

Document Field	Oracle Applications Table/View Name	Column Name	Description
ATTRIBUTE_CATEGORY	RA_CUSTOMER_TRX_LINES_ALL	ATTRIBUTE_CATEGORY	The Invoice Line Information flexfield attribute information for this transaction.
ATTRIBUTE1		ATTRIBUTE1	
ATTRIBUTE2		ATTRIBUTE2	
ATTRIBUTE3		ATTRIBUTE3	
ATTRIBUTE4		ATTRIBUTE4	
ATTRIBUTE5		ATTRIBUTE5	
ATTRIBUTE6		ATTRIBUTE6	
ATTRIBUTE7		ATTRIBUTE7	
ATTRIBUTE8		ATTRIBUTE8	
ATTRIBUTE9		ATTRIBUTE9	
ATTRIBUTE10		ATTRIBUTE10	
ATTRIBUTE11		ATTRIBUTE11	
ATTRIBUTE12		ATTRIBUTE12	
ATTRIBUTE13		ATTRIBUTE13	
ATTRIBUTE14		ATTRIBUTE14	
ATTRIBUTE15		ATTRIBUTE15	

1.2.3 SALES_CREDITS

Document Field	Oracle Applications Table/View Name	Column Name	Description
CUSTOMER_TRX_ID	RA_CUST_TRX_LINE_SALESREPS_ALL	CUSTOMER_TRX_ID	Customer transaction ID.
CUSTOMER_TRX_LINE_ID	RA_CUST_TRX_LINE_SALESREPS_ALL	CUSTOMER_TRX_LINE_ID	Customer transaction line ID.
CUST_TRX_LINE_SALESREP_ID	RA_CUST_TRX_LINE_SALESREPS_ALL	CUST_TRX_LINE_SALESREP_ID	
SALESREP_NUMBER	RA_CUST_TRX_LINE_SALESREPS_ALL	SALESREP_ID	The salesperson number for this sales credit assignment.
REVENUE_PERCENT_SPLIT	RA_CUST_TRX_LINE_SALESREPS_ALL	REVENUE_PERCENT_SPLIT	The sales credit percent for this salesperson.
REVENUE_AMOUNT_SPLIT	RA_CUST_TRX_LINE_SALESREPS_ALL	REVENUE_AMOUNT_SPLIT	The sales credit amount for this salesperson.
NON_REVENUE_PERCENT_SPLIT	RA_CUST_TRX_LINE_SALESREPS_ALL	NON_REVENUE_PERCENT_SPLIT	Not used.
NON_REVENUE_AMOUNT_SPLIT	RA_CUST_TRX_LINE_SALESREPS_ALL	NON_REVENUE_AMOUNT_SPLIT	Not used.
SALESREP_NAME	RA_CUST_TRX_LINE_SALESREPS_ALL	SALESREP_ID	Sales representative name.
SALES_CREDIT_TYPE_NAME	RA_CUST_TRX_LINE_SALESREPS_ALL	SALESREP_ID	The name of the sales credit type for this sales credit assignment.
INTERFACE_LINE_CONTEXT	RA_CUST_TRX_LINE_SALESREPS_ALL	CUSTOMER_TRX_LINE_ID	The line transaction flexfield for this transaction.
INTERFACE_LINE_ATTRIBUTE1			
INTERFACE_LINE_ATTRIBUTE2			
INTERFACE_LINE_ATTRIBUTE3			
INTERFACE_LINE_ATTRIBUTE4			

Document Field	Oracle Applications Table/View Name	Column Name	Description
INTERFACE_LINE_ATTRIBUTE5			
INTERFACE_LINE_ATTRIBUTE6			
INTERFACE_LINE_ATTRIBUTE7			
INTERFACE_LINE_ATTRIBUTE8			
INTERFACE_LINE_ATTRIBUTE9			
INTERFACE_LINE_ATTRIBUTE10			
INTERFACE_LINE_ATTRIBUTE11			
INTERFACE_LINE_ATTRIBUTE12			
INTERFACE_LINE_ATTRIBUTE13			
INTERFACE_LINE_ATTRIBUTE14			
INTERFACE_LINE_ATTRIBUTE15			
ATTRIBUTE_CATEGORY	RA_CUST_TRX_LINE_SALESREPS_ALL	ATTRIBUTE_CATEGORY	The Invoice Line information flexfield attribute information for this transaction. Descriptive flexfield attributes allow you to store additional columns, the contents of which you define.
ATTRIBUTE1		ATTRIBUTE1	
ATTRIBUTE2		ATTRIBUTE2	
ATTRIBUTE3		ATTRIBUTE3	
ATTRIBUTE4		ATTRIBUTE4	
ATTRIBUTE5		ATTRIBUTE5	

Document Field	Oracle Applications Table/View Name	Column Name	Description
ATTRIBUTE6		ATTRIBUTE6	
ATTRIBUTE7		ATTRIBUTE7	
ATTRIBUTE8		ATTRIBUTE8	
ATTRIBUTE9		ATTRIBUTE9	
ATTRIBUTE10		ATTRIBUTE10	
ATTRIBUTE11		ATTRIBUTE11	
ATTRIBUTE12		ATTRIBUTE12	
ATTRIBUTE13		ATTRIBUTE13	
ATTRIBUTE14		ATTRIBUTE14	
ATTRIBUTE15		ATTRIBUTE15	

1.1.4 GL_DISTRIBUTIONS



Note: All GL Distributions business document structures use the following table.

Document Field	Oracle Applications Table/View Name	Column Name	Description
CUSTOMER_TRX_LINE_ID	RA_CUST_TRX_LINE_GL_DIST_ALL	CUSTOMER_TRX_LINE_ID	Customer Transaction Line ID.
ACCOUNT_CLASS	RA_CUST_TRX_LINE_GL_DIST_ALL	ACCOUNT_CLASSES	The account class for this accounting distribution.
ACCTD_AMOUNT	RA_CUST_TRX_LINE_GL_DIST_ALL	ACCTD_AMOUNT	The accounted amount for this distribution.
AMOUNT	RA_CUST_TRX_LINE_GL_DIST_ALL	AMOUNT	The amount for this accounting distribution.
ATTRIBUTE_CATEGORY	RA_CUST_TRX_LINE_GL_DIST_ALL	ATTRIBUTE_CATEGORY	The invoice line information flexfield attribute information for this transaction. descriptive flexfield attributes allow you to store additional columns, the contents of which you define.
ATTRIBUTE1		ATTRIBUTE1	
ATTRIBUTE2		ATTRIBUTE2	

Document Field	Oracle Applications Table/View Name	Column Name	Description
ATTRIBUTE3		ATTRIBUTE3	
ATTRIBUTE4		ATTRIBUTE4	
ATTRIBUTE5		ATTRIBUTE5	
ATTRIBUTE6		ATTRIBUTE6	
ATTRIBUTE7		ATTRIBUTE7	
ATTRIBUTE8		ATTRIBUTE8	
ATTRIBUTE9		ATTRIBUTE9	
ATTRIBUTE10		ATTRIBUTE10	
ATTRIBUTE11		ATTRIBUTE11	
ATTRIBUTE12		ATTRIBUTE12	
ATTRIBUTE13		ATTRIBUTE13	
ATTRIBUTE14		ATTRIBUTE14	
ATTRIBUTE15		ATTRIBUTE15	
ACCOUNT_NUMBER	RA_CUST_TRX_LINE_GL_DIST_ALL	CODE_COMBINATION_ID	The concatenated segments value for the Accounting flexfields, which maps to the CODE_COMBINATION_ID.
COMMENTS	RA_CUST_TRX_LINE_GL_DIST_ALL	COMMENTS	Comments about this accounting distribution.
INTERFACE_LINE_CONTEXT	RA_CUST_TRX_LINE_GL_DIST_ALL	CUSTOMER_TRX_LINE_ID	The line transaction flexfield for this transaction.
INTERFACE_LINE_ATTRIBUTE1			
INTERFACE_LINE_ATTRIBUTE2			
INTERFACE_LINE_ATTRIBUTE3			
INTERFACE_LINE_ATTRIBUTE4			
INTERFACE_LINE_ATTRIBUTE5			
INTERFACE_LINE_ATTRIBUTE6			

Document Field	Oracle Applications Table/View Name	Column Name	Description
INTERFACE_LINE_ATTRIBUTE7			
INTERFACE_LINE_ATTRIBUTE8			
INTERFACE_LINE_ATTRIBUTE9			
INTERFACE_LINE_ATTRIBUTE10			
INTERFACE_LINE_ATTRIBUTE11			
INTERFACE_LINE_ATTRIBUTE12			
INTERFACE_LINE_ATTRIBUTE13			
INTERFACE_LINE_ATTRIBUTE14			
INTERFACE_LINE_ATTRIBUTE15			
ACCT_DISTRIBUTION_PERCENT	RA_CUST_TRX_LINE_GL_DIST_ALL	PERCENT	The percentage for this accounting distribution.

1.2 CHARGES

Document Field	Oracle Applications Table/View Name	Column Name	Description
CUSTOMER_TRX_ID	RA_CUSTOMER_TRX_LINES_ALL	CUSTOMER_TRX_ID	Customer Transaction ID
CUSTOMER_TRX_LINE_ID	RA_CUSTOMER_TRX_LINES_ALL	CUSTOMER_TRX_LINE_ID	Customer Transaction Line ID
LINK_TO_CUST_TRX_LINE_ID	RA_CUSTOMER_TRX_LINES_ALL	LINK_TO_CUST_TRX_LINE_ID	
DESCRIPTION	RA_CUSTOMER_TRX_LINES_ALL	DESCRIPTION	Description
QUANTITY	RA_CUSTOMER_TRX_LINES_ALL	QUANTITY	Number of Units shipped.

Document Field	Oracle Applications Table/View Name	Column Name	Description
QUANTITY_ORDERED	RA_CUSTOMER_TRX_LINES_ALL	QUANTITY_ORDERED	The original number of units ordered for this transaction.
INVENTORY_ITEM	RA_CUSTOMER_TRX_LINES_ALL	INVENTRY_ID	The concatenated Inventory Flexfiled, which will be used to derive the INVENTORY_ID for this transaction.
UNIT_STANDARD_PRICE	RA_CUSTOMER_TRX_LINES_ALL	UNIT_STANDARD_PRICE	The standard price per unit for this transaction.
UNIT_SELLING_PRICE	RA_CUSTOMER_TRX_LINES_ALL	UNIT_SELLING_PRICE	The selling price per unit for this transaction.
MEMO_LINE_NAME	RA_CUSTOMER_TRX_LINES_ALL	MEMO_LINE_ID	The name of the standard memo line for this transaction.
SALES_ORDER_SOURCE	RA_CUSTOMER_TRX_LINES_ALL	SALES_ORDER_SOURCE	The source of the sales order for this transaction.
SALES_ORDER	RA_CUSTOMER_TRX_LINES_ALL	SALES_ORDER	The sales order number for this transaction.
SALES_ORDER_REVISION	RA_CUSTOMER_TRX_LINES_ALL	SALES_ORDER_REVISION	The sales order revision for this transaction.
SALES_ORDER_LINE	RA_CUSTOMER_TRX_LINES_ALL	SALES_ORDER_LINE	The sales order line number for this transaction.
SALES_ORDER_DATE	RA_CUSTOMER_TRX_LINES_ALL	SALES_ORDER_DATE	The date of the sales order for this transaction.
INTERFACE_LINE_CONTEXT	RA_CUSTOMER_TRX_LINES_ALL	INTERFACE_LINE_CONTEXT	The line transaction flexfield for this transaction.
INTERFACE_LINE_ATTRIBUTE1	RA_CUSTOMER_TRX_LINES_ALL	INTERFACE_LINE_ATTRIBUTE1	
INTERFACE_LINE_ATTRIBUTE2		INTERFACE_LINE_ATTRIBUTE2	
INTERFACE_LINE_ATTRIBUTE3		INTERFACE_LINE_ATTRIBUTE3	

Document Field	Oracle Applications Table/View Name	Column Name	Description
INTERFACE_LINE_ATTRIBUTE4		INTERFACE_LINE_ATTRIBUTE4	
INTERFACE_LINE_ATTRIBUTE5		INTERFACE_LINE_ATTRIBUTE5	
INTERFACE_LINE_ATTRIBUTE6		INTERFACE_LINE_ATTRIBUTE6	
INTERFACE_LINE_ATTRIBUTE7		INTERFACE_LINE_ATTRIBUTE7	
INTERFACE_LINE_ATTRIBUTE8		INTERFACE_LINE_ATTRIBUTE8	
INTERFACE_LINE_ATTRIBUTE9		INTERFACE_LINE_ATTRIBUTE9	
INTERFACE_LINE_ATTRIBUTE10		INTERFACE_LINE_ATTRIBUTE10	
INTERFACE_LINE_ATTRIBUTE11		INTERFACE_LINE_ATTRIBUTE11	
INTERFACE_LINE_ATTRIBUTE12		INTERFACE_LINE_ATTRIBUTE12	
INTERFACE_LINE_ATTRIBUTE13		INTERFACE_LINE_ATTRIBUTE13	
INTERFACE_LINE_ATTRIBUTE14		INTERFACE_LINE_ATTRIBUTE14	
INTERFACE_LINE_ATTRIBUTE15		INTERFACE_LINE_ATTRIBUTE15	
REFERENCE_LINE_CONTEXT	RA_CUSTOMER_TRX_LINES_ALL	PREVIOUS_CUSTOMER_TRX_LINE_ID/INITIAL_CUSTOMER_TRX_LINE_ID	If this transaction is a credit memo, the transaction flexfield of the transaction line you are crediting in these columns of the transaction you are crediting.
REFERENCE_LINE_ATTRIBUTE1			
REFERENCE_LINE_ATTRIBUTE2			

Document Field	Oracle Applications Table/View Name	Column Name	Description
REFERENCE_LINE_ ATTRIBUTE3			
REFERENCE_LINE_ ATTRIBUTE4			
REFERENCE_LINE_ ATTRIBUTE5			
REFERENCE_LINE_ ATTRIBUTE6			
REFERENCE_LINE_ ATTRIBUTE7			
REFERENCE_LINE_ ATTRIBUTE8			
REFERENCE_LINE_ ATTRIBUTE9			
REFERENCE_LINE_ ATTRIBUTE10			
REFERENCE_LINE_ ATTRIBUTE11			
REFERENCE_LINE_ ATTRIBUTE12			
REFERENCE_LINE_ ATTRIBUTE13			
REFERENCE_LINE_ ATTRIBUTE14			
REFERENCE_LINE_ ATTRIBUTE15			
ATTRIBUTE_ CATEGORY	RA_CUSTOMER_TRX _LINES_ALL	ATTRIBUTE_ CATEGORY	Descriptive flexfield attributes allow you to store additional columns, the contents of which you define.
ATTRIBUTE1		ATTRIBUTE1	
ATTRIBUTE2		ATTRIBUTE2	
ATTRIBUTE3		ATTRIBUTE3	
ATTRIBUTE4		ATTRIBUTE4	

Document Field	Oracle Applications Table/View Name	Column Name	Description
ATTRIBUTE5		ATTRIBUTE5	
ATTRIBUTE6		ATTRIBUTE6	
ATTRIBUTE7		ATTRIBUTE7	
ATTRIBUTE8		ATTRIBUTE8	
ATTRIBUTE9		ATTRIBUTE9	
ATTRIBUTE10		ATTRIBUTE10	
ATTRIBUTE11		ATTRIBUTE11	
ATTRIBUTE12		ATTRIBUTE12	
ATTRIBUTE13		ATTRIBUTE13	
ATTRIBUTE14		ATTRIBUTE14	
ATTRIBUTE15		ATTRIBUTE15	

1.3 FREIGHT_LINES

Document Field	Oracle Applications Table/View Name	Column Name	Description
CUSTOMER_TRX_ID	RA_CUSTOMER_TRX_LINES_ALL	CUSTOMER_TRX_ID	Customer Transaction ID.
CUSTOMER_TRX_LINE_ID	RA_CUSTOMER_TRX_LINES_ALL	CUSTOMER_TRX_LINE_ID	Customer Transaction Line ID.
LINK_TO_CUST_TRX_LINE_ID	RA_CUSTOMER_TRX_LINES_ALL	LINK_TO_CUST_TRX_LINE_ID	
DESCRIPTION	RA_CUSTOMER_TRX_LINES_ALL	DESCRIPTION	Description.
QUANTITY	RA_CUSTOMER_TRX_LINES_ALL	QUANTITY_INVOICED	Number of units shipped.
QUANTITY_ORDERED	RA_CUSTOMER_TRX_LINES_ALL	QUANTITY_ORDERED	The original number of units ordered for this transaction.
UNIT_STANDARD_PRICE	RA_CUSTOMER_TRX_LINES_ALL	UNIT_STANDARD_PRICE	The standard price per unit for this transaction.
UNIT_SELLING_PRICE	RA_CUSTOMER_TRX_LINES_ALL	UNIT_SELLING_PRICE	The selling price per unit for this transaction.

Document Field	Oracle Applications Table/View Name	Column Name	Description
AMOUNT	RA_CUSTOMER_TRX_LINES_ALL	REVENUE_AMOUNT	The revenue amount for this transaction.
ACCOUNTING_RULE_DURATION	RA_CUSTOMER_TRX_LINES_ALL	ACCOUNTING_RULE_DURATION	Duration of accounting rule.
ACCOUNTING_RULE_NAME	RA_CUSTOMER_TRX_LINES_ALL	ACCOUNTING_RULE_NAME	The accounting rule name for this transaction.
RULE_START_DATE	RA_CUSTOMER_TRX_LINES_ALL	RULE_START_DATE	The date that you want to start the accounting rule for this transaction.
LAST_PERIOD_TO_CREDIT	RA_CUSTOMER_TRX_LINES_ALL	LAST_PERIOD_TO_CREDIT	For unit credit memos, the last period number from which you want to start crediting.
MEMO_LINE_NAME	RA_CUSTOMER_TRX_LINES_ALL	MEMO_LINE_ID	The name of the standard memo line for this transaction.
SALES_ORDER_SOURCE	RA_CUSTOMER_TRX_LINES_ALL	SALES_ORDER_SOURCE	The source of the sales order for this transaction.
SALES_ORDER	RA_CUSTOMER_TRX_LINES_ALL	SALES_ORDER	The sales order number for this transaction.
SALES_ORDER_REVISION	RA_CUSTOMER_TRX_LINES_ALL	SALES_ORDER_REVISION	The sales order revision for this transaction.
SALES_ORDER_LINE	RA_CUSTOMER_TRX_LINES_ALL	SALES_ORDER_LINE	The sales order line number for this transaction.
SALES_ORDER_DATE	RA_CUSTOMER_TRX_LINES_ALL	SALES_ORDER_DATE	The date of the sales order for this transaction.
INTERFACE_LINE_CONTEXT	RA_CUSTOMER_TRX_LINES_ALL	INTERFACE_LINE_CONTEXT	The Line Transaction flexfield for this transaction.
INTERFACE_LINE_ATTRIBUTE1		INTERFACE_LINE_ATTRIBUTE1	
INTERFACE_LINE_ATTRIBUTE2		INTERFACE_LINE_ATTRIBUTE2	
INTERFACE_LINE_ATTRIBUTE3		INTERFACE_LINE_ATTRIBUTE3	

Document Field	Oracle Applications Table/View Name	Column Name	Description
INTERFACE_LINE_ATTRIBUTE4		INTERFACE_LINE_ATTRIBUTE4	
INTERFACE_LINE_ATTRIBUTE5		INTERFACE_LINE_ATTRIBUTE5	
INTERFACE_LINE_ATTRIBUTE6		INTERFACE_LINE_ATTRIBUTE6	
INTERFACE_LINE_ATTRIBUTE7		INTERFACE_LINE_ATTRIBUTE7	
INTERFACE_LINE_ATTRIBUTE8		INTERFACE_LINE_ATTRIBUTE8	
INTERFACE_LINE_ATTRIBUTE9		INTERFACE_LINE_ATTRIBUTE9	
INTERFACE_LINE_ATTRIBUTE10		INTERFACE_LINE_ATTRIBUTE10	
INTERFACE_LINE_ATTRIBUTE11		INTERFACE_LINE_ATTRIBUTE11	
INTERFACE_LINE_ATTRIBUTE12		INTERFACE_LINE_ATTRIBUTE12	
INTERFACE_LINE_ATTRIBUTE13		INTERFACE_LINE_ATTRIBUTE13	
INTERFACE_LINE_ATTRIBUTE14		INTERFACE_LINE_ATTRIBUTE14	
INTERFACE_LINE_ATTRIBUTE15		INTERFACE_LINE_ATTRIBUTE15	
REFERENCE_LINE_CONTEXT	RA_CUSTOMER_TRX_LINES_ALL	PREVIOUS_CUSTOMER_TRX_LINE_ID/INITIAL_CUSTOMER_TRX_LINE_ID	If this transaction is a credit memo, the transaction flexfield of the transaction line you are crediting in these columns, or the transaction you are crediting.
REFERENCE_LINE_ATTRIBUTE1			
REFERENCE_LINE_ATTRIBUTE2			

Document Field	Oracle Applications Table/View Name	Column Name	Description
REFERENCE_LINE_ATTRIBUTE3			
REFERENCE_LINE_ATTRIBUTE4			
REFERENCE_LINE_ATTRIBUTE5			
REFERENCE_LINE_ATTRIBUTE6			
REFERENCE_LINE_ATTRIBUTE7			
REFERENCE_LINE_ATTRIBUTE8			
REFERENCE_LINE_ATTRIBUTE9			
REFERENCE_LINE_ATTRIBUTE10			
REFERENCE_LINE_ATTRIBUTE11			
REFERENCE_LINE_ATTRIBUTE12			
REFERENCE_LINE_ATTRIBUTE13			
REFERENCE_LINE_ATTRIBUTE14			
REFERENCE_LINE_ATTRIBUTE15			
ATTRIBUTE_CATEGORY	RA_CUSTOMER_TRX_LINES_ALL	ATTRIBUTE_CATEGORY	Descriptive flexfield attributes allow you to store additional columns, the contents of which you define.
ATTRIBUTE1		ATTRIBUTE1	
ATTRIBUTE2		ATTRIBUTE2	
ATTRIBUTE3		ATTRIBUTE3	
ATTRIBUTE4		ATTRIBUTE4	

Document Field	Oracle Applications Table/View Name	Column Name	Description
ATTRIBUTE5		ATTRIBUTE5	
ATTRIBUTE6		ATTRIBUTE6	
ATTRIBUTE7		ATTRIBUTE7	
ATTRIBUTE8		ATTRIBUTE8	
ATTRIBUTE9		ATTRIBUTE9	
ATTRIBUTE10		ATTRIBUTE10	
ATTRIBUTE11		ATTRIBUTE11	
ATTRIBUTE12		ATTRIBUTE12	
ATTRIBUTE13		ATTRIBUTE13	
ATTRIBUTE14		ATTRIBUTE14	
ATTRIBUTE15		ATTRIBUTE15	

Send Customer Service

The name of this service is:

WmOAFIN107SC.receivables107SC.fromOA.customer:sendCustomer

This service provides the list of new or changed customers.

You set up Customers in Oracle Applications Receivables to record customer-related information, such as persons or organizations.

Database Scripts

This service uses the following database scripts:

Database Script	Description
wm_install_from_customer.sql	Runs all the scripts listed below, except the uninstall script.
wm_from_customer_vw.sql	<div>Creates all required views for customer outbound transactions:</div> <div><ul style="list-style-type: none">■ WM_AR_CUSTOMERS_VW■ WM_CUST_PHONES_VW■ WM_CONT_PHONES_VW■ WM_AR_SITE_USES_VW■ WM_AR_CUST_RELATIONSHIPS_VW■ WM_AR_CUST_RECEIPT_METHODS_VW■ WM_AR_CUST_BANK_ACCOUNTS_VW■ WM_AR_CUSTOMER_PROF_AMTS_VW■ WM_AR_CUSTOMER_PROFILES_VW■ WM_AR_CONTACT_ROLES_VW■ WM_AR_CONTACTS_VW■ WM_AR_ADDRESSES_VW■ WM_ADDR_PHONES_VW■ WM_AR_CUSTOMERS_QRY_VW</div>

Database Script	Description
wm_from_customer_trg.sql	<p>Creates the following trigger components to be used for writing to the WM_TRACKCHANGES table to indicate a new or updated document:</p> <ul style="list-style-type: none"> ■ WM_RA_CUSTOMERS_IU_TRG ■ WM_RA_ADDRESSES_IU_TRG ■ WM_AR_CUSTOMER_PROFILES_IU_TRG ■ WM_AR_CUST_PROF_AMTS_IUD_TRG ■ WM_AP_BANK_ACCOUNT_USES_IU_TRG ■ WM_RA_SITE_USES_IU_TRG ■ WM_RA_PHONES_IUD_TRG ■ WM_RA_CUST_RELATIONS_IU_TRG ■ WM_RA_CUST_RECIP_METHODS_IU_TRG ■ WM_RA_CONTACTSIU_TRG ■ WM_RA_CONTACT_ROLES_IUD_TRG
wm_disable_from_customer.sql	Disables the triggers installed by wm_from_customer_trg.sql.
wm_enable_from_customer.sql	Re-enables the triggers installed by wm_from_customer_trg.sql.
wm_drop_from_customer.sql	Uninstalls all components created by wm_install_from_customer.sql.

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definition:

- getCustomerTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow executes as follows:

- **specifyDefaultSettings** specifies the default parameter settings required for service execution. You should change these settings accordingly.
- **lockTxnCtrl** determines whether the **sendJournal** service is ready for execution. That is, it determines whether there is no other instance of this service already in process.
 - If **lockTxnCtrl** returns **False**, it means that another instance of this service is already in process. The service exits, and waits for next scheduled execution.
 - If **lockTxnCtrl** returns **True**, it means that the service is ready to execute, the Journal row in the control table is locked and updated, and status is changed to **INPROCESS**. This prevents any other Journal service from executing.
- **getJournalTxn** queries the Oracle Applications database for any Journal Transaction to be processed. The number of records returned depends on the parameter value specified in the above map.
- **processBizDoc** is the customizable step that sends the business document to the recipient such as a trading partner by looping against each document. You must customize this step to receive a **SUCCESS** or an **ERROR** status of the document transfer along with the Error information. Transfer Status and any Error Information is logged against each document.
- Loop against each document. Based on the Debug Mode specified during execution, it either purges or updates the records in the **WM_TRACKCHANGES** custom table.
 - If the Debug Mode is **TRUE**, the records in the **WM_TRACKCHANGES** table are updated and the **PROCESSED_FLAG** is set to **Y**. This ensures that same sets of records are not picked up during next polling interval. The **updateTrackChanges** service updates the **PROCESSED_FLAG** in the **WM_TRACKCHANGES** table to **Y** and updates **PROCESSED_DATE** to **sysdate** so that same information is not picked up again during next polling instance.
 - If the Debug Mode is **FALSE**, the records in the **WM_TRACKCHANGES** table are deleted. The **purgeTrackChanges** service purges the records from the **WM_TRACKCHANGES** table.
- Loop against each document. Based on the Transfer Status, **insertTransferERRInfo** service inserts a new record in the **WM_TRACKCHANGES** table so that same document can be picked up during the next polling interval.
- **unlockTxnCtrl** releases the lock on the Custom Control table so that next polling instance of **sendVendor** service can begin.

- `getLastError` service logs any errors.
- `unlockTxnCtrl` service releases the lock on the Custom Control table.

For more details on send transactions, see [“Overview of Send Service Transaction Processing” on page 38](#).

Business Document Structure

This service uses the following business document structure:

- 1.0 CUSTOMERS
 - 1.1 SITES
 - 1.1.1 SITE_CONTACTS
 - 1.1.1.1 SITE_CONTACT_ROLES
 - 1.1.1.2 SITE_CONTACT_PHONES
 - 1.1.2 SITE_USES
 - 1.1.2.1 SITE_PAYMENT_METHODS
 - 1.1.2.2 SITE_BANK_ACCOUNTS
 - 1.1.2.3 SITE_PROFILES
 - 1.1.2.3.1 SITE_PROFILE_AMOUNTS
 - 1.1.3 SITE_PHONES
 - 1.2 **PHONES
 - 1.3 **CONTACTS
 - 1.3.1 **CONTACT_ROLES
 - 1.3.2 **CONTACT_PHONES
 - 1.4 **BANK_ACCOUNTS
 - 1.5 **PAYMENT_METHODS
 - 1.6** CUSTOMER_PROFILES
 - 1.6.1 **PROFILE_AMOUNTS
 - 1.7 CUSTOMER_RELATIONSHIPS

**Some documents use the same document fields, tables, views, and columns as shown in the following table:

For These Documents...	Use the Following Tables
■ 1.3 CONTACTS	“1.1.1 SITE_CONTACTS” on page 189
■ 1.3.1 CONTACT_ROLES	“1.1.1.1 SITE_CONTACT_ROLES” on page 190
■ 1.1.3 SITE_PHONES ■ 1.3.2 CONTACT_PHONES	“1.1.1.2 SITE_CONTACT_PHONES” on page 190
■ 1.4 BANK_ACCOUNTS	“1.1.2.2 SITE_BANK_ACCOUNTS” on page 193
■ 1.5 PAYMENT_METHODS	“1.1.2.1 SITE_PAYMENT_METHODS” on page 192
■ 1.6 CUSTOMER_PROFILES	“1.1.2.3 SITE_PROFILES” on page 195
■ 1.6.1 PROFILE_AMOUNTS	“1.1.2.3.1 SITE_PROFILE_AMOUNTS” on page 197

1.0 CUSTOMERS

Document Field	Oracle Applications Table/View Name	Column Name	Description
WEB_TRANSACTION_ID			
DOCUMENT_TYPE			Value is CUSTOMER.
DOCUMENT_STATUS			Value is UPDATE or INSERT.
CUSTOMER_ID	HZ_CUST_ACCOUNTS	CUST_ACCOUNT_ID	
CUSTOMER_NAME	HZ_CUST_ACCOUNTS	CUSTOMER_NAME	Name of the customer.
CUSTOMER_NUMBER	HZ_CUST_ACCOUNTS	CUSTOMER_NUMER	Customer Number.
CUSTOMER_KEY	HZ_PARTIES	CUSTOMER_KEY	Derived key created by Oracle Sales and Marketing to facilitate querying.
CUSTOMER_STATUS	HZ_CUST_ACCOUNTS	STATUS	Customer Status flag.
ORIG_SYSTEM_REFERENCE	HZ_CUST_ACCOUNTS	ORIG_SYSTEM_REFERENCE	Unique customer identifier from foreign system.
CUSTOMER_PROSPECT_CODE	AR_CUSTOMERS_V	CUSTOMER_PROSPECT_CODE	

Document Field	Oracle Applications Table/View Name	Column Name	Description
CUSTOMER_CATEGORY_CODE	HZ_PARTIES	CATEGORY_CODE	User-definable category.
CUSTOMER_CLASS_CODE	HZ_CUST_ACCOUNTS	CUSTOMER_CLASS_CODE	Customer class identifier.
CUSTOMER_TYPE	HZ_CUST_ACCOUNTS	CUSTOMER_TYPE	Receivables lookup code for CUSTOMER_TYPE. Use: I for internal customers R for external customers
PRIMARY_SALESREP_NAME	JTF_RS_SALESREPS	NAME	Name of sales representative.
SIC_CODE	HZ_PARTIES	SIC_CODE	Standard Industry classification code.
TAX_REFERENCE	HZ_PARTIES	TAX_REFERENCE	Taxpayer identification number
TAX_CODE	HZ_CUST_ACCOUNTS	TAX_CODE	Tax code for this customer
FOB_POINT	HZ_CUST_ACCOUNTS	FOB_POINT	The point in a shipment at which title to the goods is transferred, for example FOB delivered, where the title changes hands at the point of delivery.
SHIP_VIA	HZ_CUST_ACCOUNTS	SHIP_VIA	Name of shipping firm.
GSA_INDICATOR	HZ_PARTIES	GSA_INDICATOR_FLAG	If the organization is a public sector service agency.
SHIP_PARTIAL	HZ_CUST_ACCOUNTS	SHIP_PARTIAL	Flag to indicate whether a partial shipments can be sent.
TAXPAYER_ID	HZ_PARTIES	JGZZ_FISCAL_CODE	Fiscal code for certain European countries.
PRICE_LIST_NAME	OE_PRICE_LISTS_VL	NAME	
FREIGHT_TERM	HZ_CUST_ACCOUNTS	FREIGHT_TERM	Order Management lookup code for FREIGHT_TERMS.

Document Field	Oracle Applications Table/View Name	Column Name	Description
ORDER_TYPE_NAME	OE_TRANSACTION_TYPES_VL	NAME	
SALES_CHANNEL_CODE	HZ_CUST_ACCOUNTS	SALES_CHANNEL_CODE	Order Management lookup code for SALES_CHANNEL.
WAREHOUSE_NAME	HR_ALL_ORGANIZATION_UNITS	NAME	Warehouse name.
MISSION_STATEMENT	HZ_PARTIES	MISSION_STATEMENT	Corporate charter of organization.
NUM_OF_EMPLOYEES	HZ_PARTIES	EMPLOYEES_TOTAL	Total number of employees.
POTENTIAL_REVENUE_CURR_FY	HZ_PARTIES	CURR_FY_POTENTIAL_REVENUE	Potential revenue this fiscal year for organization.
POTENTIAL_REVENUE_NEXT_FY	HZ_PARTIES	NEXT_FY_POTENTIAL_REVENUE	Potential revenue next fiscal year for organization.
FISCAL_YEAREND_MONTH	HZ_PARTIES	FISCAL_YEAREND_MONTH	Month in which fiscal year ends for organization.
YEAR_ESTABLISHED	HZ_PARTIES	YEAR_ESTABLISHED	Year in which organization began doing business.
ANALYSIS_FY	HZ_PARTIES	ANALYSIS_FY	Fiscal year for financial information such as net worth, number of employees, and potential revenue.
COMPETITOR_FLAG	HZ_PARTIES	COMPETITOR_FLAG	Indicates whether organization is a competitor.
REFERENCE_USE_FLAG	HZ_PARTIES	REFERENCE_USE_FLAG	Indicates whether customer has agreed to be a reference.

Document Field	Oracle Applications Table/View Name	Column Name	Description
THIRD_PARTY_FLAG	HZ_PARTIES	THIRD_PARTY_FLAG	Tracks whether a customer is a direct customer of the organization using receivables or a third party supplier.
PROFILE_CLASS_NAME	HZ_CUST_PROFILE_CLASSES	NAME	Name of customer profile class.
CUSTOMER_NAME_PHONETIC	HZ_PARTIES	ORGANIZATION_NAME_PHONETIC	Japanese Kana, or phonetic representation of organization name
TAX_HEADER_LEVEL_FLAG	HZ_CUST_ACCOUNTS	TAX_HEADER_LEVEL_FLAG	Tax calculation level. Use: Y for header N for line
TAX_ROUNDING_RULE	HZ_CUST_ACCOUNTS	TAX_ROUNDING_RULE	Tax amount rounding rule.

1.1 SITES

Document Field	Oracle Applications Table/View Name	Column Name	Description
ADDRESS_ID	HZ_CUST_ACCT_SITES_ALL	CUST_ACCT_SITE_ID	Customer Site identifier.
STATUS	HZ_CUST_ACCT_SITES_ALL	STATUS	Customer status flag.
ORIG_SYSTEM_REFERENCE	HZ_CUST_ACCT_SITES_ALL	ORIG_SYSTEM_REFERENCE	Address identifier from foreign system.
TERRITORY_SHORT_NAME	FND_TERRITORIES_TL	TERRITORY_SHORT_NAME	Territory short name.
ADDRESS_STYLE	HZ_LOCATIONS	ADDRESS_STYLE	Do not use this column. (Used as context value for flexible address Format descriptive flexfield.)
ADDRESS1	HZ_LOCATIONS	ADDRESS1	First line for address.
ADDRESS2	HZ_LOCATIONS	ADDRESS2	Second line for address.

Document Field	Oracle Applications Table/View Name	Column Name	Description
ADDRESS3	HZ_LOCATIONS	ADDRESS3	Third line for address.
ADDRESS4	HZ_LOCATIONS	ADDRESS4	Fourth line for address.
CITY	HZ_LOCATIONS	CITY	City.
COUNTY	HZ_LOCATIONS	COUNTY	County.
STATE	HZ_LOCATIONS	STATE	State.
PROVINCE	HZ_LOCATIONS	PROVINCE	Province.
COUNTRY	HZ_LOCATIONS	COUNTRY	Country.
POSTAL_CODE	HZ_LOCATIONS	POSTAL_CODE	Postal code.
SU_BILL_TO_FLAG	AR_ADDRESSES_V	SU_BILL_TO_FLAG	
SU_SHIP_TO_FLAG	AR_ADDRESSES_V	SU_SHIP_TO_FLAG	
SU_MARKET_FLAG	AR_ADDRESSES_V	SU_MARKET_FLAG	
SU_DUN_FLAG	AR_ADDRESSES_V	SU_DUN_FLAG	
SU_STMT_FLAG	AR_ADDRESSES_V	SU_STMT_FLAG	
SU_LEGAL_FLAG	AR_ADDRESSES_V	SU_LEGAL_FLAG	
KEY_ACCOUNT_FLAG	AR_ADDRESSES_V	KEY_ACCOUNT_FLAG	
LANGUAGE	HZ_LOCATIONS	LANGUAGE	Language.
LANGUAGE_DESCRIPTION	FND_LANGUAGES_TL	DESCRIPTION	Language description.
ADDRESS_LINES_PHONETIC	HZ_LOCATIONS	ADDRESS_LINES_PHONETIC	Phonetic or Kana representation of the Kanji address lines, used in Japan.
ORGANIZATION_NAME	ORG_ORGANIZATION_DEFINITIONS	ORGANIZATION_NAME	Organization name.

1.1.1 SITE_CONTACTS

Document Field	Oracle Applications Table/View Name	Column Name	Description
CONTACT_ID	HZ_CUST_ACCOUNT_ROLES	CUST_ACCOUNT_ROLE_ID	Unique identifier for the role played by the party in the customer account.
CONTACT_NUMBER	HZ_ORG_CONTACTS	CONTACT_NUMBER	Contact Number.
TITLE	HZ_ORG_CONTACTS	TITLE	Title.
TITLE_MEANING	AR_LOOKUPS	MEANING	Title Meaning.
FIRST_NAME	HZ_PARTIES	PERSON_FIRST_NAME	Person first name.
LAST_NAME	HZ_PARTIES	PERSON_LAST_NAME	Person last name.
STATUS	HZ_CUST_ACCOUNT_ROLES	CURRENT_ROLE_STATE	Status of a role that the customer or party has assumed.
JOB_TITLE	HZ_ORG_CONTACTS	JOB_TITLE	Job title.
MAIL_STOP	HZ_ORG_CONTACTS	MAIL_STOP	Mail stop.
ORIG_SYSTEM_REFERENCE	HZ_CUST_ACCOUNT_ROLES	ORIG_SYSTEM_REFERENCE	Reference to identify foreign system.
CONTACT_KEY	HZ_PARTIES	CUSTOMER_KEY	Derived key created by Oracle Sales and Marketing to facilitate querying.
EMAIL_ADDRESS	HZ_PARTIES	EMAIL_ADDRESS	Email address.

1.1.1.1 SITE_CONTACT_ROLES

Document Field	Oracle Applications Table/View Name	Column Name	Description
CONTACT_ROLE_ID	HZ_ROLE_RESPONSIBILITY	CONTACT_ROLE_ID	Contact Role ID
USAGE_CODE	HZ_ROLE_RESPONSIBILITY	RESPONSIBILITY_TYPE	Lookup for Responsibility Type.
ORIG_SYSTEM_REFERENCE	HZ_ROLE_RESPONSIBILITY	ORIG_SYSTEM_REFERENCE	Unique identifier from foreign system.
PRIMARY_FLAG	HZ_ROLE_RESPONSIBILITY	PRIMARY_FLAG	The primary role responsibility for the party account.

1.1.1.2 SITE_CONTACT_PHONES

Document Field	Oracle Applications Table/View Name	Column Name	Description
PHONE_NUMBER	HZ_CONTACT_POINTS	PHONE_NUMBER	Telephone number formatted in the local format.
STATUS	HZ_CONTACT_POINTS	STATUS	Active or inactive status
PHONE_TYPE	HZ_CONTACT_POINTS	PHONE_LINE_TYPE	Lookup code for the type of phone line, for example, general, fax, inbound, outbound.
AREA_CODE	HZ_CONTACT_POINTS	PHONE_AREA_CODE	Area code within a country code.
EXTENSION	HZ_CONTACT_POINTS	PHONE_EXTENSION	Additional number used by an internal telephone system after the internal telephone system is contacted.
PRIMARY_FLAG	HZ_CONTACT_POINTS	PRIMARY_FLAG	The primary contact point for referenced party, site, or location.
ORIG_SYSTEM_REFERENCE	HZ_CONTACT_POINTS	ORIG_SYSTEM_REFERENCE	Identifier for this record from foreign system.

1.1.2 SITE_USES

Document Field	Oracle Applications Table/View Name	Column Name	Description
SITE_USE_ID	HZ_CUST_SITE_USES_ALL	SITE_USE_ID	Site Use identifier.
SITE_USE_CODE	HZ_CUST_SITE_USES_ALL	SITE_USE_CODE	Business purpose.
PRIMARY_FLAG	HZ_CUST_SITE_USES_ALL	PRIMARY_FLAG	Indicates whether site is primary.
STATUS	HZ_CUST_SITE_USES_ALL	STATUS	Site Use Status flag.
CONTACT_NAME	AR_SITE_USES_V	CONTACT_NAME	Contact Name.
BILL_TO_LOCATION	HZ_CUST_SITE_USES_ALL	LOCATION	Site Use Identifier.
ORIG_SYSTEM_REFERENCE	HZ_CUST_SITE_USES_ALL	ORIG_SYSTEM_REFERENCE	Site Use Identifier from foreign system.
SIC_CODE	HZ_CUST_SITE_USES_ALL	SIC_CODE	Standard Industry Classification code.
PAYMENT_TERM_NAME	RA_TERMS_TL	NAME	Payment Terms name.
GSA_INDICATOR	HZ_CUST_SITE_USES_ALL	GSA_INDICATOR	Indicates whether this site is a public sector government service agency.
SHIP_PARTIAL	HZ_CUST_SITE_USES_ALL	SHIP_PARTIAL	Indicates that the customer will accept partial shipments. The default is Y (yes).
SHIP_VIA	HZ_CUST_SITE_USES_ALL	SHIP_VIA	Name of the preferred shipping company.
FOB_POINT	HZ_CUST_SITE_USES_ALL	FOB_POINT	Free On Board (FOB). The point at which title of goods is transferred to the buyer.
ORDER_TYPE_NAME	OE_TRANSACTION_TYPES_TL	NAME	Order Type Name,
PRICE_LIST_NAME	OE_PRICE_LISTS_VL	NAME	Price List Name,

Document Field	Oracle Applications Table/View Name	Column Name	Description
FREIGHT_TERM	HZ_CUST_SITE_USES_ALL	FREIGHT_TERM	Order Management lookup code for FREIGHT_TERMS,
WAREHOUSE_NAME	HR_ALL_ORGANIZATION_UNITS	NAME	Warehouse Name,
TAX_REFERENCE	HZ_CUST_SITE_USES_ALL	TAX_REFERENCE	Taxpayer iDentification Number,
TAX_CODE	HZ_CUST_SITE_USES_ALL	TAX_CODE	Tax Code associated with this site,
DEMAND_CLASS_CODE	HZ_CUST_SITE_USES_ALL	DEMAND_CLASS_CODE	Demand Class.
INVENTORY_ORGANIZATION_NAME	HR_ALL_ORGANIZATION_UNITS	NAME	Inventory Organization Name.
TAX_CLASSIFICATION	HZ_CUST_SITE_USES_ALL	TAX_CLASSIFICATION	Classification Code for tax.
TAX_HEADER_LEVEL_FLAG	HZ_CUST_SITE_USES_ALL	TAX_HEADER_LEVEL_FLAG	Used by Oracle Sales Compensation.
TAX_ROUNDING_RULE	HZ_CUST_SITE_USES_ALL	TAX_ROUNDING_RULE	Tax rounding rule, overrides system and customer tax rounding rule.

1.1.2.1 SITE_PAYMENT_METHODS

Document Field	Oracle Applications Table/View Name	Column Name	Description
PRIMARY_FLAG	RA_CUST_RECEIPT_METHODS	PRIMARY_FLAG	Indicates whether customer receipt method is primary.
RECEIPT_METHOD_NAME	AR_RECEIPT_METHODS	NAME	Receipt Method Name.
START_DATE	RA_CUST_RECEIPT_METHODS	START_DATE	Start date of the Customer Receipt Methods.
END_DATE	RA_CUST_RECEIPT_METHODS	END_DATE	End date of the Customer Receipt Methods.

1.1.2.2 SITE_BANK_ACCOUNTS

Document Field	Oracle Applications Table/View Name	Column Name	Description
BANK_ACCOUNT_ID	AP_BANK_ACCOUNT_USES_ALL	BANK_ACCOUNT_ID	Bank Account ID.
EXTERNAL_BANK_ACCOUNT_ID	AP_BANK_ACCOUNT_USES_ALL	EXTERNAL_BANK_ACCOUNT_ID	External Bank Account ID.
START_DATE	AP_BANK_ACCOUNT_USES_ALL	START_DATE	Start Date used by the e bank account.
END_DATE	AP_BANK_ACCOUNT_USES_ALL	END_DATE	End date used by the bank account.
PRIMARY_FLAG	AP_BANK_ACCOUNT_USES_ALL	PRIMARY_FLAG	Indicates whether the bank account use is primary (Y) or not (N).
BANK_NUMBER	AP_BANK_BRANCHES	BANK_NUMBER	Bank Number.
BANK_NAME	AP_BANK_BRANCHES	BANK_NAME	Bank Name.
BANK_BRANCH_NAME	AP_BANK_BRANCHES	BANK_BRANCH_NAME	Bank Branch name.
BANK_BRANCH_NUM	AP_BANK_BRANCHES	BANK_NUM	Branch Number.
BANK_BRANCH_DESCRIPTION	AP_BANK_BRANCHES	DESCRIPTION	Description.
BANK_ACCOUNT_NAME	AP_BANK_ACCOUNTS_ALL	BANK_ACCOUNT_NAME	Bank account name.
BANK_ACCOUNT_NUM	AP_BANK_ACCOUNTS_ALL	BANK_ACCOUNT_NUM	Bank account number.
BANK_ACCOUNT_DESCRIPTION	AP_BANK_ACCOUNTS_ALL	DESCRIPTION	Description.
BANK_CURRENCY_CODE	AP_BANK_ACCOUNTS_ALL	CURRENCY_CODE	Currency Code.
BANK_ACCOUNT_INACTIVE_DATE	AP_BANK_ACCOUNTS_ALL	INACTIVE_DATE	Inactive date.
BANK_BRANCH_ADDRESS1	AP_BANK_BRANCHES	ADDRESS_LINE1	First address line.
BANK_BRANCH_ADDRESS2	AP_BANK_BRANCHES	ADDRESS_LINE2	Second address line.

Document Field	Oracle Applications Table/View Name	Column Name	Description
BANK_BRANCH_.ADDRESS3	AP_BANK_BRANCHES	ADDRESS_LINE3	Third address line.
BANK_BRANCH_.ADDRESS4	AP_BANK_BRANCHES	ADDRESS_LINE4	Fourth address line.
BANK_BRANCH_.CITY	AP_BANK_BRANCHES	CITY	CITY
BANK_BRANCH_.STATE	AP_BANK_BRANCHES	STATE	STATE
BANK_BRANCH_.ZIP	AP_BANK_BRANCHES	ZIP	ZIP
BANK_BRANCH_.PROVINCE	AP_BANK_BRANCHES	PROVINCE	PROVINCE
BANK_BRANCH_.COUNTRY	AP_BANK_BRANCHES	COUNTRY	COUNTRY
BANK_BRANCH_.AREA_CODE	AP_BANK_BRANCHES	AREA_CODE	AREA_CODE
BANK_BRANCH_.PHONE	AP_BANK_BRANCHES	PHONE	PHONE
BANK_BRANCH_.COUNTY	AP_BANK_BRANCHES	COUNTY	COUNTY
BANK_BRANCH_.EFT_USER_NUMBER	AP_BANK_BRANCHES	EFT_USER_NUMBER	The number that identifies you as a user of electronic funds transfer services to your bank or clearing organization.
BANK_ACCOUNT_.CHECK_DIGITS	AP_BANK_ACCOUNTS_ALL	CHECK_DIGITS	Holds any check digits that result from bank account number validation in FBS.

1.1.2.3 SITE_PROFILES

Document Field	Oracle Applications Table/View Name	Column Name	Description
CUSTOMER_PROFILE_ID	HZ_CUSTOMER_PROFILES	CUST_ACCOUNT_PROFILE_ID	Unique identifier of this customer profile.
STATUS	HZ_CUSTOMER_PROFILES	STATUS	Indicates whether the profile is active or inactive.
PROFILE_CLASS_NAME	HZ_CUST_PROFILE_CLASSES	NAME	Profile Class Name.
PROFILE_CLASS_DESCRIPTION	HZ_CUST_PROFILE_CLASSES	DESCRIPTION	Profile Class Description.
COLLECTOR_ID	AR_CUSTOMER_PROFILES	COLLECTOR_ID	Collector ID.
COLLECTOR_NAME	AR_COLLECTORS	NAME	Name of collector.
CREDIT_CHECKING	HZ_CUSTOMER_PROFILES	CREDIT_CHECKING	Indicates whether a credit check is to be carried out.
TOLERANCE	HZ_CUSTOMER_PROFILES	TOLERANCE	Percentage over credit limit that this customer can exceed before action is taken.
DISCOUNT_TERMS	HZ_CUSTOMER_PROFILES	DISCOUNT_TERMS	Indicates whether to allow discount terms. Default value is Y (Yes).
DUNNING_LETTERS	HZ_CUSTOMER_PROFILES	DUNNING_LETTERS	Indicates whether to send dunning letters to this customer when invoices, debit memos or chargebacks become past due.
INTEREST_CHARGES	HZ_CUSTOMER_PROFILES	INTEREST_CHARGES	Indicates whether to charge this customer interest.
STATEMENTS	HZ_CUSTOMER_PROFILES	SEND_STATEMENTS	Indicates whether to send this customer statements.
CREDIT_BALANCE_STATEMENTS	HZ_CUSTOMER_PROFILES	CREDIT_BALANCE_STATEMENTS	Indicates whether to send statements that have a credit balance.
CREDIT_HOLD	HZ_CUSTOMER_PROFILES	CREDIT_HOLD	Indicates whether to put a hold on this customer's credit.
CREDIT_RATING	HZ_CUSTOMER_PROFILES	CREDIT_RATING	List of values choice for credit rating.

Document Field	Oracle Applications Table/View Name	Column Name	Description
CREDIT_RATING_MEANING	AR_LOOKUPS	MEANING	Credit rating meaning.
RISK_CODE	HZ_CUSTOMER_PROFILES	RISK_CODE	
RISK_MEANING	HZ_CUSTOMER_PROFILES	RISK_MEANING	Risk meaning.
STANDARD_TERMS	HZ_CUSTOMER_PROFILES	STANDARD_TERMS	User-defined payment.
OVERRIDE_TERMS	HZ_CUSTOMER_PROFILES	OVERRIDE_TERMS	Indicates whether to allow override or use standard terms.
DUNNING_LETTER_SET_NAME	AR_DUNNING_LETTER_SETS	NAME	Dunning letter set name.
INTEREST_PERIOD_DAYS	HZ_CUSTOMER_PROFILES	INTEREST_PERIOD_DAYS	Number of days for the interest rate.
PAYMENT_GRACE_DAYS	HZ_CUSTOMER_PROFILES	PAYMENT_GRACE_DAYS	Maximum number of overdue days allowed before action.
DISCOUNT_GRACE_DAYS	HZ_CUSTOMER_PROFILES	DISCOUNT_GRACE_DAYS	Number of days for the discount term.
STATEMENT_CYCLE_NAME	AR_STATEMENT_CYCLES	NAME	Statement cycle name.
ACCOUNT_STATUS	HZ_CUSTOMER_PROFILES	ACCOUNT_STATUS	User defined account status.
ACCOUNT_STATUS_MEANING	HZ_CUSTOMER_PROFILES	ACCOUNT_STATUS_MEANING	User-defined account status.
AUTOCASH_HIERARCHY_NAME	AR_AUTOCASH_HIERARCHIES	HIERARCHY_NAME	Autocash hierarchy name.
AUTO_REC_INCL_DISPUTED_FLAG	HZ_CUSTOMER_PROFILES	AUTO_REC_INCL_DISPUTED_FLAG	Indicates whether to include disputed transactions.
TAX_PRINTING_OPTION	HZ_CUSTOMER_PROFILES	TAX_PRINTING_OPTION	Tax printing option.
TAX_PRINTING_OPTION_MEANING	HZ_CUSTOMER_PROFILES	TAX_PRINTING_OPTION_MEANING	Tax Printing Option.

Document Field	Oracle Applications Table/View Name	Column Name	Description
CHARGE_ON_FINANCE_CHARGE_FLAG	HZ_CUSTOMER_PROFILES	CHARGE_ON_FINANCE_CHARGE_FLAG	Indicates whether to compound finance charges for this customer or site.
GROUPING_RULE_NAME	RA_GROUPING_RULES	NAME	Grouping Rule Name.
CONS_INV_FLAG	HZ_CUSTOMER_PROFILES	CONS_INV_FLAG	Indicates whether customer will be sent a consolidated billing invoice.
CONS_INV_TYPE	HZ_CUSTOMER_PROFILES	CONS_INV_TYPE	Type of consolidated billing invoice, summary, or detail, sent to customer.
PERCENT_COLLECTABLE	HZ_CUSTOMER_PROFILES	PERCENT_COLLECTABLE	Percentage of customer's account balance that you expect to collect on a regular basis.
CLEARING_DAYS	HZ_CUSTOMER_PROFILES	CLEARING_DAYS	Number of clearing days before a customer's or site's receipts can be cleared by the automatic clearing program. This field overrides the value of the payment method and bank account.

1.1.2.3.1 SITE_PROFILE_AMOUNTS

Document Field	Oracle Applications Table/View Name	Column Name	Description
CURRENCY_CODE	HZ_CUST_PROFILE_AMTS	CURRENCY_CODE	Code defined for a currency.
TRX_CREDIT_LIMIT	HZ_CUST_PROFILE_AMTS	TRX_CREDIT_LIMIT	Credit limit for an order.
OVERALL_CREDIT_LIMIT	HZ_CUST_PROFILE_AMTS	OVERALL_CREDIT_LIMIT	Overall credit limit.
MIN_DUNNING_AMOUNT	HZ_CUST_PROFILE_AMTS	MIN_DUNNING_AMOUNT	Minimum total a dunning letter should have before the letter is printed.

Document Field	Oracle Applications Table/View Name	Column Name	Description
MIN_DUNNING_INVOICE_AMOUNT	HZ_CUST_PROFILE_AMTS	MIN_DUNNING_INVOICE_AMOUNT	The balance due on a payment schedule must be at least this value before it can be dunned.
MAX_INTEREST_CHARGE	HZ_CUST_PROFILE_AMTS	MAX_INTEREST_CHARGE	Maximum interest to be charged per invoice for a currency.
MIN_STATEMENT_AMOUNT	HZ_CUST_PROFILE_AMTS	MIN_STATEMENT_AMOUNT	Minimum total a statement should have before the statement is printed.
AUTO_REC_MIN_RECEIPT_AMOUNT	HZ_CUST_PROFILE_AMTS	AUTO_REC_MIN_RECEIPT_AMOUNT	Limits the minimum receipt amount for a currency.
INTEREST_RATE	HZ_CUST_PROFILE_AMTS	INTEREST_RATE	The interest rate to be charged to this customer account or site for invoices in this currency.
MIN_FC_BALANCE_AMOUNT	HZ_CUST_PROFILE_AMTS	MIN_FC_BALANCE_AMOUNT	Minimum balance that a customer account or customer account site should have before any finance charges can be charged to invoices.
MIN_FC_INVOICE_AMOUNT	HZ_CUST_PROFILE_AMTS	MIN_FC_INVOICE_AMOUNT	Minimum balance on an invoice before any finance charges can be computed for it.

1.7 CUSTOMER_RELATIONSHIPS

Document Field	Oracle Applications Table/View Name	Column Name	Description
RELATIONSHIP_TYPE_CODE	HZ_CUST_ACCT_RELATE_ALL	RELATIONSHIP_TYPE	Revenue Accounting code for RELATIONSHIP_TYPE.
RELATED_CUSTOMER_NAME	HZ_PARTIES	PARTY_NAME	Name of party.
RELATED_CUSTOMER_NUMBER	HZ_CUST_ACCOUNTS	ACCOUNT_NUMBER	Customer Account Number.

Document Field	Oracle Applications Table/View Name	Column Name	Description
STATUS	HZ_CUST_ACCT_RELATE_ALL	STATUS	Active or inactive status.
COMMENTS	HZ_CUST_ACCT_RELATE_ALL	COMMENTS	Additional comments
CUSTOMER_RECIPROCAL_FLAG	HZ_CUST_ACCT_RELATE_ALL	CUSTOMER_RECIPROCAL_FLAG	Indicates whether this relationship is reciprocal.
RELATIONSHIP_TYPE	HZ_CUST_ACCT_RELATE_ALL	RELATIONSHIP_TYPE	Revenue Accounting code for RELATIONSHIP_TYPE.

Send Journal Service

The name of this service is:

WmOAFIN107SC.generalLedger107SC.fromOA:journal:sendJournal

This service retrieves Journal information, such as general, recurring, mass allocation or reversing journal entries.

You must post the Journal entries to qualify them for sending to the trading partner.

While configuring the queryJournalTxn service, a java.outOfMemory error can occur. Because the database can have so many records, it would be unable to configure the service. One workaround is to use a restricting condition in the query, such as 1 = 2, so that the service configures successfully. After you have configured this service, remove the restricting condition from your transaction definitions. In addition, update the SQLOut parameters of the service's transactionRecord in the webMethods Developer and remove the restricting condition. (In this example, remove the 1 = 2 condition.)

Database Scripts

This service uses the following database scripts:

Database Script	Description
wm_install_from_journal.sql	Runs all the scripts listed below, except the uninstall script: <ul style="list-style-type: none"> ■ WM_GL_JOURNALS_QRY_VW ■ WM_GL_JOURNALS_VW
wm_from_journal_vw.sql	Creates all required views for Journal Outbound.

Database Script	Description
wm_from_journal_trg.sql	Creates the following trigger component to be used for writing to the WM_TRACKCHANGES table to indicate a new or updated document: ■ WM_GL_JE_LINES_IU_TRG
wm_disable_from_journal.sql	Disables the triggers installed by wm_from_journal_trg.sql.
wm_enable_from_journal.sql	Re-enables the triggers installed by wm_from_journal_trg.sql.
wm_drop_from_journal.sql	Uninstalls all components created by wm_install_from_journal.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definition:

- getJournalTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow executes as follows:

- specifyDefaultSettings specifies the default parameter settings required for service execution. You should change these settings accordingly.
- lockTxnCtrl determines whether the sendJournal service is ready for execution. That is, it determines whether there is no other instance of this service already in process.
 - If lockTxnCtrl returns False, it means that another instance of this service is already in process. The service exits, and waits for next scheduled execution.
 - If lockTxnCtrl returns True, it means that the service is ready to execute, the Journal row in the control table is locked and updated, and status is changed to INPROCESS. This prevents any other Journal service from executing.
- getJournalTxn queries the Oracle Applications database for any Journal Transaction to be processed. The number of records returned depends on the parameter value specified in the above map.

- **processBizDoc** is the customizable step that sends the business document to the recipient such as a trading partner by looping against each document. You must customize this step to receive a SUCCESS or an ERROR status of the document transfer along with the Error information. Transfer Status and any Error Information is logged against each document.
- Loop against each document. Based on the Debug Mode specified during execution, it either purges or updates the records in the WM_TRACKCHANGES custom table.
 - If the Debug Mode is TRUE, the records in the WM_TRACKCHANGES table are updated and the PROCESSED_FLAG is set to Y. This ensures that same sets of records are not picked up during next polling interval. **updateTrackChanges** service updates the PROCESSED_FLAG in the WM_TRACKCHANGES table to Y and updates PROCESSED_DATE to sysdate so that same information is not picked up again during next polling instance.
 - If the Debug Mode is FALSE, the records in the WM_TRACKCHANGES table are deleted. The **purgeTrackChanges** service purges the records from the WM_TRACKCHANGES table.
- Loop against each document. Based on the Transfer Status, **insertTransferERRInfo** service inserts a new record in the WM_TRACKCHANGES table so that same document can be picked up during the next polling interval.
- **unlockTxnCtrl** releases the lock on the Custom Control table so that next polling instance of sendVendor service can begin.
- Using **getLastError** service, any errors are logged. The **unlockTxnCtrl** service executes to release the lock on the Custom Control table.

For more details on send transactions, see [“Overview of Send Service Transaction Processing” on page 38](#).

Business Document Structure

This service uses the following business document structure:

- JOURNAL

JOURNAL

Document Field	Oracle Applications Table/View Name	Column Name	Description
WEB_TRANSACTION_ID			
DOCUMENT_TYPE			
DOCUMENT_STATUS			

Document Field	Oracle Applications Table/View Name	Column Name	Description
JE_LINE_ID	GL_JE_LINES	JE_LINE_ID	Document identifier for outbound transactions only.
SET_OF_BOOKS_SHORT_NAME	GL_SETS_OF_BOOKS	SHORT_NAME	Short name for set of books.
ACCOUNTING_DATE	GL_JE_HEADERS	DATE_CREATED	Accounting Date.
CURRENCY_CODE	GL_JE_HEADERS	CURRENCY_CODE	Currency Code.
DATE_CREATED	GL_JE_HEADERS	DATE_CREATED	Creation date.
CREATED_BY			
CREATED_BY_EMPLOYEE_NUMBER	PER_ALL_PEOPLE_F	EMPLOYEE_NUMBER	Employee Number.
CREATED_BY_NAME	PER_ALL_PEOPLE_F	FULL_NAME	Employee Name.
ACTUAL_FLAG	GL_JE_HEADERS	ACTUAL_FLAG	Actual flag.
CATEGORY_NAME	GL_JE_HEADERS	JE_CATEGORY	Category.
SOURCE_NAME	GL_JE_HEADERS	JE_SOURCE	Source.
CURRENCY_CONVERSION_DATE	GL_JE_HEADERS	CURRENCY_CONVERSION_DATE	Currency Conversion Date.
ENCUMBRANCE_TYPE	GL_ENCUMBRANCE_TYPES	ENCUMBRANCE_TYPE	Encumbrance type.
BUDGET_NAME	GL_BUDGET_VERSIONS	BUDGET_NAME	Budget Name.
USER_CURRENCY_CONVERSION_TYPE	GL_JE_HEADERS	CURRENCY_CONVERSION_TYPE	Currency cOnversion Type.
CURRENCY_CONVERSION_RATE	GL_JE_HEADERS	CURRENCY_CONVERSION_RATE	Currency Conversion Rate.
ENTERED_DR	GL_JE_LINES	ENTERED_DR	Entered Debit.

Document Field	Oracle Applications Table/View Name	Column Name	Description
ENTERED_CR	GL_JE_LINES	ENTERED_CR	Entered Credit.
ACCOUNTED_DR	GL_JE_LINES	ACCOUNTED_DR	Accounted Debit.
ACCOUNTED_CR	GL_JE_LINES	ACCOUNTED_CR	Accounted Credit.
REFERENCE	GL_JE_LINES	REFERENCE_1	Reference.
BATCH_NAME	GL_JE_BATCHES	NAME	Batch Name.
PERIOD_NAME	GL_JE_HEADERS	PERIOD_NAME	Period Name.
ACCOUNT_CODE	GL_CODE_COMBINATIONS_KFV	CONCATENATED_SEGMENTS	Account Code.
STAT_AMOUNT	GL_JE_LINES	STAT_AMOUNT	Statistical Amount.
INVOICE_DATE	GL_JE_LINES	INVOICE_DATE	DATE Value added tax descriptive flexfield column.
TAX_CODE	GL_JE_LINES	TAX_CODE	Value-added tax descriptive flexfield column.
INVOICE_IDENTIFIER	GL_JE_LINES	INVOICE_IDENTIFIER	Value-added tax descriptive flexfield column.
INVOICE_AMOUNT	GL_JE_LINES	INVOICE_AMOUNT	Value-added tax descriptive flexfield column.
USSGL_TRANSACTION_CODE	GL_JE_HEADERS	USSGL_TRANSACTION_CODE	Government transaction Code.
JGZZ_RECON_REF	GL_JE_HEADERS	JGZZ_RECON_REF	Global reconciliation reference.
AVERAGE_JOURNAL	GL_JE_BATCHES	AVERAGE_JOURNAL_FLAG	Average Journal flag.
POSTED_DATE	GL_JE_HEADERS	POSTED_DATE	Date Journal was posted.

Send Vendor Service

The name of this service is:
WmOAFIN107SC.payables107SC.fromOA.vendor:sendVendor

This service provides a list of new or changed vendors.

You set up Vendors (also known as Suppliers) in Oracle Applications Payables and Oracle Applications Purchasing to record information about individuals and companies from whom you purchase goods and services.

The following business rules will apply for polling Vendor data from Oracle Applications:

- Picks up new and changed Vendor records only.
- Picks up active Vendors only.
- No deletion of Vendors, once you define them in Oracle Applications. Therefore, Business Document with document status of INSERT or UPDATE will be created. The document status of DELETE does not hold in this case.

Database Scripts

This service uses the following database scripts:

Database Script	Description
wm_install_from_vendor.sql	Runs all the scripts listed below, except the uninstall script.
wm_out_vendor_vw.sql	Creates the following required view components: <ul style="list-style-type: none">■ WM_PO_VENDORS_VW■ WM_PO_VENDOR_SITES_ALL_VW■ WM_PO_VENDOR_CONTACTS_VW■ WM_BANK_ACCOUNT_USES_VENDOR_VW■ WM_BANK_ACCUSE_VEND_SITES_VW■ WM_PO_VENDORS_QRY_VW■ WM_BANK_ACC_USE_VEND_SITES_VW■ WM_BANK_ACCOUNT_USES_VENDOR_VW

Database Script	Description
wm_out_vendor_trg.sql	Creates the following trigger component to be used for writing to the WM_TRACKCHANGES table to indicate a new or updated document: ■ WM_PO_VENDORS_IU_TRG
wm_disable_from_vendor.sql	Disables the triggers installed by wm_from_vendor_trg.sql.
wm_enable_from_vendor.sql	Re-enables the triggers installed by wm_from_vendor_trg.sql.
wm_drop_from_vendor.sql	Uninstalls all components created by wm_install_from_vendor.sql.

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definition:

- sendVendorTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow executes as follows:

- **specifyDefaultSettings** specifies the default parameter settings required for service execution. You should change these settings accordingly.
 - **noOfRowToFetch** specifies the number of rows to fetch from the Oracle Applications database. The default value is 100000 to fetch all the rows. Change this value to fetch less rows.
 - **DebugMode** default value is FALSE. To turn on the debug mode, set this field to TRUE.
- **lockTxnCtrl** checks if no other instance of this service is already in process. If not, it takes control of processing in a single transaction. Specify VENDOR as value to the parameter P_TRANSACTION_TYPE.
 - If lockTxnCtrl returns FALSE, it means that another instance of this service is already in process. The service exits, and waits for next scheduled execution.
 - If lockTxnCtrl returns TRUE, it means that the service is ready to execute, the Vendor row in the control table is locked and updated to INPROCESS. This

prevents any other Vendor service from executing. The \$dbAlias, status (TRUE or FALSE) and message are retained at this step.

- **getVendorTxn** queries the Oracle Applications database for any Vendor Transaction to be processed. The number of records returned depends on the parameter value specified in the above map.
- **processBizDoc** is the customizable step that sends the business document to the recipient (such as Trading Partner) by looping against each document. You must customize this step to receive a SUCCESS or an ERROR status of the document transfer along with the Error information. The Transfer Status and the Error Information, if any, are logged against each document.
- Loop against each document. Records will be purged or updated in the WM_TRACKCHANGES custom table based on the Debug Mode specified during execution.
 - If the Debug Mode is TRUE, the records in the WM_TRACKCHANGES table are updated and the PROCESSED_FLAG is set to Y to ensure that the same sets of records are not picked up during the next polling interval. The **updateTrackChanges** service changes the PROCESSED_FLAG in the WM_TRACKCHANGES table to Y and updates PROCESSED_DATE to sysdate so that same information is not picked up again during the next polling instance.
 - If the Debug Mode is FALSE, the records in the WM_TRACKCHANGES table are deleted. The **purgeTrackChanges** service purges the records from the WM_TRACKCHANGES table.
- Loop against each document. and based on the Transfer Status, **insertTransferERRInfo** service is used to insert a new record in the WM_TRACKCHANGES table so that same document can be picked up during the next polling interval.
- Loop against each document. The **insertTransferERRInfo** service will insert a new record in the WM_TRACKCHANGES table based on the Transfer Status specified during execution.
- **unlockTxnCtrl** releases the lock on the Custom Control table so that next polling instance of **sendVendor** service can begin.
- **getLastError** service logs any errors.

For more details on send transactions, see [“Overview of Send Service Transaction Processing” on page 38](#).

Business Document Structure

This service uses the following business document structure:

- 1.0 VENDORS
 - 1.1. BANK_ACCOUNTS
 - 1.2 VENDOR_SITES
 - 1.2.1 SITE_BANK ACCOUNTS
 - 1.2.2 VENDOR_CONTACTS

1.0 VENDORS

Document Field	Oracle Applications Table/View Name	Column Name	Description
WEB_TRANSACTION_ID			Populated from a sequence and used internally in the Flow. Will contain NULL value for Queried Vendor data.
DOCUMENT_TYPE			Valid value is VENDOR.
DOCUMENT_STATUS			Valid values are INSERT and UPDATE and QUERY.
VENDOR_ID	PO_VENDORS	VENDOR_ID	Unique vendor identifier in Oracle Applications.
VENDOR_NUMBER	PO_VENDORS	SEGMENT1	Vendor number assigned to the Vendor.
VENDOR_NAME	PO_VENDORS	VENDOR_NAME	Name of the Vendor.
ALTERNATE_VENDOR_NAME	PO_VENDORS	VENDOR_NAME_ALT	Alternate name for the Vendor.
VENDOR_TYPE	PO_VENDORS	VENDOR_TYPE_LOOKUP_CODE	Type of Vendor.
EMPLOYEE_NUMBER	PER_ALL_PEOPLE_F	EMPLOYEE_NUMBER	Employee Number, if the same employee is set up as a Supplier.

Document Field	Oracle Applications Table/View Name	Column Name	Description
EMPLOYEE_NAME	PER_ALL_PEOPLE_F	FULL_NAME	Full name of the employee.
PARENT_VENDOR_NUMBER	PO_VENDORS	SEGMENT1	Parent Vendor Number.
PARENT_VENDOR_NAME	PO_VENDORS	VENDOR_NAME	Parent Vendor Name.
CUSTOMER_NUMBER	PO_VENDORS	CUSTOMER_NUM	Customer Number as setup in vendor's side.
ONE_TIME_FLAG	PO_VENDORS	ONE_TIME_FLAG	Indicates whether vendor is a one-time vendor.
MINIMUM_ORDER_AMOUNT	PO_VENDORS	MIN_ORDER_AMOUNT	Minimum order amount to order goods from the supplier.
BILL_LOC_ADDRESS_LINE1	HR_LOCATIONS	ADDRESS_LINE_1	<ul style="list-style-type: none"> ■ REGION1 stores the County ■ Region 2 stores the State Code. ■ BILL_LOC indicates Billing Location while ■ SHIP_LOC indicates Shipping Location
BILL_LOC_ADDRESS_LINE2	HR_LOCATIONS	ADDRESS_LINE_2	
BILL_LOC_ADDRESS_LINE3	HR_LOCATIONS	ADDRESS_LINE_3	
BILL_LOC_TOWN_OR_CITY	HR_LOCATIONS	TOWN_OR_CITY	
BILL_LOC_COUNTRY	HR_LOCATIONS	COUNTRY	
BILL_LOC_POSTAL_CODE	HR_LOCATIONS	POSTAL_CODE	
BILL_LOC_REGION1	HR_LOCATIONS	REGION_1	
BILL_LOC_REGION2	HR_LOCATIONS	REGION_2	

Document Field	Oracle Applications Table/View Name	Column Name	Description
SHIP_LOC_ADDRESS_LINE1	HR_LOCATIONS	ADDRESS_LINE_1	
SHIP_LOC_ADDRESS_LINE2	HR_LOCATIONS	ADDRESS_LINE_2	
SHIP_LOC_ADDRESS_LINE3	HR_LOCATIONS	ADDRESS_LINE_3	
SHIP_LOC_TOWN_OR_CITY	HR_LOCATIONS	TOWN_OR_CITY	
SHIP_LOC_COUNTRY	HR_LOCATIONS	COUNTRY	
SHIP_LOC_POSTAL_CODE	HR_LOCATIONS	POSTAL_CODE	
SHIP_LOC_REGION1	HR_LOCATIONS	REGION_1	
SHIP_LOC_REGION2	HR_LOCATIONS	REGION_2	
SHIP_VIA	PO_VENDORS	SHIP_VIA_LOOKUP_CODE	Stores shipping code
FREIGHT_TERMS	PO_VENDORS	FREIGHT_TERMS_LOOKUP_CODE	Stores freight Terms Code.
FOB	PO_VENDORS	FOB_LOOKUP_CODE	Freight On Board Code.
TERMS	AP_TERMS_TL	NAME	Term name.
SET_OF_BOOKS_NAME	GL_SETS_OF_BOOKS	NAME	Set of Books Name as in Oracle Applications.
SET_OF_BOOKS_SHORT_NAME	GL_SETS_OF_BOOKS	SHORT_NAME	Set of Books Short name.
CREDIT_STATUS	PO_VENDORS	CREDIT_STATUS_LOOKUP_CODE	Credit Status of the Vendor.
CREDIT_LIMIT	PO_VENDORS	CREDIT_LIMIT	Credit limit.
ALWAYS_DISCOUNT_FLAG	PO_VENDORS	ALWAYS_TAKE_DISC_FLAG	Discount flag.
PAY_DATE_BASIS	PO_VENDORS	PAY_DATE_BASIS_LOOKUP_CODE	Type of Payment Date Basis.
PAY_GROUP	PO_VENDORS	PAY_GROUP_LOOKUP_CODE	Pay Group.
PAYMENT_PRIORITY	PO_VENDORS	PAYMENT_PRIORITY	Payment Priority.

Document Field	Oracle Applications Table/View Name	Column Name	Description
INVOICE_CURRENCY_CODE	PO_VENDORS	INVOICE_CURRENCY_CODE	Invoice Currency.
PAYMENT_CURRENCY_CODE	PO_VENDORS	PAYMENT_CURRENCY_CODE	Payment Currency.
INVOICE_AMOUNT_LIMIT	PO_VENDORS	INVOICE_AMOUNT_LIMIT	Invoice Amount Limit.
EXCHANGE_DATE_CODE	PO_VENDORS	EXCHANGE_DATE_LOOKUP_CODE	Exchange Date Code.
HOLD_ALL_PAYMENTS	PO_VENDORS	HOLD_ALL_PAYMENTS_FLAG	Hold All Payments.
HOLD_FUTURE_PAYMENTS	PO_VENDORS	HOLD_FUTURE_PAYMENTS_FLAG	Hold Future Payments.
HOLD_REASON	PO_VENDORS	HOLD_REASON	Hold Reason.
DISTRIBUTION_SET_NAME	AP_DISTRIBUTION_SETS	DISTRIBUTION_SET_NAME	Distribution Set Name.
SUPPLIER LIABILITY_ACCOUNT	GL_CODE_COMBINATIONS_KFV	CONCATENATED_SEGMENTS	Concatenated GL code combination.
NUMBER_1099	PO_VENDORS	NUM_1099	Tax Payer ID.
TYPE_1099	PO_VENDORS	TYPE_1099	Type of 1099.
WITHHOLDING_STATUS	PO_VENDORS	WITHHOLDING_STATUS_LOOKUP_CODE	Withholding status type.
WITHHOLDING_START_DATE	PO_VENDORS	WITHHOLDING_START_DATE	Withholding Start Date.
IRS_ORGANIZATION_TYPE	PO_VENDORS	ORGANIZATION_TYPE_LOOKUP_CODE	IRS organization type.
VAT_CODE	PO_VENDORS	VAT_CODE	VAT Code.
START_DATE_ACTIVE	PO_VENDORS	START_DATE_ACTIVE	Active Start Date.
END_DATE_ACTIVE	PO_VENDORS	END_DATE_ACTIVE	Active End Date.
MINORITY_GROUP	PO_VENDORS	MINORITY_GROUP_LOOKUP_CODE	Minority Group.

Document Field	Oracle Applications Table/View Name	Column Name	Description
PAYMENT_METHOD	PO_VENDORS	PAYMENT_METHOD_LOOKUP_CODE	Payment Method.
BANK_ACCOUNT_NAME	PO_VENDORS	BANK_ACCOUNT_NAME	Name of the Supplier Bank Account.
BANK_ACCOUNT_NUMBER	PO_VENDORS	BANK_ACCOUNT_NUM	Number of the Supplier Bank Account.
BANK_NUMBER	PO_VENDORS	BANK_NUM	Bank Number for the Supplier.
BANK_ACCOUNT_TYPE	PO_VENDORS	BANK_ACCOUNT_TYPE	Type of Bank Account.
WOMEN_OWNED	PO_VENDORS	WOMEN_OWNED_FLAG	Indicates whether the supplier is a woman-owned business.
SMALL_BUSINESS	PO_VENDORS	SMALL_BUSINESS_FLAG	Indicates that the supplier is a small business.
STANDARD_INDUSTRY_CLASS	PO_VENDORS	STANDARD_INDUSTRY_CLASS	Standard Industry Classification Number.
HOLD_FLAG	PO_VENDORS	HOLD_FLAG	Hold flag.
PURCHASING_HOLD_REASON	PO_VENDORS	PURCHASING_HOLD_REASON	Reason for placing the supplier on purchasing hold
HOLD_BY_EMPLOYEE_NUMBER	PER_ALL_PEOPLE_F	EMPLOYEE_NUMBER	Hold By Employee Number.
PER_HOLD_BY_FULL_NAME	PER_ALL_PEOPLE_F	FULL_NAME	Full name of employee name.
HOLD_DATE	PO_VENDORS	HOLD_DATE	Date the supplier was placed on purchasing hold.
TERMS_DATE_BASIS	PO_VENDORS	TERMS_DATE_BASIS	Type of invoice payment schedule basis.
PRICE_TOLERANCE	PO_VENDORS	PRICE_TOLERANCE	Price Tolerance.
INSPECTION_REQUIRED	PO_VENDORS	INSPECTION_REQUIRED_FLAG	Indicates whether inspection is required or not.

Document Field	Oracle Applications Table/View Name	Column Name	Description
RECEIPT_REQUIRED_FLAG	PO_VENDORS	RECEIPT_REQUIRED_FLAG	Indicates whether shipment must be received before the invoice is paid.
QUANTITY_RECEIVED_TOLERANCE	PO_VENDORS	QTY_RCV_TOLERANCE	Quantity received tolerance percentage.
DAYS_EARLY_RECEIPT_ALLOWED	PO_VENDORS	DAYS_EARLY_RECEIPT_ALLOWED	Maximum acceptable number of days items can be received early.
DAYS_LATE_RECEIPT_ALLOWED	PO_VENDORS	DAYS_LATE_RECEIPT_ALLOWED	Maximum acceptable number of days items can be received late.
ALLOW_SUBSTITUTE_RECEIPTS	PO_VENDORS	ALLOW_SUBSTITUTE_RECEIPTS_FLAG	Indicates whether substitute items can be received in place of the ordered items.
ALLOW_UNORDERED_RECEIPTS	PO_VENDORS	ALLOW_UNORDERED_RECEIPTS_FLAG	Indicates whether unordered. Receipts are allowed
HOLD_UNMATCHED_INVOICES	PO_VENDORS	HOLD_UNMATCHED_INVOICES_FLAG	Indicates whether unmatched invoices should be put on hold.
EXCLUSIVE_PAYMENT	PO_VENDORS	EXCLUSIVE_PAYMENT_FLAG	Indicates exclusive payment.
TAX_VERIFICATION_DATE	PO_VENDORS	TAX_VERIFICATION_DATE	Tax verification date.
STATE_REPORTABLE	PO_VENDORS	STATE_REPORTABLE_FLAG	State Reportable flag.
FEDERAL_REPORTABLE	PO_VENDORS	FEDERAL_REPORTABLE_FLAG	Federal Reportable flag.
OFFSET_VAT	PO_VENDORS	OFFSET_VAT_CODE	Offset Vat Code.
VAT_REGISTRATION_NUMBER	PO_VENDORS	VAT_REGISTRATION_NUM	Tax registration number
AUTO_CALCULATE_INTEREST	PO_VENDORS	AUTO_CALCULATE_INTEREST_FLAG	Indicates whether interest is to be automatically calculated.

Document Field	Oracle Applications Table/View Name	Column Name	Description
VALIDATION_NUMBER	PO_VENDORS	VALIDATION_NUMBER	Validation Number
EXCLUDE_FREIGHT_FROM_DISCOUNT	PO_VENDORS	EXCLUDE_FREIGHT_FROM_DISCOUNT	Exclude supplier freight from discount amount.
TAX_REPORTING_NAME	PO_VENDORS	TAX_REPORTING_NAME	Tax Reporting Method name.
CHECK_DIGITS	PO_VENDORS	CHECK_DIGITS	Check number used by Payables.
BANK_NUM	PO_VENDORS	BANK_NUM	Bank number for Accounts Payable.
AUTO_TAX_CALCULATION	PO_VENDORS	AUTO_TAX_CALC_FLAG	Auto Tax Calculation flag.
AUTO_TAX_CALCULATION_OVERRIDE	PO_VENDORS	AUTO_TAX_CALC_OVERRIDE	Allow Calculation Level override.
AMOUNT_INCLUDES_TAX	PO_VENDORS	AMOUNT_INCLUDES_TAX_FLAG	Distribution amounts include tax.
BANK_CHARGE_BEARER	PO_VENDORS	BANK_CHARGE_BEARER	Indicator of whether this supplier bears bank charges.
BANK_BRANCH_TYPE	PO_VENDORS	BANK_BRANCH_TYPE	Indicates which list the bank routing number is on. Valid values are ABA, CHIPS, SWIFT and OTHER.

1.1 BANK_ACCOUNTS

Document Field	Oracle Applications Table/View Name	Column Name	Description
BANK_ACCOUNT_NAME	AP_BANK_ACCOUNTS_ALL	BANK_ACCOUNT_NAME	Bank Account Name.
BANK_ACCOUNT_NUM	AP_BANK_ACCOUNTS_ALL	BANK_ACCOUNT_NUM	Bank Account Number.
CURRENCY_CODE	AP_BANK_ACCOUNTS_ALL	CURRENCY_CODE	Currency code.
BANK_NAME	AP_BANK_BRANCHES	BANK_NAME	Bank Name.

Document Field	Oracle Applications Table/View Name	Column Name	Description
BANK_NUMBER	AP_BANK_BRANCHES	BANK_NUMBER	Bank Number.
BANK_BRANCH_NAME	AP_BANK_BRANCHES	BANK_BRANCH_NAME	Bank Branch Name.
BANK_NUM	AP_BANK_BRANCHES	BANK_NUM	Bank Branch Number.
END_DATE	AP_BANK_ACCOUNTS_USES_ALL	START_DATE	Start Date.
START_DATE	AP_BANK_ACCOUNTS_USES_ALL	END_DATE	End Date.
PRIMARY	AP_BANK_ACCOUNT_USES_ALL	PRIMARY_FLAG	Primary Bank Account flag.
ORGANIZATION_NAME	ORG_ORGANIZATION_DEFINITIONS	ORGANIZATION_NAME	Name of the organization.

1.2 VENDOR SITES

Document Field	Oracle Applications Table/View Name	Column Name	Description
VENDOR_SITE_ID	PO_VENDOR_SITES_ALL	VENDOR_SITE_ID	Vendor Site Unique Identifier.
VENDOR_SITE_CODE	PO_VENDOR_SITES_ALL	VENDOR_SITE_CODE	Vendor Site Name.
ADDRESS_LINE1	PO_VENDOR_SITES_ALL	ADDRESS_LINE1	First line of Vendor Site Address.
ADDRESS_LINE2	PO_VENDOR_SITES_ALL	ADDRESS_LINE2	Second line of Vendor site address.
ADDRESS_LINE3	PO_VENDOR_SITES_ALL	ADDRESS_LINE3	Third line of Vendor Site Address.
CITY	PO_VENDOR_SITES_ALL	CITY	Town or city.
STATE	PO_VENDOR_SITES_ALL	STATE	State.
ZIP	PO_VENDOR_SITES_ALL	ZIP	Zip Code.
PROVINCE	PO_VENDOR_SITES_ALL	PROVINCE	Province.

Document Field	Oracle Applications Table/View Name	Column Name	Description
COUNTRY	PO_VENDOR_SITES_ALL	COUNTRY	Country.
AREA_CODE	PO_VENDOR_SITES_ALL	AREA_CODE	Area Code.
PHONE	PO_VENDOR_SITES_ALL	PHONE	Phone number.
ALTERNATE_ADDRESS_LINE	PO_VENDOR_SITES_ALL	ADDRESS_LINES_AL	Alternate address line.
PURCHASING_SITE	PO_VENDOR_SITES_ALL	PURCHASING_SITE_FLAG	Indicates whether you can purchase from this site.
RFQ_ONLY_SITE	PO_VENDOR_SITES_ALL	RFQ_ONLY_SITE_FLAG	Indicates whether you can only send RFQ to this site.
PAY_SITE	PO_VENDOR_SITES_ALL	PAY_SITE_FLAG	Indicates whether you can send payments to this site.
ATTENTION_AR	PO_VENDOR_SITES_ALL	ATTENTION_AR_FLAG	Indicates whether the payments should be sent to the Account Receivables department.
CUSTOMER_NUMBER	PO_VENDOR_SITES_ALL	CUSTOMER_NUMBER	Customer number associated with the supplier.
BILL_LOC_ADDRESS_LINE1	HR_LOCATIONS	ADDRESS_LINE_1	Indicates Billing Location.
BILL_LOC_ADDRESS_LINE2	HR_LOCATIONS	ADDRESS_LINE_2	
BILL_LOC_ADDRESS_LINE3	HR_LOCATIONS	ADDRESS_LINE_3	
BILL_LOC_TOWN_OR_CITY	HR_LOCATIONS	TOWN_OR_CITY	
BILL_LOC_COUNTRY	HR_LOCATIONS	COUNTRY	
BILL_LOC_POSTAL_CODE	HR_LOCATIONS	POSTAL_CODE	
BILL_LOC_REGION1	HR_LOCATIONS	REGION_1	Indicates the County.

Document Field	Oracle Applications Table/View Name	Column Name	Description
BILL_LOC_REGION2	HR_LOCATIONS	REGION_2	Indicates the State Code.
SHIP_LOC_ADDRESS_LINE1	HR_LOCATIONS	ADDRESS_LINE_1	Shipping Location.
SHIP_LOC_ADDRESS_LINE2	HR_LOCATIONS	ADDRESS_LINE_2	
SHIP_LOC_ADDRESS_LINE3	HR_LOCATIONS	ADDRESS_LINE_3	
SHIP_LOC_TOWN_OR_CITY	HR_LOCATIONS	TOWN_OR_CITY	
SHIP_LOC_REGION1	HR_LOCATIONS	REGION_1	
SHIP_LOC_REGION2	HR_LOCATIONS	REGION_2	
SHIP_LOC_COUNTRY	HR_LOCATIONS	COUNTRY	
SHIP_LOC_POSTAL_CODE	HR_LOCATIONS	POSTAL_CODE	
SHIP_VIA	PO_VENDOR_SITES_ALL	SHIP_VIA_LOOKUP_CODE	Shipping Code.
FREIGHT_TERMS	PO_VENDOR_SITES_ALL	FREIGHT_TERMS_LOOKUP_CODE	Freight terms Code.
FOB	PO_VENDOR_SITES_ALL	FOB_LOOKUP_CODE	Default fRee On Board type.
INACTIVE_DATE	PO_VENDOR_SITES_ALL	INACTIVE_DATE	Inactive Date For Record.
FAX	PO_VENDOR_SITES_ALL	FAX	Fax Number.
FAX_AREA_CODE	PO_VENDOR_SITES_ALL	FAX_AREA_CODE	Fax Area Code.
TELEX	PO_VENDOR_SITES_ALL	TELEX	Telex.
PAYMENT_METHOD	PO_VENDOR_SITES_ALL	PAYMENT_METHOD_LOOKUP_CODE	Payment Method.
BANK_ACCOUNT_NAME	PO_VENDOR_SITES_ALL	BANK_ACCOUNT_NAME	Name of the Supplier Bank Account.

Document Field	Oracle Applications Table/View Name	Column Name	Description
BANK_ACCOUNT_NUMBER	PO_VENDOR_SITES_ALL	BANK_ACCOUNT_NUMBER	Number of the Supplier Bank Account.
BANK_NUM	PO_VENDOR_SITES_ALL	BANK_NUM	Bank Branch Number.
BANK_ACCOUNT_TYPE	PO_VENDOR_SITES_ALL	BANK_ACCOUNT_TYPE	Type of bank account.
TERMS_DATE_BASIS	PO_VENDOR_SITES_ALL	TERMS_DATE_BASIS	Type of payment date basis.
VAT_CODE	PO_VENDOR_SITES_ALL	VAT_CODE	VAT code.
DISTRIBUTION_SET_NAME	AP_DISTRIBUTION_SETS	DISTRIBUTION_SET_NAME	Distribution set name.
SUPPLIER_LIABILITY_ACCOUNT	GL_CODE_COMBINATIONS_KFV	CONCATENATED_SEGMENTS	Concatenated GL Code combination.
PREPAY_ACCOUNT	GL_CODE_COMBINATIONS_KFV	CONCATENATED_SEGMENTS	Concatenated GL Code combination.
PAY_GROUP	PO_VENDOR_SITES_ALL	PAY_GROUP_LOOKUP_CODE	Pay Group.
PAYMENT_PRIORITY	PO_VENDOR_SITES_ALL	PAYMENT_PRIORITY	Payment Priority.
TERMS	AP_TERMS	NAME	Term name.
INVOICE_AMOUNT_LIMIT	PO_VENDOR_SITES_ALL	INVOICE_AMOUNT_LIMIT	Invoice Amount Limit.
PAY_DATE_BASIS	PO_VENDOR_SITES_ALL	PAY_DATE_BASIS_LOOKUP_CODE	Type of Payment Date Basis.
ALWAYS_DISCOUNT_FLAG	PO_VENDOR_SITES_ALL	ALWAYS_TAKE_DISC_FLAG	If discount is applicable for the site.
INVOICE_CURRENCY_CODE	PO_VENDOR_SITES_ALL	INVOICE_CURRENCY_CODE	Invoice Currency
PAYMENT_CURRENCY_CODE	PO_VENDOR_SITES_ALL	PAYMENT_CURRENCY_CODE	Payment Currency.
HOLD_ALL_PAYMENTS	PO_VENDOR_SITES_ALL	HOLD_ALL_PAYMENTS_FLAG	Hold All Payments.

Document Field	Oracle Applications Table/View Name	Column Name	Description
HOLD_FUTURE_PAYMENTS	PO_VENDOR_SITES_ALL	HOLD_FUTURE_PAYMENTS_FLAG	Hold Future Payments.
HOLD_REASON	PO_VENDOR_SITES_ALL	HOLD_REASON	Hold Reason.
HOLD_UNMATCHED_INVOICES	PO_VENDOR_SITES_ALL	HOLD_UNMATCHED_INVOICES_FLAG	Indicates whether unmatched Invoices should be put on hold.
EXCLUSIVE_PAYMENT	PO_VENDOR_SITES_ALL	EXCLUSIVE_PAYMENT_FLAG	Indicates exclusive payment.
TAX_REPORTING_SITE	PO_VENDOR_SITES_ALL	TAX_REPORTING_SITE_FLAG	Tax reporting site flag.
VALIDATION_NUMBER	PO_VENDOR_SITES_ALL	VALIDATION_NUMBER	Validation number.
EXCLUDE_FREIGHT_FROM_DISCOUNT	PO_VENDOR_SITES_ALL	EXCLUDE_FREIGHT_FROM_DISCOUNT	Exclude supplier freight from discount amount.
VAT_REGISTRATION_NUMBER	PO_VENDOR_SITES_ALL	VAT_REGISTRATION_NUMBER	VAT registration number.
OFFSET_VAT	PO_VENDOR_SITES_ALL	OFFSET_VAT_CODE	Offset VAT code.
ORGANIZATION_NAME	ORG_ORGANIZATION_DEFINITIONS	ORGANIZATION_NAME	Name of organization corresponding to the operating unit.
ORGANIZATION_CODE	ORG_ORGANIZATION_DEFINITIONS	ORGANIZATION_CODE	Organization code for operating unit.
CHECK_DIGITS	PO_VENDOR_SITES_ALL	CHECK_DIGITS	Check number used by Payables.
BANK_NUMBER	PO_VENDOR_SITES_ALL	BANK_NUM	Bank number for the supplier.
ADDRESS_LINE4	PO_VENDOR_SITES_ALL	ADDRESS_LINE4	Address line.
COUNTY	PO_VENDOR_SITES_ALL	COUNTY	County.
ADDRESS_STYLE	PO_VENDOR_SITES_ALL	ADDRESS_STYLE	Style of address.

Document Field	Oracle Applications Table/View Name	Column Name	Description
ALLOW_AWT	PO_VENDOR_SITES_ALL	ALLOW_AWT_FLAG	Withhold tax.
WITHHOLDING_TAX_GROUP	AP_AWT_GROUPS	NAME	Withholding tax group name.
AP_TAX_ROUNDING_RULE	PO_VENDOR_SITES_ALL	AP_TAX_ROUNDING_RULE	Tax Rounding Rule.
AUTO_TAX_CALCULATION	PO_VENDOR_SITES_ALL	AUTO_TAX_CALC_FLAG	Auto Tax Calculation Flag.
AUTO_TAX_CALCULATION_OVERRIDE	PO_VENDOR_SITES_ALL	AUTO_TAX_CALC_OVERRIDE	
AMOUNT_INCLUDES_TAX	PO_VENDOR_SITES_ALL	AMOUNT_INCLUDES_TAX_FLAG	
BANK_CHARGE_BEARER	PO_VENDOR_SITES_ALL	BANK_CHARGE_BEARER	Indicator of whether this supplier bears bank charges
BANK_BRANCH_TYPE	PO_VENDOR_SITES_ALL	BANK_BRANCH_TYPE	Lists the bank routing number is on. Valid values are ABA, CHIPS, SWIFT and OTHER.
PAY_ON_CODE	PO_VENDOR_SITES_ALL	PAY_ON_CODE	
PAY_ON_RECEIPT_SUMMARY_CODE	PO_VENDOR_SITES_ALL	PAY_ON_RECEIPT_SUMMARY_CODE	Identifies how to consolidate receipts to create invoices.

1.2.1 SITE_BANK_ACCOUNTS

Document Field	Oracle Applications Table/View Name	Column Name	Description
BANK_ACCOUNT_NAME	AP_BANK_ACCOUNTS_ALL	BANK_ACCOUNT_NAME	Bank account name.
BANK_ACCOUNT_NUM	AP_BANK_ACCOUNTS_ALL	BANK_ACCOUNT_NUM	Bank account number.
CURRENCY_CODE	AP_BANK_ACCOUNTS_ALL	CURRENCY_CODE	Currency code.

Document Field	Oracle Applications Table/View Name	Column Name	Description
BANK_NAME	AP_BANK_BRANCHES	BANK_NAME	Bank name.
BANK_NUMBER	AP_BANK_BRANCHES	BANK_NUMBER	Bank number.
BANK_BRANCH_NAME	AP_BANK_BRANCHES	BANK_BRANCH_NAME	Bank branch name.
BANK_NUM	AP_BANK_BRANCHES	BANK_NUM	Bank number.
END_DATE	AP_BANK_ACCOUNT_USES_ALL	START_DATE	Start date.
START_DATE	AP_BANK_ACCOUNT_USES_ALL	END_DATE	End date.
PRIMARY	AP_BANK_ACCOUNT_USES_ALL	PRIMARY_FLAG	Primary bank account flag.
ORGANIZATION_NAME	ORG_ORGANIZATION_DEFINITIONS	ORGANIZATION_NAME	Organization name.

1.2.2 VENDOR_CONTACTS

Document Field	Oracle Applications Table/View Name	Column Name	Description
VENDOR_SITE_ID	PO_VENDOR_CONTACTS	VENDOR_SITE_ID	Vendor Site unique identifier.
FIRST_NAME	PO_VENDOR_CONTACTS	FIRST_NAME	First name
MIDDLE_NAME	PO_VENDOR_CONTACTS	MIDDLE_NAME	Middle name.
LAST_NAME	PO_VENDOR_CONTACTS	LAST_NAME	Last name.
PREFIX	PO_VENDOR_CONTACTS	PREFIX	Prefix.
TITLE	PO_VENDOR_CONTACTS	TITLE	Title.
MAIL_STOP	PO_VENDOR_CONTACTS	MAIL_STOP	Mail stop.
AREA_CODE	PO_VENDOR_CONTACTS	AREA_CODE	Area Code of the contact.
PHONE	PO_VENDOR_CONTACTS	PHONE	Phone number of the contact.
INACTIVE_DATE	PO_VENDOR_CONTACTS	INACTIVE_DATE	Date from which the contact is inactive.

Human Resources Predefined Transaction Services

■ Overview	222
■ Receive Employee Service	223

Overview

This chapter describes the predefined transaction services provided in the Oracle Applications Adapter’s 10.7SC Human Resource package.

The table below shows the predefined transaction services organized by Oracle Applications module. This chapter presents the transactions in alphabetical order.

Oracle Applications Module	Predefined Transactions
Human Resource	■ “Receive Employee Service” on page 223

For more information about using the predefined transaction services, see [Chapter 1, “Predefined Transaction Services” on page 17](#).

Receive Employee Service

The name of this service is:

WmOAHR107SC.humanResource107SC.intoOA.employee:receiveEmployee

This service imports new and modified employee data into the Oracle Applications production tables. This predefined transaction service can handle multiple employee records in a single business document. Each record in the business document is inserted or updated according to the underlying Oracle Applications APIs and the data in the business document's columns. Each business document must contain an assignment child in the business document structure. Business documents can contain new employee records, modified employee records, or both.

Database Scripts

This service uses the following database scripts:

Database Script	Description
wm_install_into_employee.sql	Runs all the scripts listed below, except the uninstall script.
wm_into_employee_pkg.sql	<p>Installs following components, which call the concurrent process to submit the employee information.</p> <ul style="list-style-type: none">■ WM_EMP_IMP_HANDLER_PKG. WM_HANDLE_EMP■ WM_EMPLOYEE_ASG_IMPORT_PKG. WM_EMP_CREATE_ASG■ WM_EMPLOYEE_ASG_IMPORT_PKG. WM_EMP_UPDATE_ASG
wm_drop_into_employee.sql	Uninstalls all components created by wm_install_into_employee.sql.

For more information about using database scripts, see [“Database Scripts”](#) on page 26.

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definition:

- `EmployeeTransactions107SC.txp`

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow executes as follows:

- `specifyDefaultSettings` specifies the default parameter settings required for service execution. You should change these settings accordingly.
- `importEmployee` imports data into the production table using Oracle Applications HRMS APIs. It also uses the `execEmployeeProg`, `execEmpAssignCreateProg`, and `execEmpAssignUpdateProg` services to execute the required internal HRMS APIs. Since a single database transaction spans over multiple service calls for the inbound Employee transaction, set the `AutoCommit` setting to `OFF` to control the entire transaction through the `webMethods` service. If execution is successful, it commits the entire transaction for Employee Header information and Employee Assignments information to the database. If the status of the execution is `FAILED` due to exceptions thrown by HRMS APIs or due to data error, it rolls back the entire transaction. When the execution status is `FAILED`, the HRMS returns the business error messages returned by the HRMS APIs captures them in the `empImportResults` document, which can then be sent to the trading partners.
 - `execEmployeeProg` invokes the stored procedure `WM_EMP_IMP_HANDLER_PKG.WM_HANDLE_EMP`. This procedure calls the corresponding HRMS APIs to import Employee data into the Oracle Applications system. This service gives a status ID, execution status message (for normal concurrent program completion), database stored procedure error message (if an exception occurs in the stored procedure execution), data errors (if business errors are captured by HRMS APIs), and other output values required to be passed for creating, updating, or correcting the Employee Assignments. If this service returns the status of `FAILED`, the entire transaction is rolled back and the next Employee information is processed.
 - `execEmployeeProg` invokes the stored procedure `WM_EMP_IMP_HANDLER_PKG.WM_HANDLE_EMP` that calls the corresponding HRMS APIs to import Employee data into the Oracle Applications system. This service produces a status ID, execution status message for normal concurrent program completion, a database stored procedure error message (if an exception occurs in the stored procedure execution), data errors (if the HRMS APIs capture any business errors), and any other output values that must be passed to create, update, or correct the Employee Assignments. If this service

returns a status of FAILED, it rolls back the entire transaction and processes the next Employee information.

- **execEmpAssignUpdateProg** invokes the stored procedure WM_EMPLOYEE_ASG_IMPORT_PKG.WM_EMP_UPDATE_ASG. This stored procedure updates existing Employee Assignments using the corresponding HRMS APIs to import updated Employee Assignments information into the Oracle Applications system. This service produces status ID, execution status message for normal concurrent program completion, database stored procedure error message (if an exception occurs in the stored procedure execution), data errors (if the HRMS APIs capture any business errors), and other output values. If this service returns the status of FAILED, it rolls back the entire transaction and processes the next Employee information.

For more details on receive transactions, see [“Overview of Receive Service Transaction Processing” on page 31](#).

Business Document Structure

This service uses the business EmployeeBizDoc. Its structure is as follows:

- 1.0 EMPLOYEE
 - 1.1 EMPLOYEE_ASSIGNMENTS

1.0 EMPLOYEE

Field Name	Description
ACTION	Required. Indicates whether you are creating a new employee or modifying existing employee information. Valid values are New or Modify.
ACTION_SUBTYPE	If you specify Modify in the ACTION field, use UPDATE or CORRECTION in this field.
EMPLOYEE_NUMBER	Number assigned to a person who has been an employee. For new employees, leave this field blank to automatically generate a new employee number..
EFFECTIVE_START_DATE	Required. Effective start date.
EFFECTIVE_END_DATE	Effective end date.
BUSINESS_GROUP_NAME	Required. Name of the business group to which the Employee belongs.
PERSON_TYPE_CODE	Required. Person type as defined by user. Examples are Employee, Consultant, and Temporary.
LAST_NAME	

Field Name	Description
FIRST_NAME	
START_DATE	Required. Start date of hire for the Employee.
APPLICANT_NUMBER	Number assigned those persons who were ever classed as an applicant.
COMMENT	Comment.
DATE_EMPLOYEE_DATA_VERIFIED	Date when the employee last verified the data.
DATE_OF_BIRTH	Required. Date of birth.
DATE_OF_DEATH	Date of death.
EMAIL_ADDRESS	Email address.
EXPENSE_CHECK_SEND_TO_ADDRESS	Mailing address.
FULL_NAME	Full name.
KNOWN_AS	Preferred name, if different from first name.
PRIMARY_ADDRESS_FLAG	Required. The primary address.
ADDRESS_STYLE	Required. Address style for a specific region.
ADDRESS_TYPE	Required. Address type. Example values are H (Home), M (Mailing), W (weekend), T (Term), and V (Vacation).
ADDRESS_LINE1	Required. Corresponding address of the employee.
ADDRESS_LINE2	
ADDRESS_LINE3	
CITY	
STATE	
POSTAL_CODE	
COUNTY	
COUNTRY	Required.
PHONE_TYPE	Required. Type of phone. Example values are H1 (Home), W1 (Work) , and M (Mobile). This field allows for customized field values.
TELEPHONE_NUMBER	Phone number.
MARITAL_STATUS	Marital status of the person.
MIDDLE_NAME	Middle name of the person.
NATIONALITY	Nationality of the Person.

Field Name	Description
NATIONAL_IDENTIFIER	Number by which a person is identified in a given legislation.
PREVIOUS_LAST_NAME	Previous last name of person.
REGISTERED_DISABLED_FLAG	Indicates whether the person is classified as disabled.
SEX	Required. Legal gender.
TITLE	Title. Examples are Dr., Mr., and Mrs.
SUFFIX	Suffix after the surname. Examples are Sr, Jr, and III.
WORK_TELEPHONE	Work telephone number.

1.1 EMPLOYEE_ASSIGNMENTS

Field Name	Description
ACTION	Required. Indicates whether you are adding a new employee or modifying information for an existing employee.
ACTION_SUBTYPE	Indicates a value of UPDATE or CORRECTION in the case of modification.
ASSIGNMENT_NUMBER	Employee Assignment Number. Required when the employee information is modified.
ASSIGNMENT_TITLE	Title of the assignment.
EFFECTIVE_START_DATE	Required. Effective start date.
EFFECTIVE_END_DATE	Required. Effective end date.
BUSINESS_GROUP_NAME	Required. Name of the business group to which the employee belongs
RECRUITER_NAME	Name of the recruiter
GRADE	Grade of the employee.
POSITION	Position of the employee.
JOB	Job Name of the employee.
USER_ASSIGNMENT_STATUS	Required. Assignment status of the employee.
PAYROLL_NAME	Payroll name of the Employee
LOCATION_ADDRESS_LINE_1	Location address for the Employee Assignment.
LOCATION_ADDRESS_LINE_2	
LOCATION_ADDRESS_LINE_3	

Field Name	Description
LOCATION_CITY	
LOCATION_COUNTY	
LOCATION_STATE	
LOCATION_POSTAL_CODE	
LOCATION_COUNTRY	
PERSON_REFERRED_BY	Person name who referred the employee.
SUPERVISOR_NAME	Employee supervisor's name.
PROGRESSION_POINT_NAME_NUMBER	Number indicating the progression of the pay scale and their incremental order number.
RECRUITMENT_ACTIVITY_NAME	Recruitment activity name for the employee.
SOURCE_ORGANIZATION_NAME	Name of the source organization, such as Division or Business Unit.
ORGANIZATION_NAME	Required. Name of the organization.
GROUP_NAME	Name of the group to which the employee belongs.
VACANCY_NAME	Name of the vacancy position.
PAY_BASIS	Basis of payment, such as annual, hourly, monthly and period.
SALARY_APPROVED_FLAG	Indicates payment approval.
RATE_BASIS	Basis of rate.
SALARY_UOM	Unit of measure of the salary.
PROPOSED_SALARY	Salary proposed to the employee.
ASSIGNMENT_TYPE	Required. Type of the assignment.
ASSIGNMENT_PRIMARY_FLAG	Required. The primary assignment.
CURRENT_EMPLOYER	Name of the Current employer.
ASSIGNMENT_CHANGE_REASON	Reason for changing the current assignment.
COMMENT	Comment.
DATE_PROBATION_END	Date indicating the end of probation period for the employee.
GL_ACCOUNT_NUMBER	General Ledger account number.
EMPLOYMENT_CATEGORY	Category of the employment, such as Casual, Full-time, and Intern.

Field Name	Description
NORMAL_WORKING_FREQUENCY	Normal frequency at which the employee works, such as day, hour, month, week, and ear.
INTERNAL_ADDRESS_LINE	Internal address details such as floor or office number.
MANAGER_FLAG	Indicates if the employee is a manager for the assignment.
NORMAL_HOURS	Normal hours of work.
PERFORMANCE_REVIEW_PERIOD	Period of performance review
PERF_REVIEW_PERIOD_FREQUENCY	Frequency units. Used with PERF_REVIEW_PERIOD to define time between reviews.
TERMINATION_ACCEPTED_PERSON	Person accepting the termination of the employee.
ACCEPTED_TERMINATION_DATE	Date when termination of employment was accepted.
ACTUAL_TERMINATION_DATE	Actual date of termination of employment.
FINAL_PAYROLL_PROCESSED_DATE	Date when the final payroll for the employee is processed.
LAST_STANDARD_PROCESS_DATE	Last date for including the person in any standard payroll processing.
LEAVING_REASON	Reason for termination of assignment.
NOTIFIED_TERMINATION_DATE	Date when termination of employment was noted.
PROJECTED_TERMINATION_DATE	Projected employment termination date, used for information only.
PROBATION_PERIOD	Probation period for the employee.
PROBATION_UNIT	Unit of measure for the probation period.
SALARY_REVIEW_PERIOD	Period by which the salary is reviewed.
SALARY_REVIEW_PERIOD_FREQUENCY	Frequency by which the salary period is reviewed.
SET_OF_BOOKS_NAME	General Ledger set of books name.
ASSIGNMENT_SOURCE_TYPE	Employment source name such as Advertisement, Agency, walk-in, and career-fair.
WORK_TIME_NORMAL_FINISH	Time when work normally finishes.
WORK_TIME_NORMAL_START	Time when work normally starts.

Manufacturing Predefined Transaction Services

■ Overview	233
■ Query BOM Revision Service	234
■ Query BOM Routing Service	236
■ Query Engineering BOM Service	243
■ Query Item Service	244
■ Query Item Category Service	245
■ Query Manufacturing BOM Service	247
■ Query On-Hand Quantity Service	249
■ Receive Bills and Route Service	252
■ Receive Customer Item Service	266
■ Receive Customer Item Cross Reference Service	271
■ Receive Inventory Transaction Service	275
■ Receive Item Service	284
■ Receive Master Schedule Service	303
■ Receive Move Transaction Service	306
■ Receive Quality Collection Service	310
■ Receive Replenishment Service	314
■ Receive Resource Transaction Service	320
■ Send Engineering BOM Service	325

- Send Item Service336
- Send Manufacturing BOM Service359

Overview

This chapter describes the predefined transaction services provided in the Oracle Applications Adapter's 10.7SC Manufacturing package.

The table below shows the predefined transaction services organized by Oracle Applications module. This chapter lists the transactions in alphabetical order.

Oracle Applications Module	Predefined Transactions
Bills of Material	<ul style="list-style-type: none"> ■ "Query BOM Revision Service" on page 234 ■ "Query BOM Routing Service" on page 236 ■ "Query Manufacturing BOM Service" on page 247 ■ "Receive Bills and Route Service" on page 252 ■ "Send Manufacturing BOM Service" on page 359
Engineering	<ul style="list-style-type: none"> ■ "Query Engineering BOM Service" on page 243 ■ "Send Engineering BOM Service" on page 325
Inventory	<ul style="list-style-type: none"> ■ "Query Item Service" on page 244 ■ "Query Item Category Service" on page 245 ■ "Query On-Hand Quantity Service" on page 249 ■ "Receive Customer Item Cross Reference Service" on page 271 ■ "Receive Customer Item Service" on page 266 ■ "Receive Inventory Transaction Service" on page 275 ■ "Receive Item Service" on page 284 ■ "Receive Replenishment Service" on page 314 ■ "Send Item Service" on page 336
Master Scheduling	<ul style="list-style-type: none"> ■ "Receive Master Schedule Service" on page 303
Quality Collection	<ul style="list-style-type: none"> ■ "Receive Quality Collection Service" on page 310
Work in Process	<ul style="list-style-type: none"> ■ "Receive Move Transaction Service" on page 306 ■ "Receive Resource Transaction Service" on page 320

For more information about using the predefined transaction services, see [Chapter 1, "Predefined Transaction Services"](#) on page 17.

Query BOM Revision Service

The name of this service is:
WmOAMFG107SC.billsOfMaterial107SC.queryOA.BOMRevision:queryBOMRevision

This service retrieves Bills of Material revisions from Oracle Manufacturing based on the following parameters:

- ORGANIZATION_NAME: Name of the organization.
- ITEM: Item number.
- REVISION: Revision of the parent item.
- EFFECTIVITY_DATE_FROM: Beginning effective date.
- EFFECTIVITY_DATE_TO: Ending effective date.

If you do not enter any of the above parameters, the query will retrieve all rows from the Oracle Applications database.

Database Scripts

This service uses the following database scripts:

Script	Description
wm_install_from_bomrevision.sql	Runs the scripts listed below, except the uninstall script.
wm_from_bomrevision_vw.sql	Creates the following required view component for BOM revision transactions: <ul style="list-style-type: none">■ WM_BOM_REVISIONS_QRY_VW
wm_drop_from_bomrevision.sql	Uninstalls all components created by wm_install_from_bomrevision.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service uses the following transaction definition:

- queryBOMRevisionTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

You can query BOMRevision data using the following services:

- `specifyDefaultSettings` specifies the default parameter settings required for service execution. You should change these settings accordingly.
- `queryBOMRevisionTxn` queries the Oracle Applications database for any BOMRevision Transaction to be processed. The number of records returned depends on the parameter value specified.

For more details on query transactions, see [“Using Query Services” on page 36](#).

Business Document Structure

This service uses the following business document structure:

- BOM_REVISIONS

BOM_REVISIONS

Document Field	Oracle Applications Table/View Name	Column Name	Description
ITEM	MTL_SYSTEM_ITEMS_KFV	CONCATENATE_D_SEGMENTS	Concatenated flexfield segment constituting the item number.
INV_ORG_NAME	HR_ALL_ORGANIZATION_UNITS	NAME	Organization name.
REVISION	MTL_ITEM_REVISIONS	REVISION	Item revision code.
CHANGE_NOTICE	MTL_ITEM_REVISIONS	CHANGE_NOTICE	Engineering change order number.
ECN_INITIATION_DATE	MTL_ITEM_REVISIONS	ECN_INITIATION_DATE	ECN initiation date.
IMPLEMENTATION_DATE	MTL_ITEM_REVISIONS	IMPLEMENTATION_DATE	ECN implementation date.
EFFECTIVITY_DATE	MTL_ITEM_REVISIONS	EFFECTIVITY_DATE	Revision effective date.

Query BOM Routing Service

The name of this service is:

```
WmOAMFG107SC.billsOfMaterial107SC.queryOA.BOMRouting:queryBOMRouting
```

This service retrieves information on routing of parent, component, substitute items, and routing revisions based on querying the following parameters:

- ORGANIZATION_NAME: Name of the organization.
- ITEM_NUMBER: Item number.
- REVISION: Revision of the parent item.
- EFFECTIVITY_DATE_FROM: Beginning effective date.
- EFFECTIVITY_DATE_TO: Ending effective date.

If none of the above parameters are entered, the query will retrieve all rows from the Oracle Applications database.

Database Scripts

This service uses the following database scripts:

Script	Description
wm_install_from_bomrouting.sql	Runs the scripts listed below, except the uninstall script.
wm_from_bomrouting_vw.sql	Creates the following required view components for BOM Routing transactions: <ul style="list-style-type: none">■ WM_BOM_OP_ROUTING_QRY_VW■ WM_MTL_RTG_ITEM_REVS_QRY_VW■ WM_BOM_OP_SEQUENCES_QRY_VW■ WM_BOM_OP_RESOURCES_QRY_VW
wm_drop_from_bomrouting.sql	Uninstalls all components created by wm_install_from_bomrouting.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service uses the following transaction definition:

- queryBOMRoutingTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow queryBOMRouting executes as follows:

- specifyDefaultSettings specifies the default parameter settings required for service execution. You should change these settings accordingly.
- queryBOMRoutingTxn queries the Oracle Applications database for any BOM Routings information matching the parameter values. The parameters are defined as the input to this service.

For more details on query transactions, see [“Using Query Services” on page 36](#).

Business Document Structure

This service uses the following business document structure:

- 1.0. BOM_OP_ROUTINGS
 - 1.1. MTL_RTG_ITEM_REVS
 - 1.2. BOM_OP_SEQUENCES
 - 1.2.1. BOM_OP_RESOURCES

1.0. BOM_PO_ROUTINGS

Document Field	Oracle Applications Table/View Name	Column Name	Description
WEB_TRANSACTION_ID			
DOCUMENT_TYPE			Use BOM_ROUTING.
DOCUMENT_STATUS			Use QUERY.
ROUTING_SEQUENCE_ID	BOM_OPERATIONAL_ROUTINGS	ROUTING_SEQUENCE_ID	Routing unique identifier.

Document Field	Oracle Applications Table/View Name	Column Name	Description
ORGANIZATION_NAME	HR_ALL_ORGANIZATION_UNITS	NAME	
ASSEMBLY_ITEM_NUMBER	MTL_SYSTEM_ITEMS_B_KFV	CONCATENATE_D_SEGMENTS	Material item number.
ALTERNATE_ROUTING_DESIGNATOR	BOM_OPERATIONAL_ROUTINGS	ALTERNATE_ROUTING_DESIGNATOR	Alternate routing designator.
ROUTING_TYPE	BOM_OPERATIONAL_ROUTINGS	ROUTING_TYPE	Required for Insert. Use: 1 for Manufacturing, or 2 for Engineering
COMMON_ITEM_NUMBER	MTL_SYSTEM_ITEMS_B_KFV	CONCATENATE_D_SEGMENTS	Concatenated item segments of the common routing.
COMMON_ORG_NAME	HR_ALL_ORGANIZATION_UNITS	NAME	Derived from columns COMMON_ITEM_NUMBER, COMMON_ORG_NAME, and ALTERNATE_ROUTING_DESIGNATOR.
ROUTING_COMMENT	BOM_OPERATIONAL_ROUTINGS	ROUTING_COMMENT	Specific comment about routing.
COMPLETION_SUBINVENTORY	BOM_OPERATIONAL_ROUTINGS	COMPLETION_SUBINVENTOR Y	Destination subinventory for assembly.
LOCATION_NAME	MTL_ITEM_LOCATIONS	DESCRIPTION	Material item locations.

1.1. MTL_RTG_ITEM_REVS

Document Field	Oracle Applications Table/View Name	Column Name	Description
ROUTING_SEQUENCE_ID	BOM_OPERATIONAL_ROUTINGS	ROUTING_SEQUENCE_ID	Routing unique identifier.
PROCESS_REVISION	MTL_RTG_ITEM_REVISIONS	PROCESS_REVISION	Routing revision code.

Document Field	Oracle Applications Table/View Name	Column Name	Description
CHANGE_NOTICE	MTL_RTG_ITEM_REVISIONS	CHANGE_NOTICE	Engineering change order number.
ECN_INITIATION_DATE	MTL_RTG_ITEM_REVISIONS	ECN_INITIATION_DATE	Engineering change order initiation date.
IMPLEMENTATION_DATE	MTL_RTG_ITEM_REVISIONS	IMPLEMENTATION_DATE	Engineering change order implementation date.
EFFECTIVITY_DATE	MTL_RTG_ITEM_REVISIONS	EFFECTIVITY_DATE	Revision effectivity date.
INVENTORY_ITEM_NUMBER	MTL_SYSTEM_ITEMS_KFV	CONCATENATED_SEGMENTS	Concatenated item segments.
ORGANIZATION_NAME	HR_ALL_ORGANIZATIONS_UNITS	NAME	

1.2. BOM_OP_SEQUENCES

Document Field	Oracle Applications Table/View Name	Column Name	Description
OPERATION_SEQUENCE_ID	BOM_OPERATION_SEQUENCES	OPERATION_SEQUENCE_ID	Operation unique identifier.
ROUTING_SEQUENCE_ID	BOM_OPERATION_SEQUENCES	ROUTING_SEQUENCE_ID	Routing unique identifier.
OPERATION_SEQ_NUM	BOM_OPERATION_SEQUENCES	OPERATION_SEQ_NUM	Operation sequence number.
OPERATION_LEAD_TIME_PERCENT	BOM_OPERATION_SEQUENCES	OPERATION_LEAD_TIME_PERCENT	Indicates the amount of overlap its lead time has with the parent's lead time.
MINIMUM_TRANSFER_QUANTITY	BOM_OPERATION_SEQUENCES	MINIMUM_TRANSFER_QUANTITY	Minimum operation transfer quantity.
COUNT_POINT_TYPE	BOM_OPERATION_SEQUENCES	COUNT_POINT_TYPE	Use: Yes for Auto charge, No for Direct charge.

Document Field	Oracle Applications Table/View Name	Column Name	Description
OPERATION_DESCRIPTION	BOM_OPERATION_SEQUENCES	OPERATION_DESCRIPTION	
EFFECTIVITY_DATE	BOM_OPERATION_SEQUENCES	EFFECTIVITY_DATE	Date operation is effective.
DISABLE_DATE	BOM_OPERATION_SEQUENCES	DISABLE_DATE	Date operation is no longer effective; effectivity lasts until the end of the disable date.
BACKFLUSH_FLAG	BOM_OPERATION_SEQUENCES	BACKFLUSH_FLAG	Indicates whether operation requires backflushing. Use: 1 for Yes 2 for No.
OPTION_DEPENDENT_FLAG	BOM_OPERATION_SEQUENCES	OPTION_DEPENDENT_FLAG	Indicates whether to use this operation in all configuration routings, even if no components of the configuration are in this routing. Use: 1 for Yes 2 for No.
ALTERNATE_ROUTING_DESIGNATOR	BOM_OPERATION_SEQUENCES	ALTERNATE_ROUTING_DESIGNATOR	Alternate designator code.
ORGANIZATION_NAME	HR_ALL_ORGANIZATION_UNITS	NAME	Use HR_ALL_ORGANIZATION_UNITS to derive ORGANIZATION_ID.
ASSEMBLY_ITEM_NUMBER	MTL_SYSTEM_ITEMS_B_KFV	CONCATENATE_D_SEGMENTS	Concatenated item segments.
DEPARTMENT_CODE	BOM_DEPARTMENTS	DEPARTMENT_CODE	Required for Insert.
OPERATION_CODE	BOM_STANDARD_OPERATIONS	OPERATION_CODE	Standard operation code.

1.2.1. BOM_OP_RESOURCES

Document Field	Oracle Applications Table View Name	Column Name	Description
ROUTING_SEQUENCE_ID	BOM_ OPERATION_ SEQUENCES	ROUTING_ SEQUENCE_ID	
OPERATION_SEQUENCE_ID	BOM_ OPERATION_ RESOURCES	OPERATION_ SEQUENCE_ID	Operation sequence identifier.
RESOURCE_SEQ_NUM	BOM_ OPERATION_ RESOURCES	RESOURCE_ SEQ_NUM	Resource sequence number.
STANDARD_RATE_FLAG	BOM_ OPERATION_ RESOURCES	STANDARD_ RATE_FLAG	Indicates standard rate for shop floor transactions. Use: 1 for Yes 2 for No..
ASSIGNED_UNITS	BOM_ OPERATION_ RESOURCES	ASSIGNED_ UNITS	Resource units assigned.
USAGE_RATE_OR_AMOUNT	BOM_ OPERATION_ RESOURCES	USAGE_RATE_ OR_AMOUNT	Resource usage rate.
USAGE_RATE_OR_AMOUNT_INVERSE	BOM_ OPERATION_ RESOURCES	USAGE_ RATE_OR_ AMOUNT_ INVERSE	Resource usage rate inverse.
BASIS_TYPE	BOM_ OPERATION_RE SOURCES	BASIS_TYPE	Use: 1 for Item, 2 for Lot, 3 for Resource Unit, 4 for Resource Value, 5 for Total Value, 6 for Activity.
SCHEDULE_FLAG	BOM_ OPERATION_ RESOURCES	SCHEDULE_ FLAG	Indicates whether to schedule the resource. Use: 1 for Yes, 2 for No, 3 for Prior, 4 for Next.

Document Field	Oracle Applications Table View Name	Column Name	Description
RESOURCE_OFFSET_PERCENT	BOM_ OPERATION_ RESOURCES	RESOURCE_ OFFSET_ PERCENT	Resource offset percent from the start of the routing.
AUTOCHARGE_TYPE	BOM_ OPERATION_ RESOURCES	AUTOCHARGE_ _TYPE	Autocharge type for shopfloor moves. Use: 1 for WIP move, 2 for Manual, 3 for PO receipt, 4 for PO move.
ALTERNATE_ROUTING_ DESIGNATOR	BOM_ OPERATIONAL_ ROUTINGS	ALTERNATE_ ROUTING_ DESIGNATOR	Alternate designator code.
OPERATION_SEQ_NUM	BOM_ OPERATION_ SEQUENCES	OPERATION_ SEQ_NUM	Operation sequence number.
EFFECTIVITY_DATE	BOM_ OPERATION_ SEQUENCES	EFFECTIVITY_ DATE	Effective date.
ORGANIZATION_NAME	HR_ALL_ ORGANIZATION_ UNITS	NAME	Use HR_ALL_ORGANIZATION_ UNITS to derive ORGANIZATION_ID.
ASSEMBLY_ITEM_NUMBER	MTL_SYSTEM_ ITEMS_B_KFV	CONCATENATE D_SEGMENTS	Concatenated item segments.
RESOURCE_CODE	BOM_ RESOURCES	RESOURCE_ CODE	Resource name.
ACTIVITY	CST_ ACTIVITIES	ACTIVITY	Activity name.

Query Engineering BOM Service

The name of this service is:

WmOAMFG107SC.engineering107SC.queryOA.engineeringBOM:queryEngineeringBOM

This service retrieves information on the parent, component, and substitutes Engineering Bills of Material items as well as revisions. It retrieves this information based on the following parameters:

- ORGANIZATION_NAME: Organization Name.
- ITEM_NUMBER: Item Number.
- UNIT_OF_MEASURE: Unit of Measure.
- EFFECTIVITY_DATE_FROM: Effective Date.
- EFFECTIVITY_DATE_TO: Effective Date.

Database Scripts

This service uses the same database scripts as the Send Engineering BOM service.



Note: If you use this service but you do *not* use the Send Engineering BOM service, you should run the `wm_disable_from_engbom.sql` script to disable the triggers installed by the Send Engineering BOM service.

For a detailed description of these database scripts, see [“Send Engineering BOM Service” on page 325](#). For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service uses the following transaction definition:

- queryEngineeringBOMTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

You can query Engineering Bills of Material data using the following services:

- `specifyDefaultSettings` specifies the default parameter settings required for service execution. You should change these settings accordingly.
- `queryEngineeringBOMTxn` queries the Oracle Applications database for any Engineering BOM information matching the parameter values. The parameters are defined as the input to this service.

For more details on query transactions, see [“Using Query Services” on page 36](#).

Business Document Structure

The Query Engineering BOM service uses the same business document structure as the Send Engineering BOM service. For a detailed description of the business document’s structure, see [“Send Engineering BOM Service” on page 325](#).

Query Item Service

The name of this service is:

`WmOAMFG107SC.inventory107SC.queryOA.item:queryItem`

This service queries item information based on the following parameters:

- `ORGANIZATION_NAME`: Name of the organization.
- `ITEM_NUMBER`: Item number.
- `INACTIVE_DATE_FROM`: Beginning inactive date.
- `INACTIVE_DATE_TO`: Ending inactive date.
- `REVISION`: Number of an item.

Database Scripts

This service uses the same database scripts as the Send Item service.



Note: If you use this service but you do *not* use the Send Item service, you should run the `wm_disable_from_invitem.sql` script to disable the triggers installed by the Send Item service.

For a detailed description of these database scripts, see [“Send Item Service” on page 336](#). For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service uses the following transaction definition:

- queryItemTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow queryItem executes as follows:

- specifyDefaultSettings specifies the default parameter settings required for service execution. You should change these settings accordingly.
- queryItemTxn queries the Oracle Applications database for any Item information matching the parameter values. The parameters are defined as the input to this service.

For more details on query transactions, see [“Using Query Services” on page 36](#).

Business Document Structure

This service uses the same business document structure as the Send Item service. For a detailed description of the business document’s structure, see [“Send Item Service” on page 336](#).

Query Item Category Service

The name of this service is:

WmOAMFG107SC.inventory107SC.queryOA.itemCategory:queryItemCategory

This service queries item category information based on the following parameters:

- CATEGORY_NAME: Category name.
- STRUCTURE_NAME: Category set name.
- INACTIVE_DATE_FROM: Beginning inactive date.
- INACTIVE_DATE_TO: Ending inactive date.

Database Scripts

This service uses the following database scripts:

Script	Description
wm_install_from_itemcategory.sql	Runs the scripts listed below, except the uninstall script.
wm_from_itemcategory_vw.sql	Creates the following required view component: <ul style="list-style-type: none">■ WM_INV_ITEM_CATEGORY_QRY_VW
wm_drop_from_itemcategory.sql	Uninstalls all components created by wm_install_from_itemcategory.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service uses the following transaction definition:

- queryItemCategoryTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow queryItemCategory executes as follows. If no input parameters are specified, the query returns all rows.

- specifyDefaultSettings specifies the default parameter settings required for service execution. You should change these settings accordingly.
- queryItemCategoryTxn queries the Oracle Applications database for any Item Category records matching the parameter values. The parameters are defined as the input to this service. This service takes four inputs: CATEGORY_NAME, STRUCTURE_NAME, INACTIVE_DATE_FROM, and INACTIVE_DATE_TO.

For more details on query transactions, see [“Using Query Services” on page 36](#).

Business Document Structure

This service uses the following business document structure:

■ ITEM_CATEGORY

ITEM_CATEGORY

Document Field	Oracle Applications Table/View Name	Column Name	Description
WEB_TRANSACTION_ID		Not used	Not used for query outbound transactions.
DOCUMENT_TYPE			Use ITEMCATEGORY.
DOCUMENT_STATUS			Use QUERY.
CATEGORY	MTL_CATEGORIES_V	CATEGORY_CONCAT_SEGS	Segmented category name.
STRUCTURE	MTL_CATEGORIES_V	STRUCTURE_NAME	
INACTIVE_ON_DATE	MTL_CATEGORIES_V	DISABLE_DATE	
SUMMARY_FLAG	MTL_CATEGORIES_V	SUMMARY_FLAG	
ENABLED_FLAG	MTL_CATEGORIES_V	ENABLED_FLAG	
CATEGORY_DESCRIPTION	MTL_CATEGORIES_V	DESCRIPTION	

Query Manufacturing BOM Service

The name of this service is:

WmOAMFG107SC.billsOfMaterial107SC.queryOA.manufacturingBOM:queryManufacturingBOM

This service retrieves information about parent, component, and substitute Manufacturing Bills of Material items, as well as revisions. It retrieves this information based on the following parameters:

- ORGANIZATION_NAME: Organization name.
- ITEM_NUMBER: Item number.
- REVISION: Revision of the parent item.

- EFFECTIVITY_DATE_FROM: Beginning effective date.
- EFFECTIVITY_DATE_TO: Ending effective date.

Database Scripts

This service uses the same database scripts as the Send Manufacturing BOM service.



Note: If you use this service but you do *not* use the Send Manufacturing BOM service, you should run the `wm_disable_from_mfgbom.sql` script to disable the triggers installed by the Send Manufacturing BOM service.

For a detailed description of these database scripts, see [“Send Manufacturing BOM Service” on page 359](#). For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service uses the following transaction definition:

- `queryManufacturingBOMTxn107SC.txp`

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow `queryManufacturingBOM` executes as follows:

- `specifyDefaultSettings` specifies the default parameter settings required for service execution. You should change these settings accordingly.
- `queryManufacturingBOMTxn` queries the Oracle Applications database for any Manufacturing BOM information matching the parameter values. The parameters are defined as the input to this service.

For more details on query transactions, see [“Using Query Services” on page 36](#).

Business Document Structure

This service uses the same business document structure as the Send Manufacturing BOM service. For a detailed description of the business document’s structure, see [“Send Manufacturing BOM Service” on page 359](#).

Query On-Hand Quantity Service

The name of this service is:

WmOAMFG107SC.inventory107SC.intoOA.onHandQuantity:queryOnHandQuantity

This service retrieves the on-hand quantities for a particular item based on the following parameters:

- ITEM_NUMBER: Part number of the item as defined in Oracle Applications.
- REVISION: Revision number of the item.
- ORGANIZATION_NAME: Name of the inventory organization.
- SUBINVENTORY_CODE: Subinventory code.
- LOCATOR: Locator code.

Database Scripts

This service uses the following database scripts:

Script	Description
wm_install_from_onhandquantity.sql	Runs the scripts listed below, except the uninstall script.
wm_from_onhandquantity_vw.sql	Creates the following required view component: <ul style="list-style-type: none"> ■ WM_INV_ONHAND_QRY_VW
wm_drop_from_onhandquantity.sql	Uninstalls all components created by wm_install_from_onhandquantity.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service uses the following transaction definition:

- queryOnHandQuantityTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow queryOnHandQuantity executes as follows. If no input parameters are specified, the query returns all rows.

- specifyDefaultSettings specifies the default parameter settings required for service execution. You should change these settings accordingly.
- queryOnHandQuantityTxn queries the Oracle Applications database for any Inventory-On Hand records matching the parameter values. The parameters are defined as the input to this service.

For more details on query transactions, see [“Using Query Services” on page 36](#).

Business Document Structure

This service uses the following business document structure:

- ON_HAND_QUANTITY

ON_HAND_QUANTITY

Document Field	Oracle Applications Table/View Name	Column Name	Description
WEB_TRANSACTION_ID		Not used	Not used for query outbound transactions.
DOCUMENT_TYPE			INVONHAND will be populated in this field.
DOCUMENT_STATUS			QUERY will be populated in this field.
ORGANIZATION_ID	MTL_ONHAND_QUANTITIES	ORGANIZATION_ID	
INVENTORY_ITEM_ID	MTL_ONHAND_QUANTITIES	INVENTORY_ITEM_ID	
ITEM_NUMBER	MTL_SYSTEM_ITEMS_KFV	CONCATENATED_SEGMENTS	Concatenated segments.
REVISION	MTL_ONHAND_QUANTITIES	REVISION	
TOTAL_QOH	MTL_ONHAND_QUANTITIES	PRIMARY_TRANSACTION_QUANTITY	Transaction quantity.
SUBINVENTORY_CODE	MTL_ONHAND_QUANTITIES	SUBINVENTORY_CODE	

Document Field	Oracle Applications Table/View Name	Column Name	Description
LOCATOR	MTL_ITEM_LOCATIONS_KFV	CONCATENATED_SEGMENTS	Concatenated locator.
ITEM_DESCRIPTION	MTL_SYSTEM_ITEMS_KFV	DESCRIPTION	
PRIMARY_UOM_CODE	MTL_SYSTEM_ITEMS_KFV	PRIMARY_UOM_CODE	Primary unit of measure code.
ORGANIZATION_CODE	MTL_PARAMETERS	ORGANIZATION_CODE	
ORGANIZATION_NAME	HR_ORGANIZATION_UNITS	NAME	
LOCATOR_TYPE	MTL_SECONDARY_INVENTORIES	LOCATOR_TYPE	
ITEM_REV_CONTROL	MTL_SYSTEM_ITEMS_KFV	REVISION_QTY_CONTROL_CODE	
ITEM_LOCATOR_CONTROL	MTL_SYSTEM_ITEMS_KFV	LOCATION_CONTROL_CODE	
ITEM_LOT_CONTROL	MTL_SYSTEM_ITEMS_KFV	LOT_CONTROL_CODE	
ITEM_SERIAL_CONTROL	MTL_SYSTEM_ITEMS_KFV	SERIAL_NUMBER_CONTROL_CODE	

Receive Bills and Route Service

The name of this service is:
WmOAMFG107SC.billsOfMaterial107SC.intoOA.billsAndRoute:receiveBillsAndRoute

This service creates, updates, and deletes manufacturing and Engineering Bills of Material and product families. Based on the attributes of the parent item, the program creates planning, model, option class, and standard Bills of Material.

Database Scripts

This service uses the following database scripts:

Script	Description
wm_install_into_bom.sql	Runs the scripts listed below, except the uninstall script.
wm_into_bom_pkg.sql	Installs WM_BOM_IMP_HANDLER_PKG.WM_HANDLE_BOM, which calls the WM_CONC_REQUEST.WM_REQUEST_SUBMIT procedure to submit the Bills and Routing Import process.
wm_drop_from_bom.sql	Uninstalls all components created by wm_install_into_bom.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definitions:

- setBillsAndRouteTxn107SC.txp
- BillsAndRouteTransactions107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

This service calls the following services to import Bills of Materials:

- **specifyDefaultSettings** specifies the default parameter settings required for service execution. You should change these settings accordingly.
- **specifyConcProgParams** specifies the default parameter settings required for concurrent program execution. You should change these settings accordingly.
- **bizDocMapping** service maps the incoming business document structure to the required Oracle Applications data structures (the interface tables).
 - **getOrgId** service is used as one of the transformers while mapping the business doc Idata structure to the interface table Idata structure. It takes ORGANIZATION_NAME as the input parameter and queries the table ORG_ORGANIZATION_DEFINITIONS and gets the ORGANIZATION_ID corresponding to the ORGANIZATION_NAME.
 - **commonLookup** service is used as one of the transformers while mapping the business doc Idata structure to the line interface Idata structure. This is used to find the look up code for the particular look up type.
 - **convertToDateObject** service converts the text date format into object date format. This is used as a transformer in the bizDocMapping.
 - **getRevisedItemSeqId** service is used as one of the transformers while mapping the business doc Idata structure to the interface table Idata structure. It takes REVISED_ITEM_NUM, REVISION and ORG_NAME as input parameter to get the corresponding Revised Item Sequence Id (that is, REVISED_ITEM_SEQUENCE_ID).
 - **bomItemTypeLookup** service is used as one of the transformers while mapping the business doc Idata structure to the line interface Idata structure. This is used to find the look up code for the particular look up type BOM_ITEM_TYPE.
 - **getComponentSeqId** service is used as one of the transformers while mapping the business doc Idata structure to the interface table Idata structure. It takes ASSEMBLY_ITEM_NUMBER, ITEM_COMPONENT_NUMBER, OPERATION_SEQ_NUM, EFFECTIVE_DATE, ORG_NAME and ALT_BOM_DSG as input parameter to get the corresponding Component Sequence Id (that is, COMPONENT_SEQUENCE_ID).
 - **getCommonRtgSeqId** service is used as one of the transformers while mapping the business doc Idata structure to the interface table Idata structure. It takes COMMON_ITEM_NUMBER, COMMON_ORG_NAME AND ALT_RTG_DSG as input parameter to get the corresponding Common Routing Sequence Id (that is, ROUTING_SEQUENCE_ID).

- **setBillsAndRouteTxn** service inserts data into the Bills and Route interface tables. It takes data from the `Idata` structure that results from the `bizDocMapping` service, and puts the data into the interface tables in Oracle Applications for Bills and Route Interface.
- **importBillsAndRoute** imports data to the production table from the interface table. To monitor the import process, it sequentially invokes the services `execBillsAndRouteConcProg`, `checkBillsAndRouteImportStatus`, and `getBillsAndRouteImport_ERR`. If the status of execution is `FAILED`, this step maps the `dbErrorMsg` and `concProgMsg` record list. Otherwise, it checks for data errors that occurred during the import, and maps them to `errorsDoc`.
- **execBillsAndRouteConcProg** service inserts data into the production table. It picks up data from the Interface tables corresponding to Bills And Route Interface and inserts data into the production tables.
- **checkBillsAndRouteImportStatus** service queries the `MTL_INTERFACE_ERRORS` table to find the number of erroneous rows for the passed `REQUEST ID`.
- **getBillsAndRouteImport_ERR** service gets the error message that occurs during the data import (to the production table from interface table).

For more details on receive transactions, see [“Overview of Receive Service Transaction Processing” on page 31](#).

Business Document Structure

This services uses the business document structure `BOMBizDoc`. Its structure is as follows:

- 1.0. `BOM_BILL_OF_MTLS`
 - 1.1. `MTL_ITEM_REVISIONS`
 - 1.1.1. `BOM_INVENTORY_COMPS`
 - 1.1.1.1. `BOM_SUB_COMPS`
 - 1.1.1.2. `BOM_REF_DESGS`
- 2.0. `BOM_OP_ROUTINGS`
 - 2.1. `MTL_RTG_ITEM_REVS`
 - 2.2. `BOM_OP_SEQUENCES`
 - 2.2.1. `BOM_OP_RESOURCES`

1.0. BOM_BILL_OF_MTLS (Maps to BOM_BILL_OF_MTLS_INTERFACE)

Field Name	Maps to Column	Description
BILL_SEQUENCE_ID		Not used for Bills of Material inbound transaction.
ITEM_NUMBER	ITEM_NUMBER	Required. Concatenated segments.
ORGANIZATION_NAME	ORGANIZATION_ID	Required. Use HR_ALL_ORGANIZATION_UNITS to derive ORGANIZATION_ID.
COMMON_ITEM_NUMBER	COMMON_ITEM_NUMBER	Required if referencing a common Bill of Materials.
COMMON_ORG_NAME	COMMON_ORGANIZATION_ID	Required if referencing a common Bill of Materials. Use HR_ALL_ORGANIZATION_UNITS to derive COMMON_ORGANIZATION_ID.
COMMON_ALT_BOM_DESIGNATOR	COMMON_ALT_BOM_DESIGNATOR	Required if creating an alternate Bill of Materials.
ALTERNATE_BOM_DESIGNATOR	ALTERNATE_BOM_DESIGNATOR	Required if creating an alternate Bill of Materials.
SPECIFIC_ASSEMBLY_COMMENT	SPECIFIC_ASSEMBLY_COMMENT	Specific Bill of Materials comment.
PENDING_FROM_ECN	PENDING_FROM_ECN	Change notice that created this Bill of Materials.
ASSEMBLY_TYPE	ASSEMBLY_TYPE	Required for Create. Use: 1 for Manufacturing, 2 for Engineering.
DEMAND_SOURCE_LINE	DEMAND_SOURCE_LINE	For ATO.
SET_ID	SET_ID	Set identifier for ATO.
DEMAND_SOURCE_TYPE	DEMAND_SOURCE_TYPE	For ATO.
DEMAND_SOURCE_HEADER_ID	DEMAND_SOURCE_HEADER_ID	Demand source header identifier.
NEXT_EXPLODE_DATE	NEXT_EXPLODE_DATE	Next date when pre-explosion will be refreshed.

Field Name	Maps to Column	Description
REVISION	REVISION	Bill revision.
UOM		Not used for Bills of Material inbound transactions.

1.1. MTL_ITEM_REVISIONS (Maps to MTL_ITEM_REVISIONS_INTERFACE)

Field Name	Maps to Column	Description
BILL_SEQUENCE_ID		Not used for Bills of Material inbound transactions.
ITEM_NUMBER	ITEM_NUMBER	Required. Concatenated segments.
ORGANIZATION_NAME	ORGANIZATION_ID	Required. Use HR_ALL_ORGANIZATION_UNITS to derive ORGANIZATION_ID.
REVISION	REVISION	Required. Item revision code.
CHANGE_NOTICE	CHANGE_NOTICE	Engineering change order number.
ECN_INITIATION_DATE	ECN_INITIATION_DATE	ECO Initiation date.
IMPLEMENTATION_DATE	IMPLEMENTATION_DATE	ECO implementation date.
EFFECTIVITY_DATE	EFFECTIVITY_DATE	Revision effective date.
REVISED_ITEM_SEQUENCE_ID	REVISED_ITEM_SEQUENCE_ID	Revised Item sequence ID.
TRANSACTION_TYPE	TRANSACTION_TYPE	Required. Use Create, Update, or Delete.

1.1.1. BOM_INVENTORY_COMPONENTS (Maps to BOM_INVENTORY_COMPS_INTERFACE)

Field Name	Maps to Column	Description
BILL_SEQUENCE_ID		Not used for Bills of Material inbound transactions.
ASSEMBLY_ITEM_NUMBER	ASSEMBLY_ITEM_NUMBER	Required. Inventory item number of manufactured assembly.
ALTERNATE_BOM_DESIGNATOR	ALTERNATE_BOM_DESIGNATOR	Required if alternate Bill of Materials.

Field Name	Maps to Column	Description
ORGANIZATION_NAME	ORGANIZATION_ID	Required. Use HR_ALL_ORGANIZATION_UNITS to derive ORGANIZATION_ID.
COMPONENT_ITEM_NUMBER	COMPONENT_ITEM_NUMBER	Required. Component item number.
OPERATION_SEQ_NUM	OPERATION_SEQ_NUM	Required. Operation sequence number.
EFFECTIVITY_DATE	EFFECTIVITY_DATE	Required. Effective date.
LOCATION_NAME	LOCATION_NAME	Supply locator name.
SUBSTITUTE_COMP_NUMBER	SUBSTITUTE_COMP_NUMBER	Concatenated item segments of the substitute component.
ITEM_NUM	ITEM_NUM	Item sequence within Bill of Materials structure.
COMPONENT_QUANTITY	COMPONENT_QUANTITY	Required. Quantity of component in Bill of Materials.
COMPONENT_YIELD_FACTOR	COMPONENT_YIELD_FACTOR	Required. Factor used to multiply component quantity with to obtain required component quantity.
COMPONENT_REMARKS	COMPONENT_REMARKS	Component remarks.
CHANGE_NOTICE	CHANGE_NOTICE	Engineering change order number.
IMPLEMENTATION_DATE	IMPLEMENTATION_DATE	Date on which engineering change order was implemented.
DISABLE_DATE	DISABLE_DATE	Disable date.
PLANNING_FACTOR	PLANNING_FACTOR	Required. Factor used to multiply component quantity with to obtain planning quantity.
QUANTITY_RELATED	QUANTITY_RELATED	Required. Indicates whether this component has quantity related reference designators. Use: 1 for Yes 2 for No.

Field Name	Maps to Column	Description
SO_BASIS	SO_BASIS	Required. Quantity basis used by Oracle Order Management to determine how many units of component to put on an order. Use: 1 for Option Class, 2 for None.
OPTIONAL	OPTIONAL	Indicates whether component is optional in bill. Use: 1 for Yes 2 for No.
MUTUALLY_EXCLUSIVE_OPTIONS	MUTUALLY_EXCLUSIVE_OPTIONS	Indicates whether one or more children of the component can be picked when taking an order. Use: 1 for Yes 2 for No..
INCLUDE_IN_COST_ROLLUP	INCLUDE_IN_COST_ROLLUP	Required. Indicates whether the component should be used when rolling up costs. Use: 1 for Yes 2 for No.
CHECK_ATP	CHECK_ATP	Required. Indicates whether ATP check is required. Use: 1 for Yes 2 for No.
SHIPPING_ALLOWED	SHIPPING_ALLOWED	Indicates whether component is allowed to ship. Use: 1 for Yes 2 for No.
REQUIRED_TO_SHIP	REQUIRED_TO_SHIP	Indicates whether component is required to ship. Use: 1 for Yes 2 for No.
REQUIRED_FOR_REVENUE	REQUIRED_FOR_REVENUE	Indicates whether component is required for revenue. Use: 1 for Yes 2 for No.

Field Name	Maps to Column	Description
INCLUDE_ON_SHIP_DOCS	INCLUDE_ON_SHIP_DOCS	Indicates whether component is displayed on shipping documents. Use: 1 for Yes 2 for No.
INCLUDE_ON_BILL_DOCS	INCLUDE_ON_BILL_DOCS	Indicates whether component is displayed on billing documents. Use: 1 for Yes 2 for No.
LOW_QUANTITY	LOW_QUANTITY	Minimum quantity allowed on an order.
HIGH_QUANTITY	HIGH_QUANTITY	Maximum quantity allowed on an order.
ACD_TYPE	ACD_TYPE	Add, change, or disable code for component on an engineering change order. Use: 1 for Add, 2 for Change, 3 for Disable.
REVISED_ITEM_NUMBER	REVISED_ITEM_SEQUENCE_ID	Revised Item Number, Organization Name and version is used to get REVISED_ITEM_SEQUENCE_ID.
REVISION		
WIP_SUPPLY_TYPE	WIP_SUPPLY_TYPE	WIP supply type code. Use: 1 for Push 2 for Assembly Pull 3 for Operation Pull 4 for Bulk 5 for Supplier 6 for Phantom 7 for Based on Bill.
SUPPLY_SUBINVENTORY	SUPPLY_SUBINVENTORY	Supply subinventory.
REFERENCE_DESIGNATOR	REFERENCE_DESIGNATOR	Component reference designator.

Field Name	Maps to Column	Description
BOM_ITEM_TYPE	BOM_ITEM_TYPE	Type of item (de-normalized from BOM Item Type in MTL_SYSTEM_ITEMS). Use: 1 for Model, 2 for Option class, 3 for Planning, 4 for Standard, 5 for Product Family.
OPERATION_LEAD_TIME_PERCENT	OPERATION_LEAD_TIME_PERCENT	Operation offset from first operation in routing (de-normalized from corresponding column in BOM_OPERATION_SEQUENCES)
UOM		Not used for Bills of Material inbound transactions.

1.1.1.1. BOM_SUB_COMPS (Maps to BOM_SUB_COMPS_INTERFACE)

Field Name	Maps to Column	Description
BILL_SEQUENCE_ID		Not used for Bills of Material inbound transactions.
SUBSTITUTE_COMPONENT_NUMBER	SUBSTITUTE_COMPONENT_NUMBER	Required.
SUBSTITUTE_ITEM_QUANTITY	SUBSTITUTE_ITEM_QUANTITY	Required. Substitute component quantity.
ACD_TYPE	ACD_TYPE	Type to indicate add or delete on an engineering change order.
CHANGE_NOTICE	CHANGE_NOTICE	Engineering change order number.
ORGANIZATION_NAME	ORGANIZATION_ID	Required. Use HR_ALL_ORGANIZATION_UNITS to derive ORGANIZATION_ID.

Field Name	Maps to Column	Description
ASSEMBLY_ITEM_NUMBER	COMPONENT_SEQUENCE_ID	Required. Inventory item number of manufactured assembly, alternate Bill of Materials designator code, inventory item number of component. Manufacturing operation sequence number, effective date, and ORGANIZATION_NAME are used to derive COMPONENT_SEQUENCE_ID.
ALTERNATE_BOM_DESIGNATOR		
COMPONENT_ITEM_NUMBER		
OPERATION_SEQ_NUM		
EFFECTIVITY_DATE		
UOM		Not used for Bills of Material inbound transactions.

1.1.1.2. BOM_REF_DESGS (Maps to BOM_REF_DESGS_INTERFACE)

Field Name	Maps to Column	Description
BILL_SEQUENCE_ID		Not used for Bills of Material inbound transactions.
COMPONENT_REFERENCE_DESIGNATOR	COMPONENT_REFERENCE_DESIGNATOR	Required. Component reference designator.
ORGANIZATION_NAME	ORGANIZATION_ID	Required. Use HR_ALL_ORGANIZATION_UNITS to derive ORGANIZATION_ID.
ASSEMBLY_ITEM_NUMBER	COMPONENT_SEQUENCE_ID	Required. Assembly item number, alternate Bill of Materials identifier, manufacturing operation sequence number, date on which functionality will be enabled, and ORGANIZATION_CODE are used to derive COMPONENT_SEQUENCE_ID.

Field Name	Maps to Column	Description
ALTERNATE_BOM_DESIGNATOR		
COMPONENT_ITEM_NUMBER		
OPERATION_SEQ_NUM		
EFFECTIVITY_DATE		
REF_DESIGNATOR_COMMENT	REF_DESIGNATOR_COMMENT	Reference designator comment.
CHANGE_NOTICE	CHANGE_NOTICE	Engineering change order number.
ACD_TYPE	ACD_TYPE	Set the action for an engineering change order by using: 1 for Add 2 for Change 3 for Disable.

2.0. BOM_OP_ROUTINGS (Maps to BOM_OP_ROUTINGS _INTERFACE)

Field Name	Maps to Column	Description
ROUTING_SEQUENCE_ID	ROUTING_SEQUENCE_ID	
ORGANIZATION_NAME	ORGANIZATION_ID	Required. Use HR_ALL_ORGANIZATION_UNITS to derive ORGANIZATION_ID.
ASSEMBLY_ITEM_NUMBER	ASSEMBLY_ITEM_NUMBER	Required. Concatenated item segments.
ALTERNATE_ROUTING_DESIGNATOR	ALTERNATE_ROUTING_DESIGNATOR	Required to identify an alternate Bill of Materials.
ROUTING_TYPE	ROUTING_TYPE	Required. Use: 1 for Manufacturing 2 for Engineering.
COMMON_ITEM_NUMBER	COMMON_ITEM_NUMBER	Concatenated item segments of the common routing.
COMMON_ORG_NAME	COMMON_ROUTING_SEQUENCE_ID	Derived from columns COMMON_ITEM_NUMBER, COMMON_ORG_NAME, and ALTERNATE_ROUTING_DESIGNATOR.
ROUTING_COMMENT	ROUTING_COMMENT	

Field Name	Maps to Column	Description
COMPLETION_SUBINVENTORY	COMPLETION_SUBINVENTORY	Destination subinventory for assembly.
DEMAND_SOURCE_LINE	DEMAND_SOURCE_LINE	Demand source line used for ATO.
SET_ID	SET_ID	Set identifier used for ATO
PROCESS_REVISION	PROCESS_REVISION	Routing revision
DEMAND_SOURCE_TYPE	DEMAND_SOURCE_TYPE	Demand source type used for ATO
DEMAND_SOURCE_HEADER_ID	DEMAND_SOURCE_HEADER_ID	Demand source header identifier used for ATO.
LOCATION_NAME	LOCATION_NAME	Concatenated location segments.

2.1. MTL_RTG_ITEM_REVS (Maps to MTL_RTG_ITEM_REVS_INTERFACE)

Field Name	Maps to Column	Description
ROUTING_SEQUENCE_ID		Not used in inbound transactions.
PROCESS_REVISION	PROCESS_REVISION	Required. Routing revision code.
CHANGE_NOTICE	CHANGE_NOTICE	Engineering change order number.
ECN_INITIATION_DATE	ECN_INITIATION_DATE	ECO initiation date.
IMPLEMENTATION_DATE	IMPLEMENTATION_DATE	ECO implementation date.
EFFECTIVITY_DATE	EFFECTIVITY_DATE	Revision effective date.
INVENTORY_ITEM_NUMBER	INVENTORY_ITEM_NUMBER	Required. Concatenated item segments.
ORGANIZATION_NAME	ORGANIZATION_ID	Required. Use HR_ALL_ORGANIZATION_UNITS to derive ORGANIZATION_ID.
TRANSACTION_TYPE	TRANSACTION_TYPE	Required. Use Insert, Update, or Delete.

2.2. BOM_OP_SEQUENCES (Maps to BOM_OP_SEQUENCES_INTERFACE)

Field Name	Maps to Column	Description
ROUTING_SEQUENCE_ID		Not used in inbound transactions.
OPERATION_SEQ_NUM	OPERATION_SEQ_NUM	Required. Operation sequence number.
OPERATION_LEAD_TIME_PERCENT	OPERATION_LEAD_TIME_PERCENT	Indicates the amount of overlap its lead time has with the parent's lead time.
MINIMUM_TRANSFER_QUANTITY	MINIMUM_TRANSFER_QUANTITY	Minimum operation transfer quantity.
COUNT_POINT_TYPE	COUNT_POINT_TYPE	Use: 1 for Yes auto charge, 2 for No auto charge, 3 for No direct charge.
OPERATION_DESCRIPTION	OPERATION_DESCRIPTION	Description of operation.
EFFECTIVITY_DATE	EFFECTIVITY_DATE	Required. Date operation is effective.
DISABLE_DATE	DISABLE_DATE	Date operation is no longer effective. Effective operation lasts until the end of the disable date.
BACKFLUSH_FLAG	BACKFLUSH_FLAG	Indicates whether operation requires backflushing. Use: 1 for Yes 2 for No.
OPTION_DEPENDENT_FLAG	OPTION_DEPENDENT_FLAG	Indicates whether to use this operation in all configuration routings, even if no components of the configuration are used in this routing. Use: 1 for Yes 2 for No.
ALTERNATE_ROUTING_DESIGNATOR	ALTERNATE_ROUTING_DESIGNATOR	Required. Alternate designator code.
ORGANIZATION_NAME	ORGANIZATION_ID	Required. Use HR_ALL_ORGANIZATION_UNITS to derive ORGANIZATION_ID.
ASSEMBLY_ITEM_NUMBER	ASSEMBLY_ITEM_NUMBER	Required. Concatenated item segments.

Field Name	Maps to Column	Description
DEPARTMENT_CODE	DEPARTMENT_CODE	Required. Department Name.
OPERATION_CODE	OPERATION_CODE	Standard operation code.
RESOURCE_CODE1	RESOURCE_CODE1	First resource name.
RESOURCE_CODE2	RESOURCE_CODE2	Second resource name.
RESOURCE_CODE3	RESOURCE_CODE3	Third resource name.
INSTRUCTION_CODE1	INSTRUCTION_CODE1	First standard instruction code.
INSTRUCTION_CODE2	INSTRUCTION_CODE2	Second standard instruction code.
INSTRUCTION_CODE3	INSTRUCTION_CODE3	Third standard instruction code.

2.2.1. BOM_OP_RESOURCES (Maps to BOM_OP_RESOURCES_INTERFACE)

Field Name	Maps to Column	Description
ROUTING_SEQUENCE_ID		Not used in inbound transactions.
RESOURCE_SEQ_NUM	RESOURCE_SEQ_NUM	Resource sequence number.
STANDARD_RATE_FLAG	STANDARD_RATE_FLAG	Use standard rate for shop floor transactions. Use: 1 for Yes 2 for No.
ASSIGNED_UNITS	ASSIGNED_UNITS	Resource units assigned.
USAGE_RATE_OR_AMOUNT	USAGE_RATE_OR_AMOUNT	Resource usage rate.
USAGE_RATE_OR_AMOUNT_INVERSE	USAGE_RATE_OR_AMOUNT_INVERSE	Resource usage rate inverse.
BASIS_TYPE	BASIS_TYPE	Basis type identifier. Use: 1 for Item, 2 for Lot, 3 for Resource Units, 4 for Resource Value, 5 for Total Value, 6 for Activity.
SCHEDULE_FLAG	SCHEDULE_FLAG	Schedule the resource. Use: 1 for Yes, 2 for No, 3 for Prior, 4 for Next.
RESOURCE_OFFSET_PERCENT	RESOURCE_OFFSET_PERCENT	Resource offset percent from the start of the routing.

Field Name	Maps to Column	Description
AUTOCHARGE_TYPE	AUTOCHARGE_TYPE	Auto charge type for shop floor moves. Use: 1 for WIP move 2 for Manual 3 for PO receipt 4 for PO move.
ALTERNATE_ROUTING_DESIGNATOR	ALTERNATE_ROUTING_DESIGNATOR	Required.
OPERATION_SEQ_NUM	OPERATION_SEQ_NUM	Required. Operation sequence number.
EFFECTIVITY_DATE	EFFECTIVITY_DATE	Required. Effective date.
ORGANIZATION_NAME	ORGANIZATION_ID	Required. Use HR_ALL_ORGANIZATION_UNITS to derive ORGANIZATION_ID.
ASSEMBLY_ITEM_NUMBER	ASSEMBLY_ITEM_NUMBER	Required. Concatenated item segments.
RESOURCE_CODE	RESOURCE_CODE	
ACTIVITY	ACTIVITY	

Receive Customer Item Service

The name of this service is:

WmOAMFG107SC.inventory107SC.intoOA.customerItem:receiveCustomerItem

This service imports Customer Items into Oracle Inventory.

You can import Customer Items at any of the following levels: Customer-level, Address Category-level, or Address-level. The level is specified in the Item Definition Level column. An item imported at the Customer Level is visible to all organizations. Items can also be imported with an inactive status.

While submitting the concurrent program Import Customer Items to load the Customer Item records from interface tables to production tables, you cannot restrict the processing only to the records uploaded by the IS flow instance. The program will process all the unprocessed records present in the interface table at the time of execution. After executing the concurrent program, errors will be retrieved for all processed records that might not be uploaded by this flow instance. We suggest that you keep the interface table free of error records, to minimize ambiguity.

Database Scripts

This service uses the following database scripts:

Script	Description
wm_install_into_custitem.sql	Runs the scripts listed below, except the uninstall script.
wm_into_custitem_pkg.sql	Installs WM_INV_CTITEM_IMP_HANDLER_PKG.WM_HANDLE_CUSTITEM, which calls the WM_CONC_REQUEST.WM_REQUEST_SUBMIT procedure to submit the Customer Item Import process. The process in the Oracle Inventory module is Import Customer Items.
wm_drop_into_custitem.sql	Uninstalls all components created by wm_install_into_custitem.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definitions:

- setCustomerItemTxn 107SC.txp
- CustomerItemTransactions107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

This service calls the following services to import inventory transactions:

- **specifyDefaultSettings** specifies the default parameter settings required for service execution. You should change these settings accordingly.
- **specifyConcProgParams** specifies the default parameter settings required for concurrent program execution. You should change these settings accordingly.
- **bizDocMapping** maps the incoming business document structure to the required Oracle Applications data structures (the interface tables).
 - **getOASystemDateObject** returns the Oracle Applications system date as a date object.

- **getOracleAppsUserId** is a transformer for mapping the business document IData structure to the interface table IData structure. It takes ORACLE_APPS_USER_NAME as the input parameter from the business document, and queries the table FND_USER to get the USER_ID. The USER_ID information is required for insertion into the interface tables.
- **getProgramAndAppId** returns the concurrent program ID and corresponding Application ID of the Customer Item Import Concurrent program.
- **setCustomerItemTxn** inserts data into the interface table. It extracts data from the IData structure that results from the bizDocMapping service, and puts the data into the interface table in Oracle Applications for Customer Items.
- **importCustomerItem** imports data to the production table from the interface table. It calls the services **execCustomerItemConcProg**, **checkCustomerItemImportStatus**, and **getCustomerItemImport_ERR** to execute the corresponding concurrent program that inserts data into the production table, and to generate the error/ acknowledgement message. If erroneous records are found, it indicates an error during import and calls **getCustomerItemImport_ERR** to retrieve the errors. If no error record is found, it checks if the status of the execution is FAILED (returned by the service **execCustomerItemConcProg**). If this is the case, it maps the program failure message to **dbErrorMessage** and it comes out of the flow. If the status is successful, it deletes unwanted variables, comes out of the flow, and indicates successful completion of the import process.
- **execCustomerItemConcProg** invokes the stored procedure WM_INV_CTITEM_IMP_HANDLER_PKG.WM_HANDLE_CUSTITEM. This procedure calls a corresponding concurrent subroutine to execute the data import process for Customer Items into Oracle Applications. This service gives Status Id, Request Id, Execution Status Message (in case of normal completion of concurrent program) and database Stored Procedure error message (in case an exception occurs in Stored Procedure execution).
- **checkCustomerItemImportStatus** checks the status of the execution by checking the Interface Table for any rejected record. If the query does not return any rows, it indicates successful import. If the query returns rows, it indicates that the concurrent program could not import the data successfully in the production tables of Oracle Applications.



Note: Because the Interface table is not updated with the REQUEST_ID, it is not possible to check errors corresponding to only the current bizDoc import. The error doc will show old errors as well.

- `getCustomerItemImport_ERR` gets the error message that occurs during the data import (to the production table from interface table). It scans the `MTL_CI_INTERFACE` table to get the rejected records.



Note: Because the Interface table is not updated with the `REQUEST_ID`, it is not possible to check errors corresponding to only the current bizDoc import. The error doc will show old errors as well.

For more details on receive transactions, see [“Overview of Receive Service Transaction Processing” on page 31](#).

Business Document Structure

This service uses the business document `customerItemBizDoc`. Its structure is as follows:

- `CUST_ITEM`

`CUST_ITEM` (Maps to `MTL_CI_INTERFACE`)

Field Name	Maps to Column	Description
<code>CUSTOMER_NAME</code>	<code>CUSTOMER_NAME</code>	
<code>CUSTOMER_CATEGORY_CODE</code>	<code>CUSTOMER_CATEGORY_CODE</code>	
<code>ADDRESS1</code>	<code>ADDRESS1</code>	
<code>ADDRESS2</code>	<code>ADDRESS2</code>	
<code>ADDRESS3</code>	<code>ADDRESS3</code>	
<code>ADDRESS4</code>	<code>ADDRESS4</code>	
<code>CITY</code>	<code>CITY</code>	
<code>STATE</code>	<code>STATE</code>	
<code>COUNTY</code>	<code>COUNTY</code>	
<code>COUNTRY</code>	<code>COUNTRY</code>	
<code>POSTAL_CODE</code>	<code>POSTAL_CODE</code>	
<code>CUSTOMER_ITEM_NUMBER</code>	<code>CUSTOMER_ITEM_NUMBER</code>	Required.

Field Name	Maps to Column	Description
ITEM_DEFINITION_LEVEL	ITEM_DEFINITION_LEVEL	Definition Level for the Customer Item. Use: 1 for Customer, 2 for Address Category, 3 for Address. If this is at the Address Category Level, then Customer and Address Category are required. If this is at the Address Level, then the complete address fields and the Customer field are required.
CUSTOMER_ITEM_DESC	CUSTOMER_ITEM_DESC	Description of the Customer Item defined above.
MODEL_CUSTOMER_ITEM_NUMBER	MODEL_CUSTOMER_ITEM_NUMBER	
COMMODITY_CODE	COMMODITY_CODE	Commodity code as defined in Oracle Inventory.
MASTER_CONTAINER	MASTER_CONTAINER	Multi-segmented values for a container item.
CONTAINER_ITEM_ORG_NAME	CONTAINER_ITEM_ORG_NAME	Organization name for the container item.
DETAIL_CONTAINER	DETAIL_CONTAINER	Detail container item name (flexfield concatenated segments).
MIN_FILL_PERCENTAGE	MIN_FILL_PERCENTAGE	Minimum fill percentage.
DEP_PLAN_REQUIRED_FLAG	DEP_PLAN_REQUIRED_FLAG	Flag to indicate whether departure planning is required for the Customer Item. Use: 1 for Yes or 2 for No.
DEP_PLAN_PRIOR_BLD_FLAG	DEP_PLAN_PRIOR_BLD_FLAG	Indicates whether departure plan before build is required. Use: 1 for Yes 2 for No.
INACTIVE_FLAG	INACTIVE_FLAG	Required. Use: 1 for Yes 2 for No.

Field Name	Maps to Column	Description
DEMAND_TOLERANCE_POSITIVE	DEMAND_TOLERANCE_POSITIVE	Positive tolerance (high end) for customer item demand.
DEMAND_TOLERANCE_NEGATIVE	DEMAND_TOLERANCE_NEGATIVE	Negative tolerance (low end) for customer item demand.

Receive Customer Item Cross Reference Service

The name of this service is:

WmOAMFG107SC.inventory107SC.intoOA.CICrossRef.receiveCICrossRef

This service imports Customer Item Cross-References into Oracle Inventory.

You can import Customer Item Cross References at any of the following levels: Customer-level, Address Category-level, or Address-level. The level is specified in the Item Definition Level column. An item imported at the Customer level is visible to all organizations. You can also import items with an inactive status. It is possible to define multiple references between a customer item and several inventory items. In this case, a preference ranking is established for processing in Oracle Inventory when a customer item is demanded.

While submitting the concurrent program Customer Item Cross References Import to load the Customer Item cross reference records from interface tables to production tables, it is not possible to restrict the processing to only the records uploaded by the IS flow instance. The program will process all unprocessed records that exist in the interface table at the time of execution. After executing the concurrent program, errors will be retrieved for all processed records that might not be uploaded by this flow instance. We suggest that you keep the interface table free of error records to minimize ambiguity.

Database Scripts

This service uses the following database scripts:

Script	Description
wm_install_into_cicrossref.sql	Runs the scripts listed below, except the uninstall script.

Script	Description
wm_install_into_cicrossref_pkg.sql	Installs WM_INV_CICREF_IMP_HANDLER_PKG.WM_HANDLE_CICREF, which calls the WM_CONC_REQUEST.WM_REQUEST_SUBMIT procedure to submit the Customer Item Import process. The process in Oracle Inventory module is Import Customer Item Cross References.
wm_drop_into_cicrossref.sql	Uninstalls the components created by wm_install_into_cicrossref.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definitions:

- setCICrossRefTxn 107SC.txp
- CICrossRefTransactions107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

This service calls the following services to import customer item cross references:

- specifyDefaultSettings specifies the default parameter settings required for service execution. You should change these settings accordingly.
- specifyConcProgParams specifies the default parameter settings required for concurrent program execution. You should change these settings accordingly.
- bizDocMapping maps the incoming business document structure to the required Oracle Applications data structures (the interface tables).
 - getOASystemDateObject returns the Oracle Applications system date as a date object.
 - getOracleAppsUserId is a transformer for mapping the business document IData structure to the interface table IData structure. It takes ORACLE_APPS_USER_NAME as the input parameter from the business document, and queries the table FND_USER to get the USER_ID. The USER_ID information is required for insertion into the interface tables.

- **getProgramAndApplId** returns the Concurrent Program ID and corresponding Application ID of the Customer Item Cross Reference Import Concurrent program.
- **setCICrossRefTxn** inserts data into the interface table. It extracts data from the Idata structure results from the bizDocMapping service, and puts the data into the interface table in Oracle Applications for CICrossRef.
- **importCICrossRef** imports data to the production table from the interface table. It calls the services **execCICrossRefConcProg**, **checkCICrossRefImportStatus** and **getCICrossRefImport_ERR** to execute the corresponding concurrent program that inserts data into the production table, and to generate the error/ acknowledgement message. It also checks for the records rejected. If rejected records are found, it indicates an error during import and calls **getCICrossRefImport_ERR** to retrieve the errors. If no rejected records are found, it checks if the status of the execution is FAILED (returned by the service **execCICrossRefConcProg**). If so, it maps the program failure message to **dbErrorMessage** and comes out of the flow. In case of success, it deletes unwanted variables, comes out of the flow, and indicates successful completion of the import process.
- **execCICrossRefConcProg** invokes the stored procedure **WM_INV_CICRREF_IMP_HANDLER_PKG.WM_HANDLE_CICRREF**. This procedure calls a corresponding concurrent subroutine to execute the data import process for Customer Item Cross Reference into Oracle Applications. This service gives Status Id, Request Id, Execution Status Message (in case of normal completion of concurrent program), and database Stored Procedure error message (in case an exception occurs in Stored Procedure execution).
- **checkCICrossRefImportStatus** checks the status of the execution by checking the Interface Table for rejected records. If the query does not return any rows, it indicates successful import. If the query returns any row, it indicates that the concurrent program could not import data successfully in the production tables of Oracle Applications.



Note: Because the Interface table is not updated with the REQUEST_ID, there is no way to check errors corresponding to only the current bizDoc import. The error doc will show old errors as well.

- **getCICrossRefImport_ERR** gets the error message that occurs during the data import (to the production table from interface table). It scans the **MTL_CI_XREFS_INTERFACE** table to get the rejected records.



Note: Because the Interface table is not updated with the REQUEST_ID, there is no way to check errors corresponding to only the current bizDoc import. The error doc will show old errors as well.

For more details on receive transactions, see [“Overview of Receive Service Transaction Processing” on page 31](#).

Business Document Structure

This service uses the business document CustomerItemCrossReferencesBizDoc. Its structure is as follows:

■ CUST_ITEM_CROSS_REF

CUST_ITEM_CROSS_REF (Maps to MTL_CI_XREFS_INTERFACE)

Field Name	Map to Column	Description
CUSTOMER_NAME	CUSTOMER_NAME	Required.
CUSTOMER_CATEGORY_CODE	CUSTOMER_CATEGORY_CODE	Address category code of the customer.
ADDRESS1	ADDRESS1	
ADDRESS2	ADDRESS2	
ADDRESS3	ADDRESS3	
ADDRESS4	ADDRESS4	
CITY	CITY	
STATE	STATE	
COUNTY	COUNTY	
COUNTRY	COUNTRY	
POSTAL_CODE	POSTAL_CODE	
CUSTOMER_ITEM_NUMBER	CUSTOMER_ITEM_NUMBER	Required. Item number of the customer.
ITEM_DEFINITION_LEVEL	ITEM_DEFINITION_LEVEL	Item definition level for import. Use: 1 for Customer, 2 for Address Category, 3 for Address. If you use 2, Customer and Address Category fields are required. If you use 3, Customer and Address fields are required.
INVENTORY_ITEM	INVENTORY_ITEM	Item number as defined in Oracle Inventory for the item to be used for cross reference.
MASTER_ORGANIZATION_NAME	MASTER_ORGANIZATION_NAME	Name of the master inventory organization.

Field Name	Map to Column	Description
PREFERENCE_NUMBER	PREFERENCE_NUMBER	Required. Preference ranking for multiple references to the same item.
INACTIVE_FLAG	INACTIVE_FLAG	Required. Use: 1 for Yes or 2 for No.

Receive Inventory Transaction Service

The name of this service is:

WmOAMFG107SC.inventory107SC.intoOA.INVTransaction:receiveInvTransaction

This service loads transactions from external applications and feeder systems into Oracle Inventory. These transactions can include:

- Inventory issues and receipts (including user-defined transaction types)
- Subinventory transfers
- Direct inter-organization transfers
- Intransit shipments
- WIP component issues and returns
- Sales orders shipments
- Inventory average cost updates

Material transactions are processed by the Inventory Transaction Manager, which spawns transaction workers depending on the load. Normally, this is run asynchronously. For this transaction, one transaction worker will be spawned to process the records in the interface table. Because the Inventory Transaction Manager runs asynchronously, it will not pick up these records.

Database Scripts

This service uses the following database scripts:

Script	Description
wm_install_into_invtxn.sql	Runs the scripts listed below, except the uninstall script.
wm_into_invtxn_pkg.sql	Installs WM_INV_TXN_IMP_HANDLER_PKG.WM_HANDLER_INV_TXN, which calls the WM_CONC_REQUEST.WM_REQUEST_SUBMIT procedure to submit the Inventory Open Transaction Import process. The process in the Inventory Module is Material Transaction.
wm_into_invtxn_seq.sql	Creates the following components: <ul style="list-style-type: none"> ■ WM_MTL_TRANSACTION_HEADER_S, which creates the TRANSACTION_HEADER_ID sequence ■ WM_SERIAL_TRANSACTION_TEMP_S, which create the SERIAL_TRANSACTION_TEMP_ID sequence ■ WM_MATERIAL_TRANSACTION_S, which creates the TRANSACTION_INTERFACE_ID sequence
wm_drop_into_invtxn.sql	Uninstalls all components created by wm_install_into_invtxn.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definitions:

- setInvTransactionTxn107SC.txp
- InvTransactionTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

This service calls the following services to import inventory transactions:

- **specifyDefaultSettings** specifies the default parameter settings required for service execution. You should change these settings accordingly.
- **specifyConcProgParams** specifies the default parameter settings required for concurrent program execution. You should change these settings accordingly.
- **bizDocMapping** service maps the incoming business document structure to the required Oracle Applications data structures (the interface tables).
 - **getInventoryItemId** is used as one of the transformers while mapping the business doc Idata structure to the interface table Idata structure. It takes ITEM_CODE , ORGANIZATION_NAME as the input parameter and queries the table MTL_SYSTEM_ITEMS_B_KFV and HR_ALL_ORGANIZATIONS_UNITS to get the INVENTORY_ITEM_ID corresponding to the input parameters.
 - **getLocatorId** is used as one of the transformers while mapping the business doc Idata structure to the interface table Idata structure. It takes LOCATOR and ORGANIZATION_NAME as the input parameter and queries the view MTL_ITEM_LOCATIONS_KFV and the table ORG_ORGANIZATION_DEFINITIONS and gets the INVENTORY_LOCATION_ID corresponding to the LOCATOR and ORGANIZATION_NAME.
 - **getSourceId** is used as one of the transformers while mapping the business doc Idata structure to the interface table Idata structure. Transaction source id can be generated from one of the input parameters. This service takes all the parameters and calls appropriate service to result source_id.
 - **getDispositionId** is used as one of the transformers while mapping the business doc Idata structure to the interface table Idata structure. It takes ACCOUNT_ALIAS as the input parameter and queries the table MTL_GENERIC_DISPOSITIONS and gets the DISPOSITION_ID corresponding to the alias.
 - **getWIPEntityId** is used as one of the transformers while mapping the business doc that takes the WIP_ENTITY_NAME and ORGANIZATION_NAME as inputs and results WIP_ENTITY_ID by querying WIP_ENTITIES table.
 - **getSalesOrderHeaderId** is used as one of the transformers while mapping the business doc Idata structure to the interface table Idata structure. It takes SALES_ORDER as the input parameter and queries the table OE_ORDER_HEADERS_ALL and gets the HEADER_ID corresponding to the SALES_ORDER.

- **getTransactionTypeId** is used as one of the transformers while mapping the business doc Idata structure to the interface table Idata structure. It takes TRANSACTION_TYPE_NAME as the input parameter and queries the table MTL_TRANSACTION_TYPES and gets the TRANSACTION_TYPE_ID, TRANSACTION_ACTION_ID, TRANSACTION_SOURCE_TYPE_ID corresponding to the TRANSACTION_TYPE_NAME.
- **getReasonId** is used as one of the transformers while mapping the business doc Idata structure to the interface table Idata structure. It takes REASON_NAME as the input parameter and queries the table MTL_TRANSACTION_REASONS and gets the REASON_ID corresponding to the REASON_NAME.
- **getRequisitionLineId** is used as one of the transformers while mapping the business doc Idata structure to the interface table Idata structure. It takes PO_REQUISITION_LINE_NUMBER as the input parameter and queries the table PO_REQUISITION_HEADERS_ALL, PO_REQUISITION_LINES_ALL and gets the REQUISITION_LINE_ID corresponding to the PO_REQUISITION_LINE_NUMBER.
- **getEntityTypeValue** is used as one of the transformers that converts the entity type text to a corresponding value.
- **getSerialTempId** is used as one of the transformers while mapping the business doc Idata structure to the interface table Idata structure. It takes a record List containing records of serial controlled inventory. If the item is both lot and serial controlled then only it picks the sequence number.
- **setInvTransactionTx** inserts data into the interface table. It extracts data from the Idata structure that results from the bizDocMapping service, and puts the data into the interface table in Oracle Applications for Inventory Transaction.
- **importInvTransaction** imports data to the production table from the interface table. It calls services execInvTransactionConcProg, checkInvTransactionImportStatus, and getInvTransactionImport_ERR to execute the corresponding concurrent program that inserts data into the production table, and to generate the error/ acknowledgement message. If the status of the execution is SUCCESS (returned by the service execInvTransactionConcProg), it checks for the record in the interface table. If a record is found, it then indicates an error during import. In this case, the service calls getInvTransactionImport_ERR to retrieve the errors. If no record is found, it comes out of the flow, indicating success of the data import process. If the status of the execution is FAILED, it comes out of execution.
- **execInvTransactionConcProg** invokes the stored procedure WM_INV_TXN_IMP_HANDLER_PKG.WM_HANDLE_INV_TXN. This procedure calls a corresponding concurrent subroutine to execute the data import process for Inventory Open Transactions into Oracle Applications. This service gives Status Id, Request Id, Execution Status Message (in case of normal completion of concurrent program) and database error message (in case an exception occurs in Stored Procedure execution).

- `checkInvTransactionImportStatus` checks the status of the execution by checking the Interface Table for any rejected record corresponding to the current TRANSACTION_HEADER_ID. If the query does not return any rows, it indicates successful import. If the query returns any row, it indicates that the concurrent program could not import data successfully in the production tables of Oracle Applications.
- `getInvTransactionImport_ERR` gets the error message that occurs during the data import (to the production table from interface table). Based on the parameter TRANSACTION_HEADER_ID, it scans the MTL_TRANSACTIONS table to get the corresponding message.

For more details on receive transactions, see [“Overview of Receive Service Transaction Processing” on page 31](#).

Business Document Structure

This service uses the business document InvTransactionBizDoc. Its structure is as follows:

- 1.0. MTL_TRANSACTIONS
 - 1.1. MTL_SERIAL_NUMBERS
 - 1.2. MTL_TRANSACTION_LOTS
 - 1.2.1. MTL_SERIAL_NUMBERS

Use the appropriate business document structures for items as follows:

For items that are ...	Use this document structure ...
Serial-controlled	MTL_TRANSACTIONS and MTL_SERIAL_NUMBERS
Lot-controlled	MTL_TRANSACTIONS and MTL_TRANSACTION_LOTS
Serial-controlled and lot-controlled	MTL_TRANSACTIONS, MTL_SERIAL_NUMBERS, and MTL_TRANSACTION_LOTS
Neither serial number-controlled nor lot-controlled	MTL_TRANSACTIONS

The total number of records in the Material Serial Numbers/Material Lot Numbers must match the corresponding header transaction quantity for serial-controlled items. The total number of records in Material Serial Numbers must match the Material Lot Numbers transaction quantity if the item is both serial-controlled and lot-controlled.

1.0. MTL_TRANSACTIONS (Maps to MTL_TRANSACTIONS_INTERFACE)

Field Name	Maps to Column	Description
SOURCE_CODE	SOURCE_CODE	Required. Identifies the external system.
SOURCE_LINE_ID	SOURCE_LINE_ID	Required. External System Reference.
SOURCE_HEADER_ID	SOURCE_HEADER_ID	Required. External System Reference.
INVENTORY_ITEM	INVENTORY_ITEM_ID	Derives the Inventory Item ID from MTK_SYSTEM_ITEMS_B_KFV table.
REVISION	REVISION	Revision number if under revision control.
ORGANIZATION_NAME	ORGANIZATION_ID	Required. Derived from ORG_ORGANIZATION_DEFINITIONS table.
SET_OF_BOOKS_NAME		Required.
SUBINVENTORY_CODE	SUBINVENTORY_CODE	Subinventory code for the item; needed for all transaction types except cost update.
LOCATOR	LOCATOR_ID	Derived from MTK_ITEM_LOCATIONS_KFV table; to be provided if item is under locator control.
TRANSACTION_QUANTITY	TRANSACTION_QUANTITY	Required.
TRANSACTION_UOM	TRANSACTION_UOM	Required. Unit of Measure.
TRANSACTION_DATE	TRANSACTION_DATE	Required.
TRANSACTION_SOURCE_NAME	TRANSACTION_SOURCE_NAME	User-defined source name.
TRANSACTION_TYPE_NAME	TRANSACTION_TYPE_ID TRANSACTION_SOURCE_ID	Derived from MTL_TRANSACTION_TYPES table.
ACCOUNT		Required only in case of an Account Type of Transaction. COMBINATION_ID is derived using GL_CODE_COBINATIONS_KFV.
ACCOUNT_ALIAS		Required only for Account Alias Transactions. DISPOSITION_ID is derived using MTL_GENERIC_DISPOSITIONS table.
JOB		Required only for WIP and Schedule transactions. WIP_ENTITY_ID is derived using WIP_ENTITIES table.

Field Name	Maps to Column	Description
SALES_ORDER		Required only for Sales Order Transactions. Key flexfield segments provided. HEADER_ID is derived from the OE_ORDER_HEADERS_ALL table.
REASON	REASON_ID	Derived from MTL_TRANSACTION_REASONS.
PO_REQUISITION_NUMBER	REQUISITION_LINE_ID	Derived from the PO_REQUISITION_HEADERS_ALL and PO_REQUISITION_LINES_ALL tables.
PO_REQUISITION_LINE_NUMBER		
TRANSACTION_REFERENCE	TRANSACTION_REFERENCE	
TRANSACTION_COST	TRANSACTION_COST	Cost to be used for Inventory Issues. If left blank, the cost in the system is used.
DISTRIBUTION_ACCOUNT_CODE	DISTRIBUTION_ACCOUNT_ID	Required in case of Inventory Issues/ receipts of an asset item and for sales order shipment transactions.
CURRENCY_CODE	CURRENCY_CODE	Required if transaction cost is in a different currency than the SOB currency.
CURRENCY_CONVERSION_TYPE	CURRENCY_CONVERSION_TYPE	
CURRENCY_CONVERSION_RATE	CURRENCY_CONVERSION_RATE	
CURRENCY_CONVERSION_DATE	CURRENCY_CONVERSION_DATE	
USSGL_TRANSACTION_CODE	USSGL_TRANSACTION_CODE	United States general ledger code.
ENCUMBRANCE_ACCOUNT	ENCUMBRANCE_ACCOUNT	Derived from GL_CODE_COMBINATIONS_KFV.
ENCUMBRANCE_AMOUNT	ENCUMBRANCE_AMOUNT	
VENDOR_LOT_NUMBER	VENDOR_LOT_NUMBER	Cross reference supplier lot number.
TRANSFER_SUBINVENTORY	TRANSFER_SUBINVENTORY	Required for Subinventory transfers.

Field Name	Maps to Column	Description
TRANSFER_ORGANIZATION_NAME	TRANSFER_ORGANIZATION	Derive the ORGANIZATION_ID using the organization name.
TRANSFER_LOCATOR_SEGMENTED	TRANSFER_LOCATOR	Destination Locator Internal Id derived from MTL_ITEM_LOCATIONS_KFV.
SHIPMENT_NUMBER	SHIPMENT_NUMBER	Shipment number. Required for Intransit shipments.
TRANSPORTATION_COST	TRANSPORTATION_COST	
TRANSPORTATION_ACCOUNT_CODE	TRANSPORTATION_ACCOUNT	Derives CODE_COMBINATION_ID from GL_CODE_COMBINATIONS_KFV.
TRANSFER_COST	TRANSFER_COST	
FREIGHT_CODE	FREIGHT_CODE	
CONTAINERS	CONTAINERS	Number of containers.
WAYBILL_NUMBER	WAYBILL_NUMBER	
EXPECTED_ARRIVAL_DATE	EXPECTED_ARRIVAL_DATE	Expected Arrival Date (no timestamp).
NEW_AVERAGE_COST	NEW_AVERAGE_COST	Required for Average Cost Transactions.
VALUE_CHANGE	VALUE_CHANGE	Required for Average Cost Transactions.
PERCENTAGE_CHANGE	PERCENTAGE_CHANGE	Required for Average Cost Transactions.
WIP_ENTITY_TYPE	WIP_ENTITY_TYPE	Required for WIP component issues and returns: 1 for Standard discrete jobs 2 for Repetitive Schedules 3 for Non-standard discrete jobs 4 for Work Order less schedule.
OPERATION_SEQ_NUM	OPERATION_SEQ_NUM	Required for WIP component issues and returns with routings. For WIP routings, values should be 1.
SHIPPABLE_FLAG	SHIPPABLE_FLAG	

1.1. MTL_SERIAL_NUMBERS (Maps to MTL_SERIAL_NUMBERS_INTERFACE)

Field Name	Maps to Column	Description
SOURCE_CODE	SOURCE_CODE	Required. User-defined source identifier.
SOURCE_LINE_ID	SOURCE_LINE_ID	External line identifier.
VENDOR_SERIAL_NUMBER	VENDOR_SERIAL_NUMBER	Used to reference vendor information.
FM_SERIAL_NUMBER	FM_SERIAL_NUMBER	Required. The starting serial number in the range. If only the FM_SERIAL_NUMBER is entered, the transaction processor assumes that only one transaction is being transacted.
TO_SERIAL_NUMBER	TO_SERIAL_NUMBER	Required. Use to specify a range.

1.2. MTL_TRANSACTION_LOTS (MTL_TRANSACTION_LOTS_INTERFACE)

Field Name	Maps to Column	Description
SOURCE_CODE	SOURCE_CODE	User-defined source identifier.
SOURCE_LINE_ID	SOURCE_LINE_ID	User-defined line ID to identify external system.
LOT_NUMBER	LOT_NUMBER	Required. Lot number being transacted.
LOT_EXPIRATION_DATE	LOT_EXPIRATION_DATE	Required if the item is under lot expiration control.
TRANSACTION_QUANTITY	TRANSACTION_QUANTITY	Required. Quantity being transacted.
VENDOR_SERIAL_NUMBER	VENDOR_SERIAL_NUMBER	Used to reference vendor information.

1.2.1. MTL_SERIAL_NUMBERS (MTL_SERIAL_NUMBERS_INTERFACE)

Field Name	Maps to Column	Description
SOURCE_CODE	SOURCE_CODE	Required. User-defined source identifier.
SOURCE_LINE_ID	SOURCE_LINE_ID	External line identifier.

Field Name	Maps to Column	Description
VENDOR_SERIAL_NUMBER	VENDOR_SERIAL_NUMBER	Used to reference vendor information.
FM_SERIAL_NUMBER	FM_SERIAL_NUMBER	Required. Starting serial number in the range. If only the FM_SERIAL_NUMBER is entered, Transaction processor assumes that only one transaction is being transacted.
TO_SERIAL_NUMBER	TO_SERIAL_NUMBER	Required. Use to specify a range.

Receive Item Service

The name of this service is:

WmOAMFG107SC.inventory107SC.intoOA.item:receiveItem

This service imports items from external sources into Oracle Applications. You can also use this service to import item revisions simultaneously. Item Category Import is not supported.

Features of this service include the following:

- Items can be assigned to a master organization or a child organization. When you assign an item to a child organization, all item-level attributes default down from the master organization only if the attribute column is null. The only exceptions are attributes under status control.
- Item import for both a master organization and a child organization in a single run is not supported. Thus the flow is designed to run twice, in case records exist for both master and child organizations.
- Revisions always exist at the item organization-level. You need revision data for each item organization you update. If you choose not to use revisions table, do not include revisions in the bizDoc. The item interface import assigns each item a beginning revision, using the default specified in the organization parameters.

Database Scripts

This service uses the following database scripts:

Script	Description
wm_install_into_invitem.sql	Runs the scripts listed below, except the uninstall script.
wm_into_invitem_pkg.sql	Installs WM_INV_ITEM_IMP_HANDLER_PKG.WM_HANDLE_INV_ITEM, which calls the WM_CONC_REQUEST.WM_REQUEST_SUBMIT procedure to submit the Item Import process.
wm_drop_into_invitem.sql	Uninstalls all components created by wm_install_into_invitem.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definitions:

- setItemTxn107SC.txp
- ItemTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

This service calls the following services to import items:

- **specifyDefaultSettings** specifies the default parameter settings required for service execution. You should change these settings accordingly.
- **specifyConcProgParams** specifies the default parameter settings required for concurrent program execution for Item. You should change these settings accordingly.
- **bizDocMapping** maps the incoming business document structure to the required Oracle Applications data structures (the interface tables).
 - **getCoverageScheduleId** is used as one of the transformers that gets COVERAGE_SCHEDULE_ID querying the Oracle Applications database using COVERAGE_SCHEDULE_NAME.

- **getFACategoryId** is used as one of the transformers while mapping the business doc that takes the **ASSET_CATEGORY** as inputs and results **CATEGORY_ID**.
- **getFNDLookUpCode** converts the incoming Look up type to **LOOKUP_CODE** required by the Oracle Applications database.
- **getHazardClassId** is used as one of the transformers while mapping the business doc **Idata** structure to the interface table **Idata** structure. It takes **HAZARD_CLASS** as the input parameter, and gets the **HAZARD_CLASS_ID** querying Oracle Applications database.
- **getItemGroupId** is used as one of the transformers while mapping the business doc **Idata** structure to the interface table **Idata** structure. It takes **DESCRIPTION** as input to get the corresponding **ITEM_CATALOG_GROUP_ID** querying Oracle Applications database.
- **getLookUpCode** is used as one of the transformers while mapping the business doc **Idata** structure to the interface table **Idata** structure. It takes **LOOKUP_TYPE** and **MEANING** as the input parameters, and queries the table **MFG_LOOKUPS** to get the **LOOKUP_CODE** corresponding to the **MEANING** and **TYPE**.
- **getPOVendorId** is used as one of the transformers while mapping the business doc **Idata** structure to the interface table **Idata** structure. It takes **VENDOR_NAME** as the input parameter, and queries the table **PO_VENDORS** to get the **VENDOR_ID** corresponding to the **VENDOR_NAME**.
- **getPaymentTermId** is used as one of the transformers while mapping the business doc **Idata** structure to the interface table **Idata** structure. It takes **PAYMENT_TERM_NAME** as the input parameter, and queries the table **RA_TERMS** to get the **TERM_ID** corresponding to the **PAYMENT_TERM_NAME**.
- **getRoutingHeaderId** is used as one of the transformers while mapping the business doc **Idata** structure to the interface table **Idata** structure. It takes **ROUTING_NAME** as the input parameter, and queries the table **RCV_ROUTING_HEADERS** to get the **ROUTING_HEADER_ID** corresponding to the **ROUTING_NAME**.
- **getRuleId** is used as one of the transformers while mapping the business doc **Idata** structure to the interface table **Idata** structure. It takes **RULE_NAME** as the input parameter, and queries the table **RA_RULES** to get the **RULE_ID** corresponding to the input parameters.
- **getATPRuleId** is used as one of the transformers while mapping the business doc **Idata** structure to the interface table **Idata** structure. It takes **RULE_NAME** as the input parameter, and queries the table **MTL_ATP_RULES** to get the **RULE_ID** corresponding to the input parameters.

- `getServiceableItemId` is used as one of the transformers while mapping the business doc Idata structure to the interface table Idata structure. It takes `SERVICEABLE_ITEM_CLASS_NAME` as the input parameter, and queries the table `CS_SERVICEABLE_ITEM_CLASSES` to get the `SERVICEABLE_ITEM_CLASS_ID` corresponding to the `SERVICEABLE_ITEM_CLASS_NAME`.
- `getTransactionSourceId` is used as one of the transformers while mapping the business doc Idata structure to the interface table Idata structure. It takes `TRANSACTION_SOURCE_TYPE_NAME` as the input parameter, and queries the table `MTL_TXN_SOURCE_TYPES` to get the `TRANSACTION_SOURCE_TYPE_ID` corresponding to the name.
- `getProcessId` is used as one of the transformers while mapping the business doc Idata structure to the interface table Idata structure. There can be two types of transaction data master organization or child organization. A child should have a parent data already inserted. Based on the data type that is identified by organization name, it returns the associated process ID. Based on the data category, it sets a flag to indicate a parent or child data transaction. Similarly, this same transformer is used for the update operation.
- `getOrgId` takes `ORGANIZATION_NAME` as the input parameter, and queries the table `ORG_ORGANIZATION_DEFINITIONS` to get the `ORGANIZATION_ID` corresponding to the `ORGANIZATION_NAME`.
- `getOracleAppsUserId` is used as one of the transformers while mapping the business doc Idata structure to the interface table Idata structure. It takes `ORACLE_APPS_USER_NAME` as the input parameter, and queries the table `FND_USER` to get the `USER_ID`. This User ID information must be inserted into the interface tables.
- `getOASystemDateObject` returns the Oracle Applications system date as a date object.
- `getInventoryItemId` is used as one of the transformers while mapping the business doc Idata structure to the interface table Idata structure. It takes `ITEM_CODE` as the input parameter, and queries the table `MTL_SYSTEM_ITEMS_KFV` to get the `INVENTORY_ITEM_ID` corresponding to the current `ORGANIZATION_NAME`.
- `getCodeCombinationId` is used as one of the transformers while mapping the business doc Idata structure to the interface table Idata structure. It takes `CONCATENATED_SEGMENTS` as the input parameter, and queries the table `GL_CODE_COMBINATIONS_KFV` to get the `CODE_COMBINATION_ID` corresponding to the passed code.
- `convertToDateObject` gets the date string as input, and converts it to a date object.
- `pickSequence` gets the sequence name as an input, and picks up the next sequence for that particular one.

- **getLocationId** is used as one of the transformers while mapping the business doc Idata structure to the interface table Idata structure. It takes LOCATOR and ORGANIZATION_NAME as the input parameter, and queries the view MTL_ITEM_LOCATIONS_KFV and the table ORG_ORGANIZATION_DEFINITIONS to get the INVENTORY_LOCATION_ID corresponding to the LOCATOR and ORGANIZATION_NAME.
- **importItem** imports data to the production table from the interface table. It calls the services execItemConcProg, checkItemImportStatus, and getItemImport_ERR to execute the corresponding concurrent program that inserts data into the production table, and to generate the error/ acknowledgement message.

If the status of the execution is SUCCESS (returned by the service execItemConcProg), it checks for the record in the interface table. If records are found, it then indicates an error during import. In this case, it calls getItemImport_ERR to retrieve the errors. If no records are found, it comes out of the flow, indicating success of the data import process.

If the status of the execution is FAILED, it comes out of execution.



Note: Three kinds of data records can be imported: parent, child, and update operation. This flow checks for the each kind of data and calls the import programs accordingly (parent first, then child, and then update).

- **execItemConcProg** invokes the stored procedure WM_INV_ITEM_IMP_HANDLER_PKG.WM_HANDLE_INV_ITEM. This procedure calls a corresponding concurrent subroutine to execute the data import process for Item into Oracle Application. This service gives Status Id, Request Id, Execution Status Message (in case of normal completion of concurrent program), and database Stored Procedure error message (in case an exception occurs in Stored Procedure execution).
- **checkItemImportStatus** checks the status of the execution by checking the interface table for any rejected record corresponding to the current REQUEST_ID. If the query does not return any rows, it indicates successful import. If the query returns rows, it indicates that the concurrent program could not import data successfully in the production tables of Oracle Applications.
- **getItemImport_ERR** gets the error message that occurs during the data import (to the production table from interface table). Based on the parameter SET_PROCESS_ID, it scans tables MTL_INTERFACE_ERRORS to get the corresponding message

For more details on receive transactions, see [“Overview of Receive Service Transaction Processing” on page 31](#).

Business Document Structure

This services uses the business document structure itemBizDoc. Its structure is as follows:

- 1.0. MTL_SYSTEM_ITEMS
 - 1.1. MTL_ITEM_REVISIONS

1.0. MTL_SYSTEM_ITEMS (Maps to MTL_SYSTEM_ITEMS_INTERFACE)

Field Name	Maps to Column	Description
ITEM_NUMBER	INVENTORY_ITEM_ID	Segmented Item Number.
ORGANIZATION_NAME	ORGANIZATION_ID	Derives ORGANIZATION_ID from ORG_ORGANIZATION_DEFINITIONS.
SUMMARY_FLAG	SUMMARY_FLAG	Required. Flexfield summary flag.
TAX_CODE	TAX_CODE	Tax Code for the Item.
ENABLED_FLAG	ENABLED_FLAG	
START_DATE_ACTIVE	START_DATE_ACTIVE	Flexfield segment start date.
END_DATE_ACTIVE	END_DATE_ACTIVE	Flexfield segment end date.
DESCRIPTION	DESCRIPTION	Item description.
BUYER_FULL_NAME	BUYER_ID	Derives AGENT_ID from PO_AGENTS, PER_ALL_PEOPLE_F, where PO_AGENTS.AGENT_ID = PER_ALL_PEOPLE_F.PERSON_ID.
ACCOUNTING_RULE_NAME	ACCOUNTING_RULE_ID	Derives the RULE_ID from RA_RULES.
INVOICING_RULE_NAME	INVOICING_RULE_ID	Derives the RULE_ID from RA_RULES.
SEGMENT1 to SEGMENT20	SEGMENT1 to SEGMENT20	Segmented Item Number; use the ITEM_NUMBER above or the Segmented Item Number columns to enter Part No data.
PURCHASING_ITEM_FLAG	PURCHASING_ITEM_FLAG	Purchasable Item (Y/N).
SHIPPABLE_ITEM_FLAG	SHIPPABLE_ITEM_FLAG	Indicates whether the item can be shipped (Y/N).
CUSTOMER_ORDER_FLAG	CUSTOMER_ORDER_FLAG	Customer Order Enabled flag (Y/N).

Field Name	Maps to Column	Description
INTERNAL_ORDER_FLAG	INTERNAL_ORDER_FLAG	Internal Order flag (Y/N)
SERVICE_ITEM_FLAG	SERVICE_ITEM_FLAG	Serviceable item (Y/N).
INVENTORY_ITEM_FLAG	INVENTORY_ITEM_FLAG	Indicates Inventory item (Y/N).
ENG_ITEM_FLAG	END_ITEM_FLAG	Indicates Engineering Item (Y/N).
INVENTORY_ASSET_FLAG	INVENTORY_ASSET_FLAG	Indicates Inventory asset (Y/N).
PURCHASING_ENABLED_FLAG	PURCHASING_ENABLED_FLAG	Indicates whether the item is purchaseable (Y/N).
CUSTOMER_ORDER_ENABLED_FLAG	CUSTOMER_ORDER_ENABLED_FLAG	Indicates whether the item is customer orderable (Y/N).
INTERNAL_ORDER_ENABLED_FLAG	INTERNAL_ORDER_ENABLED_FLAG	Indicates whether the item is internally orderable (Y/N)
SO_TRANSACTIONS_FLAG	SO_TRANSACTIONS_FLAG	Sales Order Transaction flag (Y/N).
MTL_TRANSACTIONS_ENABLED_FLAG	MTL_TRANSACTIONS_ENABLED_FLAG	Indicates whether the item is transactable (Y/N).
STOCK_ENABLED_FLAG	STOCK_ENABLED_FLAG	Indicates whether the item is stockable (Y/N).
BOM_ENABLED_FLAG	BOM_ENABLED_FLAG	Indicates whether the item can appear on a Bill of Materials (Y/N).
BUILD_IN_WIP_FLAG	BUILD_IN_WIP_FLAG	Indicates whether the item can be built in WIP (Y/N).
REVISION_QTY_CONTROL	REVISION_QTY_CONTROL_CODE	Derives LOOKUP_CODE from MFG_LOOKUPS, where LOOKUP_TYPE is MTL_ENG_QUANTITY for meaning.
CATALOG_GROUP_DESCRIPTION	ITEM_CATALOG_GROUP_ID	Derives ITEM_CATALOG_GROUP_ID from MTL_ITEM_CATALOG_GROUP S.
CATALOG_STATUS_FLAG	CATALOG_STATUS_FLAG	Indicates whether the item catalog is complete (Y/N).
RETURNABLE_FLAG	RETURNABLE_FLAG	Indicates whether the item can be returned (Y/N).

Field Name	Maps to Column	Description
DEFAULT_SHIP_ORGANIZATION_NAME	DEFAULT_SHIPPING_ORG	Derives ORGANIZATION_ID from ORG_ORGANIZATION_DEFINITIONS.
COLLATERAL_FLAG	COLLATERAL_FLAG	Indicates whether the item is a collateral item (Y/N).
TAXABLE_FLAG	TAXABLE_FLAG	Indicates whether the item is taxable (Y/N).
QTY_RCV_EXCEPTION_CODE	QTY_RCV_EXCEPTION_CODE	Use: NONE for no receiving control enforced, REJECT for preventing receipt of goods or services, or WARNING for displaying warning message.
ALLOW_ITEM_DESC_UPDATE_FLAG	ALLOW_ITEM_DESC_UPDATE_FLAG	Allow Item description updates on PO lines (Y/N)
INSPECTION_REQUIRED_FLAG	INSPECTION_FLAG	Indicates whether inspection is required (Y/N).
RECEIPT_REQUIRED_FLAG	RECEIPT_FLAG	Indicates supplier receipt is required (Y/N).
MARKET_PRICE	MARKET_PRICE	Purchasing Market Price.
HAZARD_CLASS	HAZARD_CLASS_ID	Derives the HAZARD_CLASS_ID from PO_HAZARD_CLASSES.
RFQ_REQUIRED_FLAG	RFQ_REQUIRED_FLAG	Indicates whether RFQ is required (Y/N).
QTY_RCV_TOLERANCE	QTY_RCV_TOLERANCE	Maximum quantity permissible over receipt percentage.
LIST_PRICE_PER_UNIT	LIST_PRICE_PER_UNIT	
UN_NUMBER_ID	UN_NUMBER_ID	Purchasing UN number.
PRICE_TOLERANCE_PERCENT	PRICE_TOLERANCE_PERCENT	Price tolerance percentage.
ASSET_CATEGORY	ASSET_CATEGORY_ID	Derives the ASSET_CATEGORY_ID from FA_CATEGORIES_KFV for the concatenated segment.
ROUNDING_FACTOR	ROUNDING_FACTOR	Used to determine order quantity.
UNIT_OF_ISSUE	UNIT_OF_ISSUE	

Field Name	Maps to Column	Description
ENFORCE_SHIP_TO_LOCATION_CODE	ENFORCE_SHIP_TO_LOCATION_CODE	Requires receipt location to match ship to location.
ALLOW_SUBSTITUTE_RECEIPTS_FLAG	ALLOW_SUBSTITUTE_RECEIPTS_FLAG	Indicates whether substitute receipts are allowed (Y/N).
ALLOW_UNORDERED_RECEIPTS_FLAG	ALLOW_UNORDERED_RECEIPTS_FLAG	Indicates whether to allow unordered receipts (Y/N).
ALLOW_EXPRESS_DELIVERY_FLAG	ALLOW_EXPRESS_DELIVERY_FLAG	Indicates whether to allow express delivery (Y/N).
DAYS_EARLY_RECEIPT_ALLOWED	DAYS_EARLY_RECEIPT_ALLOWED	Days early receipt.
DAYS_LATE_RECEIPT_ALLOWED	DAYS_LATE_RECEIPT_ALLOWED	Days late receipt.
RECEIPT_DAYS_EXCEPTION_CODE	RECEIPT_DAYS_EXCEPTION_CODE	Use: NONE for no receiving control enforced, REJECT for preventing receipt of goods or services, WARNING for displaying warning message.
RECEIVING_ROUTING_NAME	RECEIVING_ROUTING_ID	Derives ROUTING_HEADER_ID from RCV_ROUTING_HEADERS.
INVOICE_CLOSE_TOLERANCE	INVOICE_CLOSE_TOLERANCE	
RECEIVE_CLOSE_TOLERANCE	RECEIVE_CLOSE_TOLERANCE	
AUTO_LOT_ALPHA_PREFIX	AUTO_LOT_ALPHA_PREFIX	Lot prefix for lot number controlled item.
START_AUTO_LOT_NUMBER	START_AUTO_LOT_NUMBER	
LOT_CONTROL	LOT_CONTROL_CODE	Derived from MFG_LOOKUPS, where LOOKUP_TYPE=MTL_LOT_CONTROL for the meaning column. Use: 1 for No lot control, 2 for Full lot control.

Field Name	Maps to Column	Description
SHELF_LIFE	SHELF_LIFE_CODE	Derived from MFG_LOOKUPS, where LOOKUP_TYPE=MTL_SHELF_LIFE for the meaning column. Use: 1 for No shelf life control, 2 for Item shelf life days, 4 for User-defined expiration date.
SHELF_LIFE_DAYS	SHELF_LIFE_DAYS	
SERIAL_NUMBER_CONTROL	SERIAL_NUMBER_CONTROL_CODE	Derived from MFG_LOOKUPS, where LOOKUP_TYPE = MTL_SERIAL_NUMBER for the meaning column. Use: 1 for No serial number control, 2 for Predefined serial numbers, 5 for Dynamic entry at inventory receipt, 6 for Dynamic entry at sales order issue.
START_AUTO_SERIAL_NUMBER	START_AUTO_SERIAL_NUMBER	
AUTO_SERIAL_ALPHA_PREFIX	AUTO_SERIAL_ALPHA_PREFIX	
SOURCE_TYPE	SOURCE_TYPE	Use Inventory or Supplier. Derived from MFG_LOOKUPS, where LOOKUP_TYPE = MTL_SOURCE_TYPES for the meaning column.
SOURCE_ORGANIZATION_NAME	SOURCE_ORGANIZATION_ID	Derives ORGANIZATION_ID from ORG_ORGANIZATION_DEFINITIONS.
SOURCE_SUBINVENTORY	SOURCE_SUBINVENTORY	
EXPENSE_ACCOUNT	EXPENSE_ACCOUNT	Derives CODE_COMBINATION_ID from GL_CODE_COMBINATIONS_KFV.

Field Name	Maps to Column	Description
ENCUMBRANCE_ACCOUNT	ENCUMBRANCE_ACCOUNT	Derives CODE_COMBINATION_ID from GL_CODE_COMBINATIONS_KFV.
RESTRICT_SUBINVENTORIES	RESTRICT_SUBINVENTORIES_CODE	Derived from MFG_LOOKUPS, where LOOKUP_TYPE = MTL_SUBINVENTORY_RESTRICTIONS for the meaning column. Use: 1 for Subinventories restricted to predefined list, 2 for Subinventories not restricted to predefined list.
UNIT_WEIGHT	UNIT_WEIGHT	
WEIGHT_UOM_CODE	WEIGHT_UOM_CODE	Weight unit of measure code.
VOLUME_UOM_CODE	VOLUME_UOM_CODE	Volume unit of measure code.
UNIT_VOLUME	UNIT_VOLUME	
RESTRICT_LOCATORS	RESTRICT_LOCATORS_CODE	Derived from MFG_LOOKUPS, where LOOKUP_TYPE = MTL_LOCATOR_RESTRICTIONS for the meaning column. Use: 1 for Locators restricted to predefined list, 2 for Locators not restricted to predefined list.
LOCATION_CONTROL	LOCATION_CONTROL_CODE	Derived from MFG_LOOKUPS, where LOOKUP_TYPE = MTL_LOCATION_CONTROL for the meaning column. Use: 1 for No locator control, 2 for Previously specified locator control, 3 for Dynamic entry locator control, 4 for Locator control determined at subinventory level, 5 for Locator control determined at item level.
SHINKRAGE_RATE	SHINKRAGE_RATE	Planned shrinkage rate.

Field Name	Maps to Column	Description
ACCEPTABLE_EARLY_DAYS	ACCEPTABLE_EARLY_DAYS	Days an order may be early before rescheduling is recommended.
PLANNING_TIME_FENCE	PLANNING_TIME_FENCE_CODE	Derived from MFG_LOOKUPS, where LOOKUP_TYPE = MTL_TIME_FENCE for the meaning column. Use: 1 for Cumulative total lead time, 2 for Cumulative manufacturing lead time, 3 for Total lead time, 4 for user-defined time fence.
DEMAND_TIME_FENCE	DEMAND_TIME_FENCE_CODE	Derived from MFG_LOOKUPS, where LOOKUP_TYPE = MTL_TIME_FENCE for the meaning column. Use: 1 for Cumulative total lead time, 2 for Cumulative manufacturing lead time, 3 for Total lead time, 4 for user-defined time fence.
LEAD_TIME_LOT_SIZE	LEAD_TIME_LOT_SIZE	
STD_LOT_SIZE	STD_LOT_SIZE	Standard lot size.
CUM_MANUFACTURING_LEAD_TIME	CUM_MANUFACTURING_LEAD_TIME	Cumulative lead time.
OVERRUN_PERCENTAGE	OVERRUN_PERCENTAGE	MRP repetitive overrun rate.
MRP_CALCULATE_ATP_FLAG	MRP_CALCULATE_ATP_FLAG	
ACCEPTABLE_RATE_INCREASE	ACCEPTABLE_RATE_INCREASE	MRP repetitive acceptable rate increase.
ACCEPTABLE_RATE_DECREASE	ACCEPTABLE_RATE_DECREASE	MRP repetitive acceptable rate decrease.
CUMULATIVE_TOTAL_LEAD_TIME	CUMULATIVE_TOTAL_LEAD_TIME	
PLANNING_TIME_FENCE_DAYS	PLANNING_TIME_FENCE_DAYS	
DEMAND_TIME_FENCE_DAYS	DEMAND_TIME_FENCE_DAYS	
END_ASSEMBLY_PEGGING_FLAG	END_ASSEMBLY_PEGGING_FLAG	

Field Name	Maps to Column	Description
REPETITIVE_PLANNING_FLAG	REPETITIVE_PLANNING_FLAG	
PLANNING_EXCEPTION_SET	PLANNING_EXCEPTION_SET	Exception Control Set.
BOM_ITEM_TYPE	BOM_ITEM_TYPE	Derived from MFG_LOOKUPS, where LOOKUP_TYPE is BOM_ITEM_TYPE for the Meaning column.
PICK_COMPONENTS_FLAG	PICK_COMPONENTS_FLAG	
REPLENISH_TO_ORDER_FLAG	REPLENISH_TO_ORDER_FLAG	
BASE_ITEM	BASE_ITEM_ID	Derives ITEM_ID from MTL_SYSTEM_ITEMS_KFV for the ITEM and Organization.
ATP_COMPONENTS_FLAG	ATP_COMPONENTS_FLAG	Required.
ATP_FLAG	ATP_FLAG	Required. Indicates whether ATP must be checked when ordering the item.
FIXED_LEAD_TIME	FIXED_LEAD_TIME	Fixed portion of the assembly lead time.
VARIABLE_LEAD_TIME	VARIABLE_LEAD_TIME	Variable lead time.
WIP_SUPPLY_LOCATOR	WIP_SUPPLY_LOCATOR_ID	Derives LOCATION_ID from MTL_ITEM_LOCATIONS_KFV for the concatenated segment.
WIP_SUPPLY_TYPE	WIP_SUPPLY_TYPE	Derived from MFG_LOOKUPS for LOOKUP_TYPE WIP_SUPPLY.
WIP_SUPPLY_SUBINVENTORY	WIP_SUPPLY_SUBINVENTORY	
PRIMARY_UOM_CODE	PRIMARY_UOM_CODE	Primary Unit of Measure Code.
ALLOWED_UNITS_LOOKUP	ALLOWED_UNITS_LOOKUP_CODE	Derived from MFG_LOOKUPS, where LOOKUP_TYPE = MTL_CONVERSION_TYPE for the meaning column's Allowed Units Lookup Code.
COST_OF_SALES_ACCOUNT	COST_OF_SALES_ACCOUNT	Derives ACCOUNT_ID from GL_CODE_COMBINATIONS_KFV for the concatenated segment.

Field Name	Maps to Column	Description
SALES_ACCOUNT	SALES_ACCOUNT	Derives ACCOUNT_ID from GL_CODE_COMBINATIONS_KFV for the concatenated segment.
DEFAULT_INCLUDE_IN_ROLLUP_FLAG	DEFAULT_INCLUDE_IN_ROLLUP_FLAG	Default value for Include In Cost Rollup (Y or N).
INVENTORY_ITEM_STATUS_CODE	INVENTORY_ITEM_STATUS_CODE	
INVENTORY_PLANNING	INVENTORY_PLANNING_CODE	Derives LOOKUP_CODE from MFG_LOOKUPS, where LOOKUP_TYPE = MTL_MATERIAL_PLANNING Inventory Planning Code.
PLANNER_CODE	PLANNER_CODE	
PLANNING_MAKE_BUY	PLANNING_MAKE_BUY_CODE	Derives LOOKUP_CODE from MFG_LOOKUPS, where LOOKUP_TYPE = MTL_PLANNING_MAKE_BUY. Indicates whether the item is planned as manufactured or purchased.
FIXED_LOT_MULTIPLIER	FIXED_LOT_MULTIPLIER	
ROUNDING_CONTROL_TYPE	ROUNDING_CONTROL_TYPE	Derived from MFG_LOOKUPS, where LOOKUP_TYPE = MTL_ROUNDING for the meaning column.
CARRYING_COST	CARRYING_COST	Annual carrying cost.
POSTPROCESSING_LEAD_TIME	POSTPROCESSING_LEAD_TIME	
PREPROCESSING_LEAD_TIME	PREPROCESSING_LEAD_TIME	
FULL_LEAD_TIME	FULL_LEAD_TIME	
ORDER_COST	ORDER_COST	
MRP_SAFETY_STOCK_PERCENT	MRP_SAFETY_STOCK_PERCENT	
MRP_SAFETY_STOCK	MRP_SAFETY_STOCK_CODE	Derives the LOOKUP_CODE from MFG_LOOKUPS for LOOKUP_TYPE MTL_SAFETY_STOCK_TYPE.

Field Name	Maps to Column	Description
MIN_MINMAX_QUANTITY	MIN_MINMAX_QUANTITY	Minimum Minmax Quantity.
MAX_MINMAX_QUANTITY	MAX_MINMAX_QUANTITY	Maximum Minmax Quantity.
MINIMUM_ORDER_QUANTITY	MINIMUM_ORDER_QUANTITY	Minimum Order Quantity.
FIXED_ORDER_QUANTITY	FIXED_ORDER_QUANTITY	
FIXED_DAYS_SUPPLY	FIXED_DAYS_SUPPLY	
MAXIMUM_ORDER_QUANTITY	MAXIMUM_ORDER_QUANTITY	
ATP_RULE_NAME	ATP_RULE_ID	Derives RULE_ID from MTL_ATP_RULES for the RULE_NAME.
PICKING_RULE_NAME	PICKING_RULE_ID	Derives PICKING_RULE_ID from MTL_PICKING_RULES.
RESERVABLE_TYPE	RESERVABLE_TYPE	Derives the LOOKUP_CODE from MFG_LOOKUPS for LOOKUP_TYPE MTL_RESERVATION_CONTROL.
POSITIVE_MEASUREMENT_ERROR	POSITIVE_MEASUREMENT_ERROR	Percent error above measured quantity.
NEGATIVE_MEASUREMENT_ERROR	NEGATIVE_MEASUREMENT_ERROR	Percent error below measured quantity.
ENGINEERING_ECN_CODE	ENGINEERING_ECN_CODE	
ENGINEERING_ITEM	ENGINEERING_ITEM_ID	Derives ITEM_ID from MTL_SYSTEM_ITEMS_KFV for the concatenated segment.
ENGINEERING_DATE	ENGINEERING_DATE	
SERVICE_STARTING_DATE	SERVICE_STARTING_DATE	Used in inbound transactions only.
SERVICE_STARTING_DELAY	SERVICE_STARTING_DELAY	Days after shipment that service begins.
VENDOR_WARRANTY_FLAG	VENDOR_WARRANTY_FLAG	
SERVICEABLE_COMPONENT_FLAG	SERVICEABLE_COMPONENT_FLAG	
SERVICEABLE_PRODUCT_FLAG	SERVICEABLE_PRODUCT_FLAG	

Field Name	Maps to Column	Description
BASE_WARRANTY_SERVICE_ITEM	BASE_WARRANTY_SERVICE_ID	Derives ITEM_ID from MTL_SYSTEM_ITEMS_KFV for the concatenated segment and organization.
PAYMENT_TERMS_NAME	PAYMENT_TERM_ID	Derives PAYMENT_TERM_ID from RA_TERMS.
PREVENTATIVE_MAINTENANCE_FLAG	PREVENTATIVE_MAINTENANCE_FLAG	
PRIMARY_SPECIALIST_FULL_NAME	PRIMARY_SPECIALIST_ID	Derives PERSON_ID from PER_ALL_PEOPLE_F for the active record.
SECONDARY_SPECIALIST_FULL_NAME	SECONDARY_SPECIALIST_ID	Derives PERSON_ID from PER_ALL_PEOPLE_F for the active record.
SERVICEABLE_ITEM_CLASS_NAME	SERVICEABLE_ITEM_CLASS_ID	Derives SERVICEABLE_ITEM_CLASS_ID from CS_SERVICEABLE_ITEM_CLASSES for the Class Name.
TIME_BILLABLE_FLAG	TIME_BILLABLE_FLAG	Indicates whether service hours are billable.
MATERIAL_BILLABLE_FLAG	MATERIAL_BILLABLE_FLAG	Indicates whether service items are billable.
EXPENSE_BILLABLE_FLAG	EXPENSE_BILLABLE_FLAG	Indicates whether service expenses are billable.
PRORATE_SERVICE_FLAG	PRORATE_SERVICE_FLAG	Indicates whether cost of service may be prorated.
COVERAGE_SCHEDULE_NAME	COVERAGE_SCHEDULE_ID	Derives COVERAGE_SCHEDULE_ID from CS_COVERAGE_SCHEDULES for the Name.
SERVICE_DURATION_PERIOD_CODE	SERVICE_DURATION_PERIOD_CODE	
SERVICE_DURATION	SERVICE_DURATION	
WARRANTY_VENDOR_NAME	WARRANTY_VENDOR_ID	Derives VENDOR_ID from PO_VENDORS for the Vendor Name.
MAX_WARRANTY_AMOUNT	MAX_WARRANTY_AMOUNT	

Field Name	Maps to Column	Description
RESPONSE_TIME_PERIOD_CODE	RESPONSE_TIME_PERIOD_CODE	
RESPONSE_TIME_VALUE	RESPONSE_TIME_VALUE	
NEW_REVISION_CODE	NEW_REVISION_CODE	Indicates how to inform customers in case of revisions.
INVOICEABLE_ITEM_FLAG	INVOICEABLE_ITEM_FLAG	Required.
INVOICE_ENABLED_FLAG	INVOICE_ENABLED_FLAG	Required.
MUST_USE_APPROVED_VENDOR_FLAG	MUST_USE_APPROVED_VENDOR_FLAG	Required. Indicates whether purchases are restricted to approved supplier.
OUTSIDE_OPERATION_FLAG	OUTSIDE_OPERATION_FLAG	Required.
OUTSIDE_OPERATION_UOM_TYPE	OUTSIDE_OPERATION_UOM_TYPE	Outside operation unit of measure.
SAFETY_STOCK_BUCKET_DAYS	SAFETY_STOCK_BUCKET_DAYS	
AUTO_REDUCE_MPS	AUTO_REDUCE_MPS	Automatically deletes MPS entries in a period.
COSTING_ENABLED_FLAG	COSTING_ENABLED_FLAG	Required.
CYCLE_COUNT_ENABLED_FLAG	CYCLE_COUNT_ENABLED_FLAG	Required. Indicates whether the item may be cycle counted.
DEMAND_SOURCE_LINE	DEMAND_SOURCE_LINE	Used for inbound transactions only.
COPY_ITEM_NUMBER	COPY_ITEM_ID	Derives ITEM_ID from MTL_SYSTEM_ITEMS_KFV for the concatenated segment. Used for inbound transactions only.
SET_ID	SET_ID	Set identifier used for ATO. Used for inbound transaction only.
REVISION	REVISION	Item revision.
AUTO_CREATED_CONFIG_FLAG	AUTO_CREATED_CONFIG_FLAG	Required. Indicates configuration item automatically created.
ITEM_TYPE	ITEM_TYPE	Derives LOOKUP_CODE from FND_COMMON_LOOKUPS, where LOOKUP_TYPE = ITEM_TYPE for the meaning. User-defined item type.

Field Name	Maps to Column	Description
MODEL_CONFIG_CLAUSE_NAME	MODEL_CONFIG_CLAUSE_NAME	
SHIP_MODEL_COMPLETE_FLAG	SHIP_MODEL_COMPLETE_FLAG	Indicates whether model must be complete to ship.
MRP_PLANNING	MRP_PLANNING_CODE	Derived from MFG_LOOKUPS, where LOOKUP_TYPE = MRP_PLANNING_CODE for the meaning column.
RETURN_INSPECTION_REQUIREMENT	RETURN_INSPECTION_REQUIREMENT	Derived from MFG_LOOKUPS, where LOOKUP_TYPE = MTL_RETURN_INSPECTION. RMA Inspection Requirement.
DEMAND_SOURCE_TYPE	DEMAND_SOURCE_TYPE	Derives TRANSACTION_SOURCE_TYPE_ID from MTL_TXN_SOURCE_TYPES for TRANSACTION_SOURCE_TYPE_NAME. Used for inbound transactions only.
DEMAND_SOURCE_HEADER_ID	DEMAND_SOURCE_HEADER_ID	
TEMPLATE_NAME	TEMPLATE_NAME	Used for inbound transactions only.
COPY_ORGANIZATION_NAME	COPY_ORGANIZATION_ID	Derives ORGANIZATION_ID from ORG_ORGANIZATION_DEFINITIONS. Used for inbound transactions only.
ATO_FORECAST_CONTROL	ATO_FORECAST_CONTROL	Derived from MFG_LOOKUPS where LOOKUP_TYPE= MRP_ATO_FORECAST_CONTROL for the meaning column. Type of Forecast Control for ATO.
TRANSACTION_TYPE	TRANSACTION_TYPE	Use CREATE or UPDATE. Item cost cannot be updated using the UPDATE mode in this interface. For inbound transactions only.
MATERIAL_COST	MATERIAL_COST	Used for inbound transactions only.

Field Name	Maps to Column	Description
MATERIAL_SUB_ELEMENT	MATERIAL_SUB_ELEMENT	Used for inbound transactions only.
MATERIAL_OH_RATE	MATERIAL_OH_RATE	Material overhead rate. For inbound transactions only.
MATERIAL_OH_SUBELEMENT	MATERIAL_OH_SUBELEMENT	Material overhead sub element. For inbound transactions only.
CONTAINER_ITEM_FLAG	CONTAINER_ITEM_FLAG	Indicates whether the item is a container. Used for shipping sales orders.
VEHICLE_ITEM_FLAG	VEHICLE_ITEM_FLAG	Indicates whether the item is a vehicle. Used for shipping sales orders.
MAXIMUM_LOAD_WEIGHT	MAXIMUM_LOAD_WEIGHT	Maximum load weight of a container or vehicle that can be used for shipping sales orders.
MINIMUM_FILL_PERCENT	MINIMUM_FILL_PERCENT	Minimum fill condition under which the container or vehicle should be used.
CONTAINER_TYPE_CODE	CONTAINER_TYPE_CODE	User-defined container type code for container items.
INTERNAL_VOLUME	INTERNAL_VOLUME	Internal volume for container items. Used by Shipping to calculate container capacity restrictions.

1.1. MTL_ITEM_REVISIONS (Maps to MTL_ITEM_REVISIONS_INTERFACE)

Field Name	Maps to Column	Description
REVISION	REVISION	Item revision code.
TRANSACTION_TYPE	TRANSACTION_TYPE	Must use CREATE. Used for inbound transactions only.
CHANGE_NOTICE	CHANGE_NOTICE	
ECN_INITIATION_DATE	ECN_INITIATION_DATE	Engineering Change Initiation Date.
IMPLEMENTATION_DATE	IMPLEMENTATION_DATE	
IMPLEMENTED_SERIAL_NUMBER	IMPLEMENTED_SERIAL_NUMBER	

Field Name	Maps to Column	Description
EFFECTIVITY_DATE	EFFECTIVITY_DATE	
REVISED_ITEM_SEQUENCE_ID	REVISED_ITEM_SEQUENCE_ID	Identifies multiple occurrences of the same item on an engineering change order.
DESCRIPTION	DESCRIPTION	Item revision description.

Receive Master Schedule Service

The name of this service is:

WmOAMFG107SC.masterScheduling107SC.intoOA.receiveMasterSchedule

This service imports master schedules.

The Planning Manager scans for rows with PROCESS_STATUS = 2. These records are then validated according to the rules in the Implementation manuals. If they are successfully processed, then the PROCESS_STATUS is set to 5. If they contain some type of validation error, then PROCESS_STATUS is set to 4 and the ERROR_MESSAGE is populated appropriately. The records with PROCESS_STATUS = 5 will remain in the interface table for the number of days specified by the profile option MRP:Interface Table History Days.

The Master Schedule Interface Load Program performs the processing. The Planning Manager periodically checks the Master Schedule Interface tables to determine new rows for processing.

Since the Planning Manager runs in asynchronous mode, this service does not execute concurrent processes.

Database Scripts

This service does not use database scripts.

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definition:

- setMasterScheduleTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

This service calls the following services to import master schedules:

- **specifyDefaultSettings** specifies the default parameter settings required for service execution. You should change these settings accordingly.
- **bizDocMapping** maps the incoming business document structure to the required Oracle Applications data structures (the interface tables).
 - **getOrgId** is a common utility and is used as one of the transformers while mapping the business doc IData structure to the interface table IData structure. It takes ORGANIZATION_NAME as the input parameter, and queries the table ORG_ORGANIZATION_DEFINITIONS to get the ORGANIZATION_ID corresponding to the ORGANIZATION_NAME.
 - **getProjectId** is used as one of the transformers while mapping the bizDoc to the interface table data structure. It takes the project name as input, and returns the project ID by querying the Oracle Applications database.
 - **getTaskId** gets the task ID from the Oracle Applications database. It takes the task name and organization name as input, and finds the matching task ID.
 - **getInventoryItemId** is used as one of the transformers while mapping the bizDoc to the interface table data structure. The service gets the inventory item ID from the input of item number and organization name.
 - **getOASystemDateObject** returns the Oracle Applications system date as a date object.
 - **getOracleAppsUserId** is a transformer for mapping the business document IData structure to the interface table IData structure. It takes ORACLE_APPS_USER_NAME as the input parameter from the business document, and queries the table FND_USER to get the USER_ID. The USER_ID information is required for insertion into the interface tables.
- **setMasterScheduleTxn** inserts data into the interface table. It extracts data from the IData structure that results from the bizDocMapping service, and puts the data into the interface table in Oracle Applications for Master Schedule.

For more details on receive transactions, see [“Overview of Receive Service Transaction Processing” on page 31](#).

Business Document Structure

This services uses the business document structure masterScheduleBizDoc. Its structure is as follows:

■ MRP_SCHEDULE

MRP_SCHEDULE (Maps to MRP_SCHEDULE_INTERFACE)

Field Name	Maps to Column	Description
ORGANIZATION_NAME	ORGANIZATION_ID	Required. Derived from ORG_ORGANIZATION_DEFINITIONS.
INVENTORY_ITEM	INVENTORY_ITEM_ID	Required. Derived from MTL_SYSTEM_ITEMS_B_KFV.
SCHEDULE_DESIGNATOR	SCHEDULE_DESIGNATOR	Required. Master Schedule Identifier.
SCHEDULE_DATE	SCHEDULE_DATE	Required.
RATE_END_DATE	RATE_END_DATE	
SCHEDULE_QUANTITY	SCHEDULE_QUANTITY	Required.
SCHEDULE_COMMENTS	SCHEDULE_COMMENTS	
WORKDAY_CONTROL	WORKDAY_CONTROL	Indicates the action that the Master Schedule Interface Load takes if a schedule date is not a valid workday. Use: 1 for Reject (default), 2 for Shift forward, 3 for Shift backward. If WORKDAY_CONTROL is set to Null, the Master Schedule Interface Load program assumes the default value.
TRANSACTION_ID	TRANSACTION_ID	Required for replacement items; it must match a record in MRP_SCHEDULE_DATES table.
SOURCE_CODE	SOURCE_CODE	Required. Identifies external source.

Field Name	Maps to Column	Description
SOURCE_LINE_ID	SOURCE_LINE_ID	Required. External Source Line Identifier.
PROJECT_NAME	PROJECT_ID	Derives PROJECT_ID from PA_PROJECTS_ALL table.
TASK_NUMBER	TASK_ID	Derives Task Id from PA_TASKS table.

Receive Move Transaction Service

The name of this service is:

WmOAMFG107SC.workInProgress107SC.intoOA.moveTransaction:receiveMoveTransaction

This service imports moved items.

The concurrent process of the Move Transaction Manager is scheduled from the Oracle Applications system. Generally, managers run continuously at regular process intervals determined at startup. They control the number of transaction workers, processing intervals, and transactions processed by the worker. The mode of operation for these managers is governed by the setting of profile options prefixed by the letters TP (transaction processing). Therefore, this inbound transaction has no custom package.

Database Scripts

This service does not use any database scripts.

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definitions:

- setMoveTransactionTxn107SC.txp
- MoveTransactionTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

This service calls the following services to import moved items:

- **specifyDefaultSettings** specifies the default parameter settings required for service execution. You should change these settings accordingly.
- **bizDocMapping** maps the incoming business document structure to the required Oracle Applications data structures (the interface tables).
 - **getOASystemDateObject** returns the Oracle Applications system date as a date object.
 - **getOracleAppsUserId** is a transformer for mapping the business document IData structure to the interface table IData structure. It takes ORACLE_APPS_USER_NAME as the input parameter from the business document, and queries the table FND_USER to get the USER_ID. The USER_ID information is required for insertion into the interface tables.
 - **getOrgIdAndCode** returns the ORGANIZATION_ID and ORGANIZATION_CODE corresponding to the supplied ORGANIZATION_NAME. This service is used as a transformer in bizDocMapping.
 - **getInventoryItemId** returns the INVENTORY_ITEM_ID corresponding to the supplied ITEM_CODE and the ORGANIZATION_NAME. This service is used as a transformer in bizDocMapping.
 - **getAcctPeriodId** returns the ACCOUNT_PERIOD_ID corresponding to the supplied ACCOUNT_PERIOD_NAME and ORGANIZATION_NAME. This service is used as a transformer in bizDocMapping.
 - **getCodeCombinationId** returns the SCRAP_ACCOUNT_ID corresponding to the supplied SCRAP_ACCOUNT_NUMBER and the SET_OF_BOOKS_NAME. This service is used as a transformer in bizDocMapping.
 - **getWIPEntityType** returns the ENTITY_TYPE corresponding to the supplied WIP_ENTITY_NAME and ORGANIZATION_NAME.
 - **convertToDateObject** returns the outDate as a date object corresponding to the supplied inDate which is in text format. This service is used as a transformer in bizDocMapping.
- **setMoveTransactionTxn** service inserts data into the interface table. It takes data from the IData structure that results from the bizDocMapping service, and puts the data into the interface table in Oracle Applications for Open Move Interface.

For more details on receive transactions, see [“Overview of Receive Service Transaction Processing” on page 31](#).

Business Document Structure

This service uses the business document MoveTransactionBizDoc. Its structure is as follows:

■ WIP_MOVE_TXN

WIP_MOVE_TXN (Maps to WIP_MOVE_TXN_INTERFACE)

Field Name	Maps to Column	Description
SOURCE_CODE	SOURCE_CODE	Code of the shop floor control system that generated the transaction record; it is null if created by Oracle Work In Process.
SOURCE_LINE_ID	SOURCE_LINE_ID	Identifier used by source shop floor control system; it is null if created by Oracle Work In Process.
TRANSACTION_TYPE	TRANSACTION_TYPE	Transaction type. Use: 1 for Move transaction 2 for Move completed transaction 3 for Move return transaction.
ORGANIZATION_NAME	ORGANIZATION_ID	
SET_OF_BOOKS_NAME		
WIP_ENTITY_NAME	WIP_ENTITY_NAME	Required. WIP job or repetitive assembly name.
PRIMARY_ITEM_CODE	PRIMARY_ITEM_ID	Inventory item identifier for assembly that the job or schedule creates.
LINE_CODE	LINE_CODE	Required for repetitive manufacturing transactions.
TRANSACTION_DATE	TRANSACTION_DATE	Required. Date when transaction was performed.
ACCT_PERIOD_NAME	ACCT_PERIOD_ID	Accounting period identifier.
FM_OPERATION_SEQ_NUM	FM_OPERATION_SEQ_NUM	From operation sequence number. Required for Move and Move Completion transactions. It must be an enabled intraoperation step.
FM_OPERATION_CODE	FM_OPERATION_CODE	From operation code.

Field Name	Maps to Column	Description
FM_DEPARTMENT_CODE	FM_DEPARTMENT_CODE	From department code.
FM_INTRAOPERATION_STEP_TYPE	FM_INTRAOPERATION_STEP_TYPE	From intraoperation step. Required for Move and Move Completion transactions. It must be an enabled intraoperation step.
TO_OPERATION_SEQ_NUM	TO_OPERATION_SEQ_NUM	To operation sequence number. Required when performing Move and Move Return transactions. It must be an enabled intraoperation step.
TO_OPERATION_CODE	TO_OPERATION_CODE	
TO_DEPARTMENT_CODE	TO_DEPARTMENT_CODE	
TO_INTRA_OPERATION_STEP_TYPE	TO_INTRA_OPERATION_STEP_TYPE	To operation step type. Required when performing Move and Move Return transactions. It must be an enabled intraoperation step.
TRANSACTION_QUANTITY	TRANSACTION_QUANTITY	Required. Transaction quantity between operations.
TRANSACTION_UOM	TRANSACTION_UOM	Required. Unit of measure used in transaction.
PRIMARY_QUANTITY	PRIMARY_QUANTITY	Quantity of assembly being moved between operations expressed in the assembly's primary unit of measure.
PRIMARY_UOM	PRIMARY_UOM	Primary unit of measure for the assembly.
SCRAP_ACCOUNT_NUMBER	SCRAP_ACCOUNT_ID	General Ledger account charged when material is scrapped.
REASON_NAME	REASON_NAME	Standard transaction reason name.
REFERENCE	REFERENCE	Transaction reference descriptive text.
QA_COLLECTION_ID	QA_COLLECTION_ID	Collection identifier for quality results.

Receive Quality Collection Service

The name of this service is:
WmOAMFG107SC.quality107SC.intoOA.qualityCollection:receiveQualityCollection

This service adds new quality results to, or updates existing quality results in, the Oracle Applications Quality Data Repository.

While submitting the concurrent program Import Quality Collection to load the Quality Collection records from interface tables to production tables, it is not possible to restrict processing to only the records uploaded by the IS flow instance. The program will process all unprocessed records present in the interface table at the time of execution. While retrieving errors, the program cannot be guaranteed to fetch errors pertaining to the flow instance. While showing the error, it may display a historical error although the current import process may have been successful. We suggest that you keep the interface table free of error records to minimize ambiguity.

Database Scripts

This service uses the following database scripts:

Script	Description
wm_install_into_qacollection.sql	Runs the scripts listed below, except the uninstall script.
wm_into_qacollection_pkg.sql	Installs WM_QA_IMP_HANDLER_PKG.WM_HANDLE_QA, which calls the WM_CONC_REQUEST.WM_REQUEST_SUBMIT procedure to submit the Quality Collection Import process. The process in the QA module is Collection Import.
wm_drop_into_qacollection.sql	Uninstalls all components created by wm_install_into_qacollection.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definitions:

- `setQualityCollectionTxn107SC.txp`
- `QualityCollectionTransactions107SC.txp`

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

This service calls the following services to add or update quality results:

- `specifyDefaultSettings` specifies the default parameter settings required for service execution. You should change these settings accordingly.
- `specifyConcProgParams` is used to specify the default parameter settings required for concurrent program execution for Quality Collection. You should change these settings accordingly.
- `bizDocMapping` maps the incoming business document structure to the required Oracle Applications data structures (the interface tables).
 - `getWIPEntityId` is used as one of the transformers while mapping the business doc that takes the `WIP_ENTITY_NAME` as input, and returns `WIP_ENTITY_ID`.
 - `getOrgIdAndCode` is used as one of the transformers while mapping the business document `Idata` structure to the interface table `Idata` structure. It takes `ORGANIZATION_NAME` as the input parameter, and queries the table `ORG_ORGANIZATION_DEFINITIONS` to get the `ORGANIZATION_ID` and `ORGANIZATION_CODE` corresponding to the `ORGANIZATION_NAME`.
- `importQualityCollection` imports data to the production table from the interface table. It calls the service `execQualityCollectionConcProg` with the appropriate parameters to execute the corresponding concurrent program that inserts data into the production table, and to generate the error/ acknowledgement message. If the status of the execution is `SUCCESS` (returned by the service `execQualityCollectionConcProg`), it checks for the record in the interface table. If it find a record, it indicates an error during import. In this case, the service calls `getQualityCollectionImport_ERR` to retrieve the errors. If no record is found, it comes out of the flow, indicating success of the data import process. If the status of the execution is `FAILED`, it comes out of execution.
 - `execQualityCollectionConcProg` invokes the stored procedure `WM_QA_IMP_HANDLER_PKG.WM_HANDLE_QA`. This procedure calls a corresponding concurrent subroutine to execute the data import process for Quality Collection into Oracle Applications. This service gives Status Id, Request

Id, Execution Status Message (in case of normal completion of concurrent program), and database Stored Procedure error message (in case an exception occurs in Stored Procedure execution).

- **checkQualityCollectionImportStatus** checks the status of the execution by checking the interface table for any rejected records corresponding to the current request_id. If the query does not return any records, it indicates successful import. If the query returns a record, it indicates that the concurrent program could not import data successfully in the production tables of Oracle Applications.
- **getQualityCollectionImport_ERR** gets the error message that occurs during the data import from the interface table to the production table. It scans the tables QA_INTERFACE_ERRORS and QA_RESULTS_INTERFACE to get the erroneous records and corresponding messages.

For more details on receive transactions, see [“Overview of Receive Service Transaction Processing” on page 31](#).

Business Document Structure

This service uses the business document QualityCollectionBizDoc. Its structure is as follows:

- QA_COLLECTION

QA_COLLECTION (Maps to QA_RESULTS_INTERFACE)

Field Name	Maps to Column	Description
SOURCE_CODE	SOURCE_CODE	Source of quality data.
SOURCE_LINE_ID	SOURCE_LINE_ID	Detailed information about quality data.
ORGANIZATION_NAME	ORGANIZATION_CODE	Required. Derives the Organization Code from ORG_ORGANIZATION_DEFINITIONS.
PLAN_NAME	PLAN_NAME	Required. Name of the collection plan as defined in Oracle Quality.
SPEC_NAME	SPEC_NAME	Quality specification name.
DEPARTMENT	DEPARTMENT	Department code of the Bills of Material department.

Field Name	Maps to Column	Description
RESOURCE_CODE	RESOURCE_CODE	
QUANTITY	QUANTITY	
ITEM	ITEM	Item number.
UOM	UOM	Unit of measure.
REVISION	REVISION	
SUBINVENTORY	SUBINVENTORY	
LOCATOR	LOCATOR	Locator of the item.
LOT_NUMBER	LOT_NUMBER	
SERIAL_NUMBER	SERIAL_NUMBER	
COMP_ITEM	COMP_ITEM	Component item number.
COMP_UOM	COMP_UOM	Component unit of measure.
COMP_REVISION	COMP_REVISION	Component revision.
COMP_SUBINVENTORY	COMP_SUBINVENTORY	Component subinventory.
COMP_LOCATOR	COMP_LOCATOR	Component locator.
COMP_LOT_NUMBER	COMP_LOT_NUMBER	Component lot number.
COMP_SERIAL_NUMBER	COMP_SERIAL_NUMBER	Component serial number.
WIP_ENTITY_NAME	WIP_ENTITY_ID	Derived from WIP_ENTITIES table.
JOB_NAME	JOB_NAME	
PRODUCTION_LINE	PRODUCTION_LINE	
TO_OP_SEQ_NUM	TO_OP_SEQ_NUMBER	To operating sequence number.
FROM_OP_SEQ_NUM	FROM_OP_SEQ_NUMBER	From operating sequence number.
VENDOR_NAME	VENDOR_NAME	Name of the supplier.
RECEIPT_NUM	RECEIPT_NUM	Receipt number.
PO_NUMBER	PO_NUMBER	Purchase order number.
PO_LINE_NUM	PO_LINE_NUM	Purchase order line number.
PO_SHIPMENT_NUM	PO_SHIPMENT_NUM	Purchase order shipment number.
CUSTOMER_NAME	CUSTOMER_NAME	

Field Name	Maps to Column	Description
SALES_ORDER	SALES_ORDER	Sales order number.
RMA_NUMBER	RMA_NUMBER	Return material authorization number.
CHARACTER1 through CHARACTER40	CHARACTER1 through CHARACTER40	These columns are planning-specific. A collection plan defined in Oracle Applications can have one or more collection elements. The different collection elements are mapped to the above columns.
TO_DEPARTMENT	TO_DEPARTMENT	To Bills of Material department.
PO_RELEASE_NUM	PO_RELEASE_NUM	Purchase order release number.
CHARACTER41 through CHARACTER100	CHARACTER41 through CHARACTER100	These columns are planning-specific. A collection plan defined in Oracle Applications can have one or more collection elements. The different collection elements are mapped to the above columns.

Receive Replenishment Service

The name of this service is:

```
WmOAMFG107SC.inventory107SC.intoOA.replenishment:receiveReplenishment
```

This service imports replenishment requests from external systems, such as barcode applications. These requests can be stock-take counts or requisition requests for subinventories where quantities are not tracked.

Database Scripts

This service uses the following database scripts:

Script	Description
wm_install_into_replenish.sql	Runs the scripts listed below, except the uninstall script.
wm_into_replenish_pkg.sql	Installs WM_OPEN_REPLENISH_HANDLER_PKG.WM_HANDLE_OPEN_REPLENISH, which calls the WM_CONC_REQUEST.WM_REQUEST_SUBMIT procedure to submit the replenishment Import process.
wm_drop_into_replenish.sql	Uninstalls all components created by wm_install_into_replenish.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definitions:

- setReplenishmentTxn107SC.txp
- ReplenishmentTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

This service calls the following services to import replenishment requests:

- **specifyDefaultSettings** specifies the default parameter settings required for service execution. You should change these settings accordingly.
- **specifyConcProgParams** specifies the default settings required for concurrent program execution. You should change these settings accordingly.
- **bizDocMapping** maps the incoming business document structure to the required Oracle Applications data structures (the interface tables).
 - **getUOMCode** is used as one of the transformers while mapping the incoming business document to the interface table structure. This service selects the UOM_CODE from given UNIT_OF_MEASURE string.

- **getLocationId** is one of the transformers that derives the `LOCATION_ID` from the Location Address provided.
- **getOrgId** is one of the transformers that derives the `ORGANIZATION_ID` from the `ORGANIZATION_NAME` provided in the business document.
- **getInventoryItemId** derives the `INVENTORY_ITEM_ID` from the `ITEM_CODE` and `ORGANIZATION_NAME` provided in the business document.
- **convertToDateObject** converts the incoming date information from string type to an object type. Primarily, this is required for storing the timestamp information.
- **getOASystemDateObject** returns the Oracle Applications system date as a date object.
- **getOracleAppsUserId** is a transformer for mapping the business document `IData` structure to the interface table `IData` structure. It takes `ORACLE_APPS_USER_NAME` as the input parameter from the business document, and queries the table `FND_USER` to get the `USER_ID`. The `USER_ID` information is required for insertion into the interface tables.
- **getTaskId** service is used as a transformer to derive the `TASK_ID` from the `PROJECT_NAME` and `TASK_NUMBER`.
- **setReplenishmentTxn** inserts data into the interface table. It extracts data from the `IData` structure that results from the `bizDocMapping` service, and puts the data into the interface table in Oracle Applications for Replenishment.
- **importReplenishment** imports data from the interface table to the production table. It then calls the services `execReplenishmentConcProg`, `checkReplenishmentImportStatus`, and `getReplenishmentImport_ERR` to execute the corresponding concurrent program that inserts data into the production table, and to generate the error/ acknowledgement message. If the execution status is `SUCCESS` (returned by the service `execReplenishmentConcProg`), it checks for a record in the error interface table using the returned request ID. If a record is found, it indicates an error during import and calls `getReplenishmentImport_ERR` to retrieve the errors. If no record is found, it indicates that the data import process was successful. If the status of the execution is `FAILED`, it returns an appropriate error message.
- **execReplenishmentConcProg** invokes the stored procedure `WM_OPEN_REPLENISH_HANDLER_PKG.WM_HANDLE_OPEN_REPLENISH`. This procedure calls the corresponding concurrent subroutine to execute the data import process for replenishment into Oracle Applications. This service gives Status Id, Request Id, Execution Status Message (if the concurrent program completes normally), and database Stored Procedure error message (if an exception occurs in Stored Procedure execution).
- **mapExecStatus** is used to capture the concurrent program execution status. If status returned is `SUCCESS`, it checks if the data has been uploaded into production database tables.

- **mapExecErrorStatus** is used to capture the concurrent program execution error message. If status returned is FAILED, it maps relevant messages to dbErrorMsg and concProgMsg, and returns back to the calling flow service (receiveReplenishment).
- **checkReplenishmentImportStatus** checks the status of the execution by checking the Error Interface table. If the query does not return any rows, it indicates a successful import. If rows exist in the Error Interface table, it indicates that the concurrent program could not import data successfully into Oracle Applications production tables.
- **getReplenishmentImport_ERR** gets the error message that occurs during the data import. Based on the parameter Count Name, it scans the tables MTL_REPLENISH_HEADERS_INT and MTL_REPLENISH_LINES_INT to get the corresponding message matches to the count name.

For more details on receive transactions, see [“Overview of Receive Service Transaction Processing” on page 31](#).

Business Document Structure

This service uses the business document ReplenishmentBizDoc. Its structure is as follows:

- 1.0. MTL_REPLENISH_HEADERS
 - 1.1. MTL_REPLENISH_LINES

1.0. MTL_REPLENISH_HEADERS (Maps to MTL_REPLENISH_HEADERS_INT)

Field Name	Maps to Column	Description
REPLENISHMENT_COUNT_NAME	REPLENISHMENT_COUNT_NAME	Required. Unique name for the replenishment count.
COUNT_DATE	COUNT_DATE	Required. Date on which the count was performed.
ORGANIZATION_NAME	ORGANIZATION_ID	Required. Name of organization from which the replenishment count originated. Use HR_ALL_ORGANIZATION_UNITS.NAME to derive the ORGANIZATION_ID. See the commonOA107SC.utils:getOrgId transformer.
SUBINVENTORY_CODE	SUBINVENTORY_CODE	Required. Code of the subinventory that is the source of the replenishment count.

Field Name	Maps to Column	Description
SUPPLY_CUTOFF_DATE	SUPPLY_CUTOFF_DATE	Date after which planned supply will not be considered in available quantity calculations. A null value here indicates that you do not want to consider planned supply when performing replenishment calculations.
DELIVER_TO_LOCATION_ADDRESS_1 DELIVER_TO_LOCATION_ADDRESS_2 DELIVER_TO_LOCATION_ADDRESS_3 DELIVER_TO_LOCATION_TOWN_OR_CITY DELIVER_TO_LOCATION_COUNTY DELIVER_TO_LOCATION_STATE DELIVER_TO_LOCATION_POSTAL_CODE DELIVER_TO_LOCATION_COUNTRY	DELIVERY_LOCATION_ID	Location to which the replenishment should be delivered. If neither is specified, the default delivery location for the organization from which the replenishment originated is defaulted by the system. Use HR_LOCATIONS to get the LOCATION_ID.

1.1. MTL_REPLENISH_LINES (Maps to MTL_REPLENISH_LINES_INT)

Field Name	Maps to Column	Description
ORGANIZATION_NAME	ORGANIZATION_ID	Name of organization from which the replenishment count originated. The line organization must be same as that of the header. Use HR_ALL_ORGANIZATION_UNITS.NAME to derive the ORGANIZATION_ID. See the commonOA107SC.utils:getOrgId transformer.
ITEM_CODE	INVENTORY_ITEM_ID	Required. Item to be replenished. Derives the Inventory Item ID using MTL_SYSTEM_ITEMS_KFV.CONCATENATED_SEGMENTS for the given ORGANIZATION_ID. See the commonOA107SC.utils:getOrgId transformer.
COUNT_TYPE_CODE	COUNT_TYPE_CODE	Required. Enter the type of the replenishment count entry. Use: 1 for On-hand quantity, 2 for Order quantity, 3 for Order maximum.
COUNT_QUANTITY	COUNT_QUANTITY	Required for all types except Order Maximum. The count quantity for the count type entered for the line.
REFERENCE	REFERENCE	Replenishment count reference information.
COUNT_UNIT_OF_MEASURE	COUNT_UOM_CODE	Required. Unit of measure code used for the count. This column is meaningful only when a value is entered in the COUNT_QUANTITY columns. Use MTL_UNITS_OF_MEASURE.UNIT_OF_MEASURE to get the UOM_CODE in the user's LANGUAGE.

Field Name	Maps to Column	Description
PROJECT	TASK_ID	Use PA_TASKS.TASK_NAME and PA_PROJECTS_ALL.NAME to get the TASK_ID.
TASK		

Receive Resource Transaction Service

The name of this service is:

WmOAMFG107SC.workInProgress107SC.intoOA.resourceTransaction:receiveResourceTransaction

This service imports work-in-process (WIP) resource transactions. The Open Resource Transaction Interface program validates the data you load in the Open Resource Transaction Interface tables by ensuring that the columns in the interface tables reference the appropriate values and columns in the rest of the system.

The Cost Manager handles the import process. Interface managers run continuously at regular intervals. They control the number of transaction workers, processing intervals, and number of transactions processed by the worker. The mode of operation for these managers is governed by the setting of profile options prefixed by the letters TP (transaction processing). The Cost Manager is also responsible for costing for other transactions, such as material transactions. Thus, no concurrent process is defined in this service.

Database Scripts

This service does not use database scripts.

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definitions:

- setResourceTransactionTxn107SC.txp
- ResourceTransactionTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

This service calls the following services to import resource transactions:

- **specifyDefaultSettings** specifies the default parameter settings required for service execution. You should change these settings accordingly.
- **bizDocMapping** maps the incoming business document structure to the required Oracle Applications data structures (the interface tables).
 - **getOrgIdAndCode** is used as one of the transformers while mapping the business doc IData structure to the interface table IData structure. It takes ORGANIZATION_NAME as the input parameter, and queries the table ORG_ORGANIZATION_DEFINITIONS to get the ORGANIZATION_ID and the ORGANIZATION_CODE.
 - **getOASystemDateObject** returns the Oracle Applications system date as a date object.
 - **getOracleAppsUserId** is a transformer for mapping the business document IData structure to the interface table IData structure. It takes ORACLE_APPS_USER_NAME as the input parameter from the business document, and queries the table FND_USER to get the USER_ID. The USER_ID information is required for insertion into the interface tables.
 - **getAcctPeriodId** is used as one of the transformers while mapping the incoming business document to the interface table data structure for Open resource interface. This service takes Period Name as an input, and returns Period ID as output querying the Oracle Applications database.
 - **getActivityId** is used as one of the transformers while mapping the incoming business document to the interface table data structure for the Open resource interface. This service takes Activity Name as an input, and returns Activity ID as output querying the Oracle Applications database.
 - **getDepartmentId** is used as one of the transformers while mapping the incoming business document to the interface table data structure for the Open resource interface. This service takes Department Code and Organization Name as input, and returns Department ID as output querying the Oracle Applications database.
 - **getPOHeaderId** is used as one of the transformers while mapping the incoming business document to the interface table data structure for Open resource interface. This service takes Purchase Order Header Number as an input and returns PO Header ID as output querying the Oracle Applications database.
 - **getPOLineId** is used as one of the transformers while mapping the incoming business document to the interface table data structure for the Open resource interface. This service takes PO Header Number and Line Number as input, and returns corresponding PO Line ID as output querying the Oracle Applications database.

- `getCodeCombinationId` is used as one of the transformers while mapping the incoming business document to the interface table data structure for the Open resource interface. This service takes concatenated accounting segments and a set of book names as input, and returns the account ID.
- `setResourceTransactionTxn` inserts data into the interface table. It extracts data from the `Idata` structure that results from the `bizDocMapping` service, and puts the data into the interface table in Oracle Applications for the Open Resource Interface.

For more details on receive transactions, see [“Overview of Receive Service Transaction Processing” on page 31](#).

Business Document Structure

This service uses the business document `ResourceTransactionBizDoc`. Its structure is as follows:

- `WIP_COST_TXN`

`WIP_COST_TXN` (Maps to `WIP_COST_TXN_INTERFACE`)

Field Name	Maps to Column	Description
<code>TRANSACTION_TYPE</code>	<code>TRANSACTION_TYPE</code>	Required. Use: 1 for Normal resource transactions, 2 for Overhead transactions, 3 for Outside processing transactions.
<code>SOURCE_CODE</code>	<code>SOURCE_CODE</code>	Code of the shop floor control system that generated the transaction record.
<code>ORGANIZATION_NAME</code>	<code>ORGANIZATION_CODE</code>	Required. Short code that uniquely identifies the organization.
<code>SET_OF_BOOKS_NAME</code>		
<code>WIP_ENTITY_NAME</code>	<code>WIP_ENTITY_NAME</code>	Required. WIP job or repetitive assembly name.
<code>LINE_CODE</code>	<code>LINE_CODE</code>	
<code>PRIMARY_ITEM_CODE</code>	<code>PRIMARY_ITEM_ID</code>	Inventory item identifier of assembly that job or schedule creates.

Field Name	Maps to Column	Description
TRANSACTION_DATE	TRANSACTION_DATE	Required. Date transaction was performed. Time component of this date should also be entered. Currently the date time format for TRANSACTION_DATE is set to dd-MMM-yy HH:mm:ss.
ACCT_PERIOD_NAME	ACCT_PERIOD_ID	Accounting period identifier.
OPERATION_SEQ_NUM	OPERATION_SEQ_NUM	Required. Operation sequence number within a routing.
RESOURCE_SEQ_NUM	RESOURCE_SEQ_NUM	Required. Resource sequence number.
DEPARTMENT_CODE	DEPARTMENT_ID	Short code that uniquely identifies the department.
EMPLOYEE_NUM	EMPLOYEE_NUM	
RESOURCE_CODE	RESOURCE_CODE	Resource code.
RESOURCE_TYPE	RESOURCE_TYPE	Resource type.
USAGE_RATE_OR_AMOUNT	USAGE_RATE_OR_AMOUNT	Rate per assembly, amount per job, or amount per schedule.
BASIS_TYPE	BASIS_TYPE	Basis for charging resource.
AUTOCHARGE_TYPE	AUTOCHARGE_TYPE	Method of charging the resource.
STANDARD_RATE_FLAG	STANDARD_RATE_FLAG	Indicates whether the resource is charged at the standard rate.
TRANSACTION_QUANTITY	TRANSACTION_QUANTITY	Required. Transaction quantity.
TRANSACTION_UOM	TRANSACTION_UOM	Required. Unit of measure used for transaction.
PRIMARY_QUANTITY	PRIMARY_QUANTITY	Quantity of transaction expressed in assembly's primary unit of measure.
PRIMARY_UOM	PRIMARY_UOM	Primary unit of measure for the resource.
PRIMARY_UOM_CLASS	PRIMARY_UOM_CLASS	Class to which the primary unit of measure belongs.
ACTUAL_RESOURCE_RATE	ACTUAL_RESOURCE_RATE	Actual rate of the resource.
CURRENCY_CODE	CURRENCY_CODE	Unique identifier for the currency.

Field Name	Maps to Column	Description
CURRENCY_CONVERSION_DATE	CURRENCY_CONVERSION_DATE	
CURRENCY_CONVERSION_RATE	CURRENCY_CONVERSION_RATE	
CURRENCY_ACTUAL_RESOURCE_RATE	CURRENCY_ACTUAL_RESOURCE_RATE	Actual resource rate in foreign currency; used when the STANDARD_RATE_FLAG is Y.
ACTIVITY_NAME	ACTIVITY_NAME	
REASON_NAME	REASON_NAME	Standard transaction reason name.
REFERENCE	REFERENCE	Required.
MOVE_TRANSACTION_ID	MOVE_TRANSACTION_ID	Transaction identifier for the move that created the costing transaction.
RCV_TRANSACTION_ID	RCV_TRANSACTION_ID	Transaction identifier for the receipt that created the costing transaction.
PO_HEADER_NUM	PO_HEADER_ID	Purchase order identifier that the receipt is against.
PO_LINE_NUM	PO_LINE_ID	Purchase order line identifier that the receipt is against.
RECEIVING_ACCOUNT_NUMBER	RECEIVING_ACCOUNT_ID	Receiving account identifier.
PROJECT_NUMBER	PROJECT_NUMBER	Required.
TASK_NUMBER	TASK_NUMBER	Required.

Send Engineering BOM Service

The name of this service is:

WmOAMFG107SC.engineering107SC.fromOA.engineeringBOM:sendEngineeringBOM

This service sends to the webMethods Integration Server information on the parent, component, and substitute Engineering Bills of Material items, as well as revisions.

Database Scripts

This service uses the following database scripts:

Script	Description
wm_install_from_engbom.sql	Runs the scripts listed below, except the uninstall script.
wm_from_engbom_vw.sql	Creates the following required view components for Engineering Bills of Material outbound transactions: <ul style="list-style-type: none"> ■ WM_ENG_BOM_BILL_OF_MTLS_VW ■ WM_ENG_BOM_ITEM_REVISIONS_VW ■ WM_ENG_BOM_INVENTORY_COMPS_VW ■ WM_ENG_BOM_SUBSTITUTE_COMPS_VW ■ WM_ENG_BOM_REFERENCE_DESGS_VW ■ WM_ENG_BOM_BILL_OF_MTLS_QRY_VW
wm_from_engbom_trg.sql	Create the following triggers to be used for writing to the WM_TRACKCHANGES table to indicate a new or updated document: <ul style="list-style-type: none"> ■ WM_ENG_BOM_BILL_OF_MTL_IUD_TRG ■ WM_ENG_MTL_ITEM_REV_IUD_TRG ■ WM_ENG_BOM_INV_COMPS_IUD_TRG ■ WM_ENG_BOM_SUB_COMPS_IUD_TRG ■ WM_ENG_BOM_REF_DESGS_IUD_TRG
wm_disable_from_engbom.sql	Disables the triggers installed by wm_from_engbom_trg.sql.

Script	Description
wm_enable_from_engbom.sql	Re-enables the triggers installed by wm_from_engbom_trg.sql.
wm_drop_from_engbom.sql	Uninstalls all components created by wm_install_from_engbom.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definition:

- `getEngineeringBOMTxn107SC.txp`

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow `sendEngineeringBOM` executes as follows:

- `specifyDefaultSettings` specifies the default parameter settings required for service execution. You should change these settings accordingly.
- `lockTxnCtrl` determines whether the `sendEngineeringBOM` service is ready for execution. That is, it determines whether there is no other instance of this service already in process.
 - If `lockTxnCtrl` returns `False`, it indicates that another instance of this service is already in process. The service exits and waits for next scheduled execution.
 - If `lockTxnCtrl` returns `True`, it indicates that the service is ready to execute, and that the Engineering BOM row in the control table is locked and updated so that the status is changed to `INPROCESS`. This prevents any other Engineering BOM service from executing.
- `getEngineeringBOMTxn` queries the Oracle Applications database for any Engineering BOM Transaction to be processed. The number of records returned depends on the parameter value specified in the above map.
- `processBizDoc` is the customizable step that sends the business document to the Trading Partner. This step also needs to be customized to receive a Success or an Error status of the document transfer. If there is an error in a particular business document transfer, the error information must be sent back to the calling service (`sendEngineeringBOM` in this case). The error information passed back should have the document identifiers. If a particular document is transferred successfully to the

Trading Partner, no information needs to be sent back to the calling program (sendEngineeringBOM).

- Based on the Debug Mode specified during execution, it is determined to either purge or update the records in the WM_TRACKCHANGES custom table.
 - If the Debug Mode is TRUE, then based on the purge criteria, the records in the WM_TRACKCHANGES table are updated and the PROCESSED_FLAG is set to Y. This ensures that the same set of records is not picked up during next polling interval. The **updateTrackChanges** service updates the PROCESSED_FLAG in the WM_TRACKCHANGES table to Y so that same information is not picked up again during next polling instance.
 - If the Debug Mode is FALSE, then based on the purge criteria, the records in the WM_TRACKCHANGES table are deleted; the **purgeTrackChanges** service purges the records from the WM_TRACKCHANGES table.
- If there is an error in transferring the business document, the **insertTransferERRInfo** service inserts a new record into the WM_TRACKCHANGES table so that same document can be picked up during the next polling interval.
- Any document that is created and deleted in between two successive polling operations will not be delivered. Such records are not selected by **getEngineeringBOMTxn** service; thus they are not updated as processed or purged in the WM_TRACKCHANGES table.
 - If the Debug Mode is TRUE, the **updateUnqualifiedRec** service recognizes such records and updates the corresponding PROCESSED_FLAG to Y.
 - If the Debug Mode is FALSE, the **purgeUnqualifiedRec** service recognizes such records and deletes them from the WM_TRACKCHANGES table.
- **unlockTxnCtrl** service releases the lock on the custom control table so that next polling instance of sendEngineeringBOM service can begin.
- If the document transfer is successful, the flow exits.

For more details on send transactions, see [“Overview of Send Service Transaction Processing” on page 38](#).

Business Document Structure

This service uses the following business document structure:

- 1.0. BOM_ENG_BILL_OF_MTLS
 - 1.1. MTL_ENG_ITEM_REVISIONS
 - 1.2. BOM_ENG_INVENTORY_COMPS
 - 1.2.1. BOM_ENG_SUB_COMPS
 - 1.2.2. BOM_ENG_REF_DESGS

1.0. BOM_ENG_BILL_OF_MTLS

Document Field	Oracle Applications Table/View Name	Column Name	Description
WEB_TRANSACTION_ID			
DOCUMENT_TYPE			ENGBOM will be populated in this field.
DOCUMENT_STATUS			UPDATE or INSERT will be populated in this field.
BILL_SEQUENCE_ID	BOM_BILL_OF_MATERIALS	BILL_SEQUENCE_ID	Unique identifier for Bill of Materials.
ITEM_NUMBER	MTL_SYSTEM_ITEMS_B_KFV	CONCATENATE_D_SEGMENTS	Segmented item number.
ORGANIZATION_NAME	HR_ALL_ORGANIZATION_UNITS	NAME	
COMMON_ITEM_NUMBER	MTL_SYSTEM_ITEMS_B_KFV	CONCATENATE_D_SEGMENTS	Concatenated segments for common bill item.
COMMON_ORG_NAME	HR_ALL_ORGANIZATION_UNITS	NAME	Organization name for common bill.
ALTERNATE_BOM_DESIGNATOR	BOM_BILL_OF_MATERIALS	ALTERNATE_BOM_DESIGNATOR	Alternate Bill of Materials designator code.
COMMON_ALT_BOM_DESIGNATOR	BOM_BILL_OF_MATERIALS	COMMON_ALT_BOM_DESIGNATOR	Common alternate Bill of Materials designator code.

Document Field	Oracle Applications Table/View Name	Column Name	Description
SPECIFIC_ASSEMBLY_COMMENT	BOM_BILL_OF_MATERIALS	SPECIFIC_ASSEMBLY_COMMENT	Specific Bill of Materials comment.
PENDING_FROM_ECN	BOM_BILL_OF_MATERIALS	PENDING_FROM_ECN	Change notice that created this Bill of Materials.
ASSEMBLY_TYPE	BOM_BILL_OF_MATERIALS	ASSEMBLY_TYPE	Engineering.
DEMAND_SOURCE_LINE			Not used for Bills of Material outbound transactions.
SET_ID			Not used for Bills of Material outbound transactions.
DEMAND_SOURCE_TYPE			Not used for Bills of Material outbound transactions.
DEMAND_SOURCE_HEADER_ID			Not used for Bills of Material outbound transactions.
NEXT_EXPLODE_DATE	BOM_BILL_OF_MATERIALS	NEXT_EXPLODE_DATE	Next date when pre-explosion will be refreshed.
UNIT_OF_MEASURE	MTL_UNITS_OF_MEASURE_TL	UNITS_OF_MEASURE	
REVISION			Not used for Bills of Material outbound transactions.

1.1. MTL_ENG_ITEM_REVISIONS

Document Field	Oracle Applications Table/View Name	Column Name	Description
BILL_SEQUENCE_ID	BOM_INVENTORY_COMPONENTS	BILL_SEQUENCE_ID	Unique bill identifier.
ITEM_NUMBER	MTL_SYSTEM_ITEMS_B_KFV	CONCATENATED_SEGMENTS	Segmented item number.
ORGANIZATION_NAME	HR_ALL_ORGANIZATION_UNITS	NAME	
REVISION	MTL_ITEM_REVISIONS	REVISION	Item revision code.

Document Field	Oracle Applications Table/View Name	Column Name	Description
CHANGE_NOTICE	MTL_ITEM_REVISIONS	CHANGE_NOTICE	Engineering change order number.
ECN_INITIATION_DATE	MTL_ITEM_REVISIONS	ECN_INITIATION_DATE	Engineering change order initiation date.
IMPLEMENTATION_DATE	MTL_ITEM_REVISIONS	IMPLEMENTATION_DATE	Engineering change order implementation date.
EFFECTIVITY_DATE	MTL_ITEM_REVISIONS	EFFECTIVITY_DATE	Revision effective date.
REVISED_ITEM_SEQUENCE_ID	MTL_ITEM_REVISIONS	REVISED_ITEM_SEQUENCE_ID	
TRANSACTION_TYPE			Not used for Bills of Material outbound transactions.

1.2. BOM_ENG_INVENTORY_COMPS

Document Field	Oracle Applications Table/View Name	Column Name	Description
COMPONENT_SEQUENCE_ID	BOM_INVENTORY_COMPONENTS	COMPONENT_SEQUENCE_ID	
BILL_SEQUENCE_ID	BOM_INVENTORY_COMPONENTS	BILL_SEQUENCE_ID	Unique bill identifier.
ASSEMBLY_ITEM_NUMBER	MTL_SYSTEM_ITEMS_B_KFV	CONCATENATED_SEGMENTS	Inventory item number of manufactured assembly.
ALTERNATE_BOM_DESIGNATOR	BOM_BILL_OF_MATERIALS	ALTERNATE_BOM_DESIGNATOR	Alternate Bill of Materials designator code.
ORGANIZATION_NAME	HR_ALL_ORGANIZATION_UNITS	NAME	
COMPONENT_ITEM_NUMBER	MTL_SYSTEM_ITEMS_B_KFV	CONCATENATED_SEGMENTS	
OPERATION_SEQ_NUM	BOM_INVENTORY_COMPONENTS	OPERATION_SEQ_NUM	Operation sequence number.

Document Field	Oracle Applications Table/View Name	Column Name	Description
EFFECTIVITY_DATE	BOM_INVENTORY_COMPONENTS	EFFECTIVITY_DATE	Effective date.
LOCATION_NAME	MTL_ITEM_LOCATIONS	DESCRIPTION	Supply locator name.
SUBSTITUTE_COMPONENT_NUMBER			Not used for Bills of Material outbound transactions.
ITEM_NUM	BOM_INVENTORY_COMPONENTS	ITEM_NUM	Item sequence within Bill of Materials structure.
COMPONENT_QUANTITY	BOM_INVENTORY_COMPONENTS	COMPONENT_QUANTITY	Quantity of component in Bill of Materials.
COMPONENT_YIELD_FACTOR	BOM_INVENTORY_COMPONENTS	COMPONENT_YIELD_FACTOR	Factor used to multiply component quantity with to obtain required component quantity.
COMPONENT_REMARKS	BOM_INVENTORY_COMPONENTS	COMPONENT_REMARKS	
CHANGE_NOTICE	BOM_INVENTORY_COMPONENTS	CHANGE_NOTICE	Engineering change order number.
IMPLEMENTATION_DATE	BOM_INVENTORY_COMPONENTS	IMPLEMENTATION_DATE	Date on which engineering change order was implemented.
DISABLE_DATE	BOM_INVENTORY_COMPONENTS	DISABLE_DATE	
PLANNING_FACTOR	BOM_INVENTORY_COMPONENTS	PLANNING_FACTOR	Factor used to multiply component quantity with to obtain planning quantity.
QUANTITY_RELATED	BOM_INVENTORY_COMPONENTS	QUANTITY_RELATED	Identifier to indicate if this component has quantity related reference designators.

Document Field	Oracle Applications Table/View Name	Column Name	Description
SO_BASIS	BOM_INVENTORY_COMPONENTS	SO_BASIS	Quantity basis used by Oracle Order Management to determine how many units of component to put on an order.
OPTIONAL	BOM_INVENTORY_COMPONENTS	OPTIONAL	Indicates whether component is optional in bill.
MUTUALLY_EXCLUSIVE_OPTIONS	BOM_INVENTORY_COMPONENTS	MUTUALLY_EXCLUSIVE_OPTIONS	Indicates whether one or more children of the component can be picked when taking an order.
INCLUDE_IN_COST_ROLLUP	BOM_INVENTORY_COMPONENTS	INCLUDE_IN_COST_ROLLUP	Indicates whether the component is to be used when rolling up costs.
CHECK_ATP	BOM_INVENTORY_COMPONENTS	CHECK_ATP	Indicates whether ATP check is required.
SHIPPING_ALLOWED	BOM_INVENTORY_COMPONENTS	SHIPPING_ALLOWED	Indicates whether component is allowed to ship.
REQUIRED_TO_SHIP	BOM_INVENTORY_COMPONENTS	REQUIRED_TO_SHIP	Indicates whether component is required to ship.
REQUIRED_FOR_REVENUE	BOM_INVENTORY_COMPONENTS	REQUIRED_FOR_REVENUE	Indicates whether component is required for revenue.
INCLUDE_ON_SHIP_DOCS	BOM_INVENTORY_COMPONENTS	INCLUDE_ON_SHIP_DOCS	Indicates whether component is displayed on shipping documents.
INCLUDE_ON_BILL_DOCS	BOM_INVENTORY_COMPONENTS	INCLUDE_ON_BILL_DOCS	Indicates whether component is displayed on billing documents.
LOW_QUANTITY	BOM_INVENTORY_COMPONENTS	LOW_QUANTITY	Minimum quantity allowed on an order.

Document Field	Oracle Applications Table/View Name	Column Name	Description
HIGH_QUANTITY	BOM_INVENTORY_COMPONENTS	HIGH_QUANTITY	Maximum quantity allowed on an order
ACD_TYPE	BOM_INVENTORY_COMPONENTS	ACD_TYPE	Add, change, or disable code for component on an engineering change order.
REVISED_ITEM_NUMBER	MTL_SYSTEM_ITEMS_KFV	CONCATENATED_SEGMENTS	
REVISION	MTL_ITEM_REVISIONS	REVISION	
WIP_SUPPLY_TYPE	BOM_INVENTORY_COMPONENTS	WIP_SUPPLY_TYPE	
SUPPLY_SUBINVENTORY	BOM_INVENTORY_COMPONENTS	SUPPLY_SUBINVENTORY	
REFERENCE_DESIGNATOR			Not used for Engineering Bills of Material outbound transactions.
BOM_ITEM_TYPE	BOM_INVENTORY_COMPONENTS	BOM_ITEM_TYPE	Type of item (de-normalized from BOM Item Type in MTL_SYSTEM_ITEMS).
OPERATION_LEAD_TIME_PERCENT	BOM_INVENTORY_COMPONENTS	OPERATION_LEAD_TIME_PERCENT	Operation offset from first operation in routing.
UNIT_OF_MEASURE	MTL_UNITS_OF_MEASURE_TL	UNITS_OF_MEASURE	

1.2.1. BOM_ENG_SUB_COMPS

Document Field	Oracle Applications Table/View Name	Column Name	Description
BILL_SEQUENCE_ID	BOM_INVENTORY_COMPONENTS	BILL_SEQUENCE_ID	Unique bill identifier.
SUBSTITUTE_COMP_NUMBER	MTL_SYSTEM_ITEMS_B_KFV	CONCATENATE_D_SEGMENTS	Substitute component number.
SUBSTITUTE_ITEM_QUANTITY	BOM_SUBSTITUTE_COMPONENTS	SUBSTITUTE_ITEM_QUANTITY	Substitute component quantity.
ACD_TYPE	BOM_SUBSTITUTE_COMPONENTS	ACD_TYPE	Type to indicate add or delete on an engineering change order.
CHANGE_NOTICE	BOM_SUBSTITUTE_COMPONENTS	CHANGE_NOTICE	Engineering change order number.
ORGANIZATION_NAME	HR_ALL_ORGANIZATION_UNITS	NAME	
ASSEMBLY_ITEM_NUMBER	MTL_SYSTEM_ITEMS_B_KFV	CONCATENATE_D_SEGMENTS	Inventory item number of manufactured assembly.
ALTERNATE_BOM_DESIGNATOR	BOM_BILL_OF_MATERIALS	ALTERNATE_BOM_DESIGNATOR	Alternate Engineering Bill of Materials designator code.
COMPONENT_ITEM_NUMBER	MTL_SYSTEM_ITEMS_B_KFV	CONCATENATE_D_SEGMENTS	Inventory item number of component.
OPERATION_SEQ_NUM	BOM_INVENTORY_COMPONENTS	OPERATION_SEQ_NUM	Manufacturing operation sequence number.
EFFECTIVITY_DATE	BOM_INVENTORY_COMPONENTS	EFFECTIVITY_DATE	Effective date.
UNIT_OF_MEASURE	MTL_UNITS_OF_MEASURE_TL	UNITS_OF_MEASURE	

1.2.2. BOM_ENG_REF_DESGS

Document Field	Oracle Applications Table/View Name	Column Name	Description
BILL_SEQUENCE_ID	BOM_INVENTORY_COMPONENTS	BILL_SEQUENCE_ID	Unique bill identifier.
COMPONENT_REFERENCE_DESIGNATOR	BOM_REFERENCE_DESIGNATORS	COMPONENT_REFERENCE_DESIGNATOR	
ORGANIZATION_NAME	HR_ALL_ORGANIZATION_UNITS	NAME	
ASSEMBLY_ITEM_NUMBER	MTL_SYSTEM_ITEMS_B_KFV	CONCATENATED_SEGMENTS	
ALTERNATE_BOM_DESIGNATOR	BOM_BILL_OF_MATERIALS	ALTERNATE_BOM_DESIGNATOR	Alternate Bill of Materials identifier.
COMPONENT_ITEM_NUMBER	MTL_SYSTEM_ITEMS_B_KFV	CONCATENATED_SEGMENTS	
OPERATION_SEQ_NUM	BOM_INVENTORY_COMPONENTS	OPERATION_SEQ_NUM	Manufacturing operation sequence number.
EFFECTIVITY_DATE	BOM_INVENTORY_COMPONENTS	EFFECTIVITY_DATE	Date on which functionality will be enabled.
REF_DESIGNATOR_COMMENT	BOM_REFERENCE_DESIGNATORS	REF_DESIGNATOR_COMMENT	
CHANGE_NOTICE	BOM_REFERENCE_DESIGNATORS	CHANGE_NOTICE	Engineering change order number.
ACD_TYPE	BOM_REFERENCE_DESIGNATORS	ACD_TYPE	Add or delete code from an engineering change order.

Send Item Service

The name of this service is:

WmOAMFG107SC.inventory107SC.fromOA.item:sendItem

This service sends item information to the webMethods Integration Server.

Items defined in Oracle Applications cannot be deleted. A business document with document status of INSERT or UPDATE will be created. The document status of DELETE does not apply in this case.

Database Scripts

This service uses the following database scripts:

Script	Description
wm_install_from_invitem.sql	Runs the scripts listed below, except the uninstall script.
wm_from_invitem_vw.sql	Creates the following required view components for item outbound transactions: <ul style="list-style-type: none"> ■ WM_INV_ITEMS_VW ■ WM_INV_ITEM_CATEGORIES_VW ■ WM_INV_ITEM_REVISIONS_VW ■ WM_INV_ITEMS_QRY_VW
wm_from_invitem_trg.sql	Creates the following required trigger components for item outbound transactions: <ul style="list-style-type: none"> ■ WM_MTL_SYSTEM_ITEMS_IU_TRG ■ WM_MTL_ITEM_CATEGORIES_IUD_TRG ■ WM_MTL_ITEM_REVISIONS_IU_TRG
wm_disable_from_invitem.sql	Disables the triggers installed by wm_from_invitem_trg.sql.
wm_enable_from_invitem.sql	Re-enables the triggers installed by wm_from_invitem_trg.sql.
wm_drop_from_invitem.sql	Uninstalls all components created by wm_install_from_invitem.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definition:

- `getItemTxn107SC.txp`

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow `sendItem` executes as follows:

- `specifyDefaultSettings` specifies the default parameter settings required for service execution. You should change these settings accordingly.
- `lockTxnCtrl` determines whether the `sendItem` service is ready for execution. That is, it determines whether there is no other instance of this service already in process.
 - If `lockTxnCtrl` returns `False`, it indicates that another instance of this service is already in process. The service exits and waits for next scheduled execution.
 - If `lockTxnCtrl` returns `True`, it indicates that the service is ready to execute, and that the item row in the control table is locked and updated so that the status is changed to be `INPROCESS`. This prevents any other item service from executing.
- `getItemTxn` queries the Oracle Applications database for any Item Transaction to be processed. The number of records returned depends on the parameter value specified in the above map.
- `processBizDoc` is the customizable step that sends the business document to the Trading Partner. This step needs to be customized to receive a Success or an Error status of the document transfer. If there is an error in a particular business document transfer, the error information must be sent back to the calling service (`sendItem` in this case). The error information passed back should have the document identifiers. If a particular document is transferred successfully to the Trading Partner, no information needs to be sent back to the calling program (`sendItem`).
- Based on the Debug Mode specified during execution, it is determined to either purge or update the records in the `WM_TRACKCHANGES` custom table.
 - If the Debug Mode is `TRUE`, then based on the purge criteria the records in the `WM_TRACKCHANGES` table are updated and the `PROCESSED_FLAG` is set to `Y`. This ensures that the same set of records is not picked up during next polling interval. The `updateTrackChanges` service updates the `PROCESSED_FLAG` in the `WM_TRACKCHANGES` table to `Y` so that same information is not picked up again during next polling instance.

- If the Debug Mode is FALSE, then based on the purge criteria the records in the WM_TRACKCHANGES table are deleted. The `purgeTrackChanges` service purges the records from the WM_TRACKCHANGES table.
- If an error occurs in transferring the business document, the `insertTransferERRInfo` service inserts a new record into the WM_TRACKCHANGES table so that same document can be picked up during the next polling interval.
- `unlockTxnCtrl` releases the lock on the custom control table so that next polling instance of `sendItem` service can begin.
- If the document transfer is successful, the flow exits.

For more details on send transactions, see [“Overview of Send Service Transaction Processing” on page 38](#).

Business Document Structure

This service uses the following business document structure:

- 1.0. MTL_SYSTEM_ITEM
 - 1.1. MTL_ITEM_REVISIONS
 - 1.1.1. MTL_ITEM_CATEGORIES

1.0. MTL_SYSTEM_ITEM

Document Field	Oracle Applications Table/View Name	Column Name	Description
WEB_TRANSACTION_ID	None		
DOCUMENT_TYPE			ITEM will be populated in this field.
DOCUMENT_STATUS			UPDATE or INSERT will be populated in this field.
INV_ITEM_ORG_ID	None		Concatenated String INVENTORY_ITEM_ID '- ' ORGANIZATION_ID. Applicable for <code>queryItem</code> service.
ITEM_NUMBER	MTL_SYSTEM_ITEMS_KFV	CONCATENATE_D_SEGMENTS	Segmented item number.

Document Field	Oracle Applications Table/View Name	Column Name	Description
ORGANIZATION_NAME	ORGANIZATION_ID	ORGANIZATION_NAME	Derives ORGANIZATION_NAME from ORG_ORGANIZATION_DEFINITIONS using ORGANIZATION_ID.
SUMMARY_FLAG	MTL_SYSTEM_ITEMS	SUMMARY_FLAG	Flexfield summary flag.
TAX_CODE	MTL_SYSTEM_ITEMS	TAX_CODE	
ENABLED_FLAG	MTL_SYSTEM_ITEMS	ENABLED_FLAG	
START_DATE_ACTIVE	MTL_SYSTEM_ITEMS	START_DATE_ACTIVE	Flex field segment start date.
END_DATE_ACTIVE	MTL_SYSTEM_ITEMS	END_DATE_ACTIVE	Flex field segment end date.
DESCRIPTION	MTL_SYSTEM_ITEMS	DESCRIPTION	
BUYER_FULL_NAME	PER_ALL_PEOPLE_F	FULL_NAME	Derives AGENT_ID from PO_AGENTS, PER_ALL_PEOPLE_F, where PO_AGENTS.AGENT_ID = PER_ALL_PEOPLE_F.PERSON_ID.
ACCOUNTING_RULE_NAME	RA_RULES	RULE_NAME	Derives the RULE_NAME from RA_RULES for the RULE_ID.
INVOICING_RULE_NAME	RA_RULES	RULE_NAME	Derives the RULE_NAME from RA_RULES for the RULE_ID
PURCHASING_ITEM_FLAG	MTL_SYSTEM_ITEMS	PURCHASING_ITEM_FLAG	Indicates whether the item can be purchased.
SHIPPABLE_ITEM_FLAG	MTL_SYSTEM_ITEMS	SHIPPABLE_ITEM_FLAG	Indicates whether the item can be shipped.
CUSTOMER_ORDER_FLAG	MTL_SYSTEM_ITEMS	CUSTOMER_ORDER_FLAG	

Document Field	Oracle Applications Table/View Name	Column Name	Description
INTERNAL_ORDER_FLAG	MTL_SYSTEM_ITEMS	INTERNAL_ORDER_FLAG	
SERVICE_ITEM_FLAG	MTL_SYSTEM_ITEMS	SERVICE_ITEM_FLAG	
INVENTORY_ITEM_FLAG	MTL_SYSTEM_ITEMS	INVENTORY_ITEM_FLAG	
ENG_ITEM_FLAG	MTL_SYSTEM_ITEMS	ENG_ITEM_FLAG	
INVENTORY_ASSET_FLAG	MTL_SYSTEM_ITEMS	INVENTORY_ASSET_FLAG	
PURCHASING_ENABLED_FLAG	MTL_SYSTEM_ITEMS	PURCHASING_ENABLED_FLAG	Indicates whether the item can be purchased.
CUSTOMER_ORDER_ENABLED_FLAG	MTL_SYSTEM_ITEMS	CUSTOMER_ORDER_ENABLED_FLAG	Indicates whether the item is customer orderable.
INTERNAL_ORDER_ENABLED_FLAG	MTL_SYSTEM_ITEMS	INTERNAL_ORDER_ENABLED_FLAG	Indicates whether the item is internally orderable.
SO_TRANSACTIONS_FLAG	MTL_SYSTEM_ITEMS	SO_TRANSACTION_S_FLAG	Sales order transaction flag.
MTL_TRANSACTIONS_ENABLED_FLAG	MTL_SYSTEM_ITEMS	MTL_TRANSACTION_S_ENABLED_FLAG	Indicates whether the item can be processed.
STOCK_ENABLED_FLAG	MTL_SYSTEM_ITEMS	STOCK_ENABLED_FLAG	Indicates whether the item can be stocked.
BOM_ENABLED_FLAG	MTL_SYSTEM_ITEMS	BOM_ENABLED_FLAG	Indicates whether the item can appear on a Bill of Materials.
BUILD_IN_WIP_FLAG	MTL_SYSTEM_ITEMS	BUILD_IN_WIP_FLAG	Indicates whether the item can be built in WIP.

Document Field	Oracle Applications Table/View Name	Column Name	Description
REVISION_QTY_CONTROL	MFG_LOOKUPS	MEANING	Derives MEANING from MFG_LOOKUPS, where LOOKUP_TYPE is MTL_ENG_QUANTITY for LOOKUP_CODE.
CATALOG_GROUP_DESCRIPTION	MTL_ITEM_CATALOG_GROUPS	ITEM_CATALOG_GROUP_NAME	Derives ITEM_CATALOG_GROUP_NAME from MTL_ITEM_CATALOG_GROUPS for ITEM_CATALOG_GROUP_ID.
CATALOG_STATUS_FLAG	MTL_SYSTEM_ITEMS	CATALOG_STATUS_FLAG	Indicates whether the item is catalog complete.
RETURNABLE_FLAG	MTL_SYSTEM_ITEMS	RETURNABLE_FLAG	Indicates whether the item can be returned.
DEFAULT_SHIP_ORGANIZATION_NAME	ORG_ORGANIZATION_DEFINITIONS	ORGANIZATION_NAME	Derives ORGANIZATION_NAME from ORG_ORGANIZATION_DEFINITIONS.
COLLATERAL_FLAG	MTL_SYSTEM_ITEMS	COLLATERAL_FLAG	Indicates whether the item is a collateral item.
TAXABLE_FLAG	MTL_SYSTEM_ITEMS	TAXABLE_FLAG	Indicates whether the item is taxable.
QTY_RCV_EXCEPTION_CODE	MTL_SYSTEM_ITEMS	QTY_RCV_EXCEPTION_CODE	Use: NONE for no receiving control enforced, REJECT for prevent receipt of goods or services, or WARNING for display warning message.
ALLOW_ITEM_DESC_UPDATE_FLAG	MTL_SYSTEM_ITEMS	ALLOW_ITEM_DESC_UPDATE_FLAG	Indicates whether to allow item description updates on inventory item lines.
INSPECTION_REQUIRED_FLAG	MTL_SYSTEM_ITEMS	INSPECTION_REQUIRED_FLAG	

Document Field	Oracle Applications Table/View Name	Column Name	Description
RECEIPT_REQUIRED_FLAG	MTL_SYSTEM_ITEMS	RECEIPT_REQUIRED_FLAG	Indicates whether a supplier receipt is required.
MARKET_PRICE	MTL_SYSTEM_ITEMS	MARKET_PRICE	
HAZARD_CLASS	PO_HAZARD_CLASSES	HAZARD_CLASS_NAME	Derives the HAZARD_CLASS_NAME from PO_HAZARD_CLASSES.
RFQ_REQUIRED_FLAG	MTL_SYSTEM_ITEMS	RFQ_REQUIRED_FLAG	Indicates whether an RFQ is required.
QTY_RCV_TOLERANCE	MTL_SYSTEM_ITEMS	QTY_RCV_TOLERANCE	Maximum permissible over receipt percentage.
LIST_PRICE_PER_UNIT	MTL_SYSTEM_ITEMS	LIST_PRICE_PER_UNIT	
UN_NUMBER_ID	MTL_SYSTEM_ITEMS	UN_NUMBER_ID	Purchasing UN number.
PRICE_TOLERANCE_PERCENT	MTL_SYSTEM_ITEMS	PRICE_TOLERANCE_PERCENT	
ASSET_CATEGORY	FA_ASSET_CATEGORIES_B_KFV	CONCATENATE_D_SEGMENT	Derives the CONCATENATED_SEGMENTS from FA_CATEGORIES_KFV for the ASSET_CATEGORY_ID.
ROUNDING_FACTOR	MTL_SYSTEM_ITEMS	ROUNDING_FACTOR	Determines order quantity.
UNIT_OF_ISSUE	MTL_SYSTEM_ITEMS	UNIT_OF_ISSUE	
ENFORCE_SHIP_TO_LOCATION_CODE	MTL_SYSTEM_ITEMS	ENFORCE_SHIP_TO_LOCATION_CODE	Requires receipt location to match ship to location.
ALLOW_SUBSTITUTE_RECEIPTS_FLAG	MTL_SYSTEM_ITEMS	ALLOW_SUBSTITUTE_RECEIPTS_FLAG	Indicates whether substitute receipts are allowed.

Document Field	Oracle Applications Table/View Name	Column Name	Description
ALLOW_UNORDERED_RECEIPTS_FLAG	MTL_SYSTEM_ITEMS	ALLOW_UNORDERED_RECEIPTS_FLAG	Indicates whether to allow unordered receipts.
ALLOW_EXPRESS_DELIVERY_FLAG	MTL_SYSTEM_ITEMS	ALLOW_EXPRESS_RECEIPTS_FLAG	Indicates whether to allow express delivery.
DAYS_EARLY_RECEIPT_ALLOWED	MTL_SYSTEM_ITEMS	DAYS_EARLY_RECEIPT_ALLOWED	
DAYS_LATE_RECEIPT_ALLOWED	MTL_SYSTEM_ITEMS	DAYS_LATE_RECEIPT_ALLOWED	
RECEIPT_DAYS_EXCEPTION_CODE	MTL_SYSTEM_ITEMS	RECEIPT_DAYS_EXCEPTION_CODE	Use: NONE for no receiving control enforced, EJECT for prevent receipt of goods or services, or WARNING for display warning message.
RECEIVING_ROUTING_NAME	RCV_ROUTING_HEADERS	ROUTING_NAME	Derives ROUTING_NAME from RCV_ROUTING_HEADERS for RCV_ROUTING_ID.
INVOICE_CLOSE_TOLERANCE	MTL_SYSTEM_ITEMS	INVOICE_CLOSE_TOLERANCE	
RECEIVE_CLOSE_TOLERANCE	MTL_SYSTEM_ITEMS	RECEIVE_CLOSE_TOLERANCE	
AUTO_LOT_ALPHA_PREFIX	MTL_SYSTEM_ITEMS	AUTO_LOT_ALPHA_PREFIX	Lot prefix for lot number controlled item.
START_AUTO_LOT_NUMBER	MTL_SYSTEM_ITEMS	START_AUTO_LOT_NUMBER	Starting lot number.

Document Field	Oracle Applications Table/View Name	Column Name	Description
LOT_CONTROL	MFG_LOOKUPS	MEANING	Derives MEANING from MFG_LOOKUPS, where LOOKUP_TYPE = MTL_LOT_CONTROL for the LOT_CONTROL_CODE. Use: L for the meaning column, 1 for No lot control, 2 for Full lot control.
SHELF_LIFE	MFG_LOOKUPS	MEANING	Derives MEANING from MFG_LOOKUPS, where LOOKUP_TYPE = MTL_SHELF_LIFE for the SHELF_LIFE_CODE column. Use: 1 for No shelf life control, 2 for Item shelf life days, 4 for User-defined expiration date.
SHELF_LIFE_DAYS	MTL_SYSTEM_ITEMS	SHELF_LIFE_DAYS	
SERIAL_NUMBER_CONTROL	MFG_LOOKUPS	MEANING	Derives MEANING from MFG_LOOKUPS, where LOOKUP_TYPE = MTL_SERIAL_NUMBER for the SERIAL_NUMBER_CONTROL_CODE column. Use: 1 for No serial number control, 2 for Predefined serial numbers, 5 for Dynamic entry at inventory receipt, 6 for Dynamic entry at sales order issue.
START_AUTO_SERIAL_NUMBER	MTL_SYSTEM_ITEMS	START_AUTO_SERIAL_NUMBER	
AUTO_SERIAL_ALPHA_PREFIX	MTL_SYSTEM_ITEMS	AUTO_SERIAL_ALPHA_PREFIX	Serial number prefix.

Document Field	Oracle Applications Table/View Name	Column Name	Description
SOURCE_TYPE	MFG_LOOKUPS	MEANING	For either Inventory or Supplier. Derives MEANING from MFG_LOOKUPS, where LOOKUP_TYPE = MTL_SOURCE_TYPES for the SOURCE_TYPE.
SOURCE_ORGANIZATION_NAME	ORG_ORGANIZATION_DEFINITIONS	ORGANIZATION_NAME	Derives ORGANIZATION_NAME from ORG_ORGANIZATION_DEFINITIONS for the SOURCE_ORGANIZATION_ID.
SOURCE_SUBINVENTORY	MTL_SYSTEM_ITEMS	SOURCE_SUBINVENTORY	
EXPENSE_ACCOUNT	GL_CODE_COMBINATIONS_KFV	CONCATENATED_SEGMENTS	Derived from GL_CODE_COMBINATIONS_KFV for the EXPENSE_ACCOUNT.
ENCUMBRANCE_ACCOUNT	GL_CODE_COMBINATIONS_KFV	CONCATENATED_SEGMENTS	Derived from GL_CODE_COMBINATIONS_KFV for the ENCUMBRANCE_ACCOUNT.
RESTRICT_SUBINVENTORIES	MFG_LOOKUPS	MEANING	Derived from MFG_LOOKUPS, where LOOKUP_TYPE = MTL_SUBINVENTORY_RESTRICTIONS for the RESTRICT_SUBINVENTORIES_CODE column. Use: 1 for Subinventories restricted to predefined list, 2 for Subinventories not restricted to predefined list.
UNIT_WEIGHT	MTL_SYSTEM_ITEMS	UNIT_WEIGHT	
WEIGHT_UOM_CODE	MTL_SYSTEM_ITEMS	WEIGHT_UOM_CODE	

Document Field	Oracle Applications Table/View Name	Column Name	Description
VOLUME_UOM_CODE	MTL_SYSTEM_ITEMS	VOLUME_UOM_CODE	
UNIT_VOLUME	MTL_SYSTEM_ITEMS	UNIT_VOLUME	
RESTRICT_LOCATORS	MFG_LOOKUPS	MEANING	Derived from MFG_LOOKUPS, where LOOKUP_TYPE = MTL_LOCATOR_RESTRICTIONS for the RESTRICT_LOCATORS_CODE. Use: 1 for Locators restricted to predefined list, 2 for Locators not restricted to predefined list.
LOCATION_CONTROL	MFG_LOOKUPS	MEANING	Derived from MFG_LOOKUPS, where LOOKUP_TYPE = MTL_LOCATION_CONTROL for the LOCATION_CONTROL_CODE column. Use: 1 for No locator control, 2 for Pre-specified locator control, 3 for Dynamic entry locator control, 4 for Locator control determined at subinventory level, 5 for Locator control determined at item level.
SHRINKAGE_RATE	MTL_SYSTEM_ITEMS	SHRINKAGE_RATE	Planned shrinkage rate.
ACCEPTABLE_EARLY_DAYS	MTL_SYSTEM_ITEMS	ACCEPTABLE_EARLY_DAYS	Days an order may be early before rescheduling is recommended.

Document Field	Oracle Applications Table/View Name	Column Name	Description
PLANNING_TIME_FENCE	MFG_LOOKUPS	MEANING	Derived from MFG_LOOKUPS, where LOOKUP_TYPE = MTL_TIME_FENCE for the PLANNING_TIME_FENCE_CODE column. Use: 1 for Cumulative total lead time, 2 for Cumulative manufacturing lead time, 3 for Total lead time, 4 for User-defined time fence.
DEMAND_TIME_FENCE	MFG_LOOKUPS	MEANING	Derived from MFG_LOOKUPS, where LOOKUP_TYPE = MTL_TIME_FENCE for the DEMAND_TIME_FENCE_CODE column. Use: 1 for Cumulative total lead time, 2 for Cumulative manufacturing lead time, 3 for Total lead time, 4 for user-defined time fence.
LEAD_TIME_LOT_SIZE	MTL_SYSTEM_ITEMS	LEAD_TIME_LOT_SIZE	
STD_LOT_SIZE	MTL_SYSTEM_ITEMS	STD_LOT_SIZE	
CUM_MANUFACTURING_LEAD_TIME	MTL_SYSTEM_ITEMS	CUM_MANUFACTURING_LEAD_TIME	Cumulative lead time.
OVERRUN_PERCENTAGE	MTL_SYSTEM_ITEMS	OVERRUN_PERCENTAGE	MRP repetitive overrun rate.
MRP_CALCULATE_ATP_FLAG	MTL_SYSTEM_ITEMS	MRP_CALCULATE_ATP_FLAG	
ACCEPTABLE_RATE_INCREASE	MTL_SYSTEM_ITEMS	ACCEPTABLE_RATE_INCREASE	MRP repetitive acceptable rate increase.

Document Field	Oracle Applications Table/View Name	Column Name	Description
ACCEPTABLE_RATE_DECREASE	MTL_SYSTEM_ITEMS	ACCEPTABLE_RATE_DECREASE	MRP repetitive acceptable rate decrease.
CUMULATIVE_TOTAL_LEAD_TIME	MTL_SYSTEM_ITEMS	CUMULATIVE_TOTAL_LEAD_TIME	
PLANNING_TIME_FENCE_DAYS	MTL_SYSTEM_ITEMS	PLANNING_TIME_FENCE_DAYS	
DEMAND_TIME_FENCE_DAYS	MTL_SYSTEM_ITEMS	DEMAND_TIME_FENCE_DAYS	
END_ASSEMBLY_PEGGING_FLAG	MTL_SYSTEM_ITEMS	END_ASSEMBLY_PEGGING_FLAG	
REPETITIVE_PLANNING_FLAG	MTL_SYSTEM_ITEMS	REPETITIVE_PLANNING_FLAG	
PLANNING_EXCEPTION_SET	MTL_SYSTEM_ITEMS	PLANNING_EXCEPTION_SET	Exception control set.
BOM_ITEM_TYPE	MFG_LOOKUPS	MEANING	Derived from MFG_LOOKUPS, where LOOKUP_TYPE is BOM_ITEM_TYPE for the BOM_ITEM_TYPE column.
PICK_COMPONENTS_FLAG	MTL_SYSTEM_ITEMS	PICK_COMPONENTS_FLAG	
REPLENISH_TO_ORDER_FLAG	MTL_SYSTEM_ITEMS	REPLENISH_TO_ORDER_FLAG	
BASE_ITEM	MTL_SYSTEM_ITEMS_KFV	CONCATENATED_SEGMENTS	Derived from MTL_SYSTEM_ITEMS_KFV for the BASE_ITEM_ID and Organization.
ATP_COMPONENTS_FLAG	MTL_SYSTEM_ITEMS	ATP_COMPONENTS_FLAG	

Document Field	Oracle Applications Table/View Name	Column Name	Description
ATP_FLAG	MTL_SYSTEM_ITEMS	ATP_FLAG	Indicates whether ATP must be checked when ordering the item.
FIXED_LEAD_TIME	MTL_SYSTEM_ITEMS	FIXED_LEAD_TIME	Fixed portion of the assembly lead time.
VARIABLE_LEAD_TIME	MTL_SYSTEM_ITEMS	VARIABLE_LEAD_TIME	
WIP_SUPPLY_LOCATOR	MTL_ITEM_LOCATIONS_KFV	CONCATENATE_D_SEGMENTS	Derived from MTL_ITEM_LOCATIONS_KFV for the WIP_SUPPLY_LOCATOR.
WIP_SUPPLY_TYPE	MFG_LOOKUPS	MEANING	Derived from MFG_LOOKUPS for the LOOKUP_TYPE WIP_SUPPLY.
WIP_SUPPLY_SUBINVENTORY	MTL_SYSTEM_ITEMS	WIP_SUPPLY_SUBINVENTOR Y	
PRIMARY_UOM_CODE	MTL_SYSTEM_ITEMS	PRIMARY_UOM_CODE	Primary unit of measure code.
ALLOWED_UNITS_LOOKUP	MFG_LOOKUP	MEANING	Derived from MFG_LOOKUPS, where LOOKUP_TYPE = MTL_CONVERSION_TYPE for the ALLOWED_UNITS_LOOKUP_CODE column.
COST_OF_SALES_ACCOUNT	GL_CODE_COMBINATIONS_KFV	CONCATENATE_D_SEGMENTS	Derives CONCATENATED_SEGMENTS from GL_CODE_COMBINATIONS_KFV for the COST_OF_SALES_ACCOUNT segment.
SALES_ACCOUNT	GL_CODE_COMBINATIONS_KFV	CONCATENATE_D_SEGMENTS	Derives CONCATENATED_SEGMENTS from GL_CODE_COMBINATIONS_KFV for the SALES_ACCOUNT.
DEFAULT_INCLUDE_IN_ROLLUP_FLAG	MTL_SYSTEM_ITEMS	DEFAULT_INCLUDE_IN_ROLLUP_FLAG	Indicates whether to include default value in cost rollup.

Document Field	Oracle Applications Table/View Name	Column Name	Description
INVENTORY_ITEM_STATUS_CODE	MTL_SYSTEM_ITEMS	INVENTORY_ITEM_STATUS_CODE	
INVENTORY_PLANNING	MFG_LOOKUPS	MEANING	Derived from MFG_LOOKUPS, where LOOKUP_TYPE = MTL_MATERIAL_PLANNING.
PLANNER_CODE	MTL_SYSTEM_ITEMS	PLANNER_CODE	
PLANNING_MAKE_BUY	MTL_SYSTEM_ITEMS	PLANNING_MAKE_BUY_CODE	Derives LOOKUP_CODE from MFG_LOOKUPS where LOOKUP_TYPE = MTL_PLANNING_MAKE_BUY. Indicates whether the item is planned as manufactured or purchased.
FIXED_LOT_MULTIPLIER	MTL_SYSTEM_ITEMS	FIXED_LOT_MULTIPLIER	
ROUNDING_CONTROL_TYPE	MFG_LOOKUPS	MEANING	Derived from MFG_LOOKUPS, where LOOKUP_TYPE = MTL_ROUNDING for the ROUNDING_CONTROL_TYPE column.
CARRYING_COST	MTL_SYSTEM_ITEMS	CARRYING_COST	Annual carrying cost.
POSTPROCESSING_LEAD_TIME	MTL_SYSTEM_ITEMS	POSTPROCESSING_LEAD_TIME	
PREPROCESSING_LEAD_TIME	MTL_SYSTEM_ITEMS	PREPROCESSING_LEAD_TIME	
FULL_LEAD_TIME	MTL_SYSTEM_ITEMS	FULL_LEAD_TIME	
ORDER_COST	MTL_SYSTEM_ITEMS	ORDER_COST	

Document Field	Oracle Applications Table/View Name	Column Name	Description
MRP_SAFETY_STOCK_PERCENT	MTL_SYSTEM_ITEMS	MRP_SAFETY_STOCK_PERCENT	
MRP_SAFETY_STOCK	MFG_LOOKUPS	MEANING	Derived from MFG_LOOKUPS for LOOKUP_TYPE MTL_SAFETY_STOCK_TYPE for the MRP_SAFETY_STOCK_CODE column.
MIN_MINMAX_QUANTITY	MTL_SYSTEM_ITEMS	MIN_MINMAX_QUANTITY	
MAX_MINMAX_QUANTITY	MTL_SYSTEM_ITEMS	MAX_MINMAX_QUANTITY	
MINIMUM_ORDER_QUANTITY	MTL_SYSTEM_ITEMS	MINIMUM_ORDER_QUANTITY	
FIXED_ORDER_QUANTITY	MTL_SYSTEM_ITEMS	FIXED_ORDER_QUANTITY	
FIXED_DAYS_SUPPLY	MTL_SYSTEM_ITEMS	FIXED_DAYS_SUPPLY	
MAXIMUM_ORDER_QUANTITY	MTL_SYSTEM_ITEMS	MAXIMUM_ORDER_QUANTITY	
ATP_RULE_NAME	MTL_ATP_RULES	RULE_NAME	Derives RULE_NAME from MTL_ATP_RULES for the ATP_RULE_ID.
PICKING_RULE_NAME	MTL_PICKING_RULES	PICKING_RULE_NAME	Derives PICKING_RULE_NAME from MTL_PICKING_RULES for the PICKING_RULE_ID.
RESERVABLE_TYPE	MFG_LOOKUPS	MEANING	Derived from MFG_LOOKUPS for the LOOKUP_TYPE MTL_RESERVATION_CONTROL.

Document Field	Oracle Applications Table/View Name	Column Name	Description
POSITIVE_MEASUREMENT_ERROR	MTL_SYSTEM_ITEMS	POSITIVE_MEASUREMENT_ERROR	Percent error above measured quantity.
NEGATIVE_MEASUREMENT_ERROR	MTL_SYSTEM_ITEMS	NEGATIVE_MEASUREMENT_ERROR	Percent error below measured quantity.
ENGINEERING_ECN_CODE	MTL_SYSTEM_ITEMS	ENGINEERING_ECN_CODE	Engineering ECN code.
ENGINEERING_ITEM	MTL_SYSTEM_ITEMS_KFV	CONCATENATE_D_SEGMENTS	Derived from MTL_SYSTEM_ITEMS_KFV for the ENGINEERING_ITEM_ID.
ENGINEERING_DATE	MTL_SYSTEM_ITEMS	ENGINEERING_DATE	
SERVICE_STARTING_DELAY	MTL_SYSTEM_ITEMS	SERVICE_STARTING_DELAY	Days after shipment that service begins.
VENDOR_WARRANTY_FLAG	MTL_SYSTEM_ITEMS	VENDOR_WARRANTY_FLAG	
SERVICEABLE_COMPONENT_FLAG	MTL_SYSTEM_ITEMS	SERVICEABLE_COMPONENT_FLAG	
SERVICEABLE_PRODUCT_FLAG	MTL_SYSTEM_ITEMS	SERVICEABLE_PRODUCT_FLAG	
BASE_WARRANTY_SERVICE_ITEM	MTL_SYSTEM_ITEMS_KFV	CONCATENATE_D_SEGMENTS	Derived from MTL_SYSTEM_ITEMS_KFV for the BASE_WARRANTY_SERVICE_ITEM_ID and organization.
PAYMENT_TERM_NAME	RA_TERMS	TERM_NAME	Derived from RA_TERMS_B.
PREVENTATIVE_MAINTENANCE_FLAG	MTL_SYSTEM_ITEMS	PREVENTATIVE_MAINTENANCE_FLAG	
PRIMARY_SPECIALIST_FULL_NAME	PER_ALL_PEOPLE_F	FULL_NAME	Derived from PER_ALL_PEOPLE_F for PRIMARY_SPECIALIST_ID.

Document Field	Oracle Applications Table/View Name	Column Name	Description
SECONDARY_SPECIALIST_FULL_NAME	PER_ALL_PEOPLE_F	FULL_NAME	Derived from PER_ALL_PEOPLE_F for SECONDARY_SPECIALIST_ID.
SERVICEABLE_ITEM_CLASS_NAME	CS_SERVICEABLE_ITEM_CLASSES	SERVICEABLE_ITEM_CLASS_NAME	Derives CS_SERVICEABLE_ITEM_CLASSES for the SERVICEABLE_ITEM_CLASSES_ID.
TIME_BILLABLE_FLAG	MTL_SYSTEM_ITEMS	TIME_BILLABLE_FLAG	Indicates whether service hours are billable.
MATERIAL_BILLABLE_FLAG	MTL_SYSTEM_ITEMS	MATERIAL_BILLABLE_FLAG	Indicates whether service items are billable.
EXPENSE_BILLABLE_FLAG	MTL_SYSTEM_ITEMS	EXPENSE_BILLABLE_FLAG	Indicates whether service expenses are billable.
PRORATE_SERVICE_FLAG	MTL_SYSTEM_ITEMS	PRORATE_SERVICE_FLAG	Indicates cost of service may be prorated.
COVERAGE_SCHEDULE_NAME	CS_COVERAGE_SCHEDULES	COVERAGE_SCHEDULE_NAME	Derived from CS_COVERAGE_SCHEDULES for the COVERAGE_SCHEDULE_ID
SERVICE_DURATION_PERIOD_CODE	MTL_SYSTEM_ITEMS	SERVICE_DURATION_PERIOD_CODE	
SERVICE_DURATION	MTL_SYSTEM_ITEMS	SERVICE_DURATION	
WARRANTY_VENDOR_NAME	PO_VENDORS	VENDOR_NAME	Derived from PO_VENDORS for the WARRANTY_VENDOR_ID.
MAX_WARRANTY_AMOUNT	MTL_SYSTEM_ITEMS	MAX_WARRANTY_AMOUNT	
RESPONSE_TIME_PERIOD_CODE	MTL_SYSTEM_ITEMS	RESPONSE_TIME_PERIOD_CODE	
RESPONSE_TIME_VALUE	MTL_SYSTEM_ITEMS	RESPONSE_TIME_VALUE	

Document Field	Oracle Applications Table/View Name	Column Name	Description
INVOICEABLE_ITEM_FLAG	MTL_SYSTEM_ITEMS	INVOICEABLE_ITEM_FLAG	
INVOICE_ENABLED_FLAG	MTL_SYSTEM_ITEMS	INVOICE_ENABLED_FLAG	
MUST_USE_APPROVED_VENDOR_FLAG	MTL_SYSTEM_ITEMS	MUST_USE_APPROVED_VENDOR_FLAG	Indicates purchases restricted to approved supplier.
OUTSIDE_OPERATION_FLAG	MTL_SYSTEM_ITEMS	OUTSIDE_OPERATION_FLAG	
OUTSIDE_OPERATION_UOM_TYPE	MTL_SYSTEM_ITEMS	OUTSIDE_OPERATION_UOM_TYPE	Outside operation unit of measure.
SAFETY_STOCK_BUCKET_DAYS	MTL_SYSTEM_ITEMS	SAFETY_STOCK_BUCKET_DAYS	
AUTO_REDUCE_MPS	MTL_SYSTEM_ITEMS	AUTO_REDUCE_MPS	Automatically deletes MPS entries in a period.
COSTING_ENABLED_FLAG	MTL_SYSTEM_ITEMS	COSTING_ENABLED_FLAG	
AUTO_CREATED_CONFIG_FLAG	MTL_SYSTEM_ITEMS	AUTO_CREATE_CONFIG_FLAG	Indicates configuration item automatically created.
CYCLE_COUNT_ENABLED_FLAG	MTL_SYSTEM_ITEMS	CYCLE_COUNT_ENABLED_FLAG	Indicates item may be cycle counted.
DEMAND_SOURCE_LINE	Not Used	Not Used	Used for inbound transactions.
COPY_ITEM_NUMBER	Not Used	Not Used	Derives ITEM_ID from MTL_SYSTEM_ITEMS_KFV for the concatenated segment. Used for inbound transactions.
SET_ID	Not Used	Not Used	Set identifier used for ATO. Used for inbound transactions only.

Document Field	Oracle Applications Table/View Name	Column Name	Description
REVISION	MTL_SYSTEM_ITEMS	REVISION	Item revision.
ITEM_TYPE	FND_COMMON_LOOKUPS	MEANING	User-defined item type. Derived from FND_COMMON_LOOKUPS, where LOOKUP_TYPE = ITEM_TYPE for the ITEM_TYPE as LOOKUP_CODE.
MODEL_CONFIG_CLAUSE_NAME	MTL_SYSTEM_ITEMS	MODEL_CONFIG_CLAUSE_NAME	
SHIP_MODEL_COMPLETE_FLAG	MTL_SYSTEM_ITEMS	SHIP_MODEL_COMPLETE_FLAG	Indicates model must be complete to ship.
MRP_PLANNING	MFG_LOOKUPS	MEANING	Derived from MFG_LOOKUPS, where LOOKUP_TYPE = MRP_PLANNING_CODE.
RETURN_INSPECTION_REQUIREMENT	MTL_SYSTEM_ITEMS	RETURN_INSPECTION_REQUIREMENT	Derived from MFG_LOOKUPS, where LOOKUP_TYPE = MTL_RETURN_INSPECTION.
DEMAND_SOURCE_TYPE	Not used	Not Used	Used for inbound transactions.
DEMAND_SOURCE_HEADER_ID	Not Used	Not Used	Used for inbound transactions.
TEMPLATE_NAME	Not Used	Not Used	Item Template Name. Used for inbound transactions.
COPY_ORGANIZATION_NAME	Not Used	Not Used	Derives ORGANIZATION_ID from ORG_ORGANIZATION_DEFINITIONS. Used for inbound transactions.

Document Field	Oracle Applications Table/View Name	Column Name	Description
ATO_FORECAST_CONTROL	MFG_LOOKUPS	MEANING	Type of forecast control for ATO. Derived from MFG_LOOKUPS, where LOOKUP_TYPE = MRP_ATO_FORECAST_CONTROL.
MATERIAL_COST	Not Used	Not Used	Used for inbound transactions.
MATERIAL_SUB_ELEMENT	Not Used	Not Used	Used for inbound transactions.
MATERIAL_OH_RATE	Not Used	Not Used	Used for inbound transactions.
MATERIAL_OH_SUBELEMENT	Not Used	Not Used	Used for inbound transactions.
RELEASE_TIME_FENCE	MFG_LOOKUPS	MEANING	Indicates whether the item is auto releasable by MRP. Derived from MFG_LOOKUPS, where LOOKUP_TYPE is MTL_RELEASE_TIME_FENCE.
RELEASE_TIME_FENCE_DAYS	MTL_SYSTEM_ITEMS	RELEASE_TIME_FENCE_DAYS	Number of days within which the item should be auto released.
CONTAINER_ITEM_FLAG	MTL_SYSTEM_ITEMS	CONTAINER_ITEM_FLAG	Indicates whether the item is a container. Used for shipping sales orders.
VEHICLE_ITEM_FLAG	MTL_SYSTEM_ITEMS	VEHICLE_ITEM_FLAG	Indicates whether the item is a vehicle. Used for shipping sales orders.
MAXIMUM_LOAD_WEIGHT	MTL_SYSTEM_ITEMS	MAXIMUM_LOAD_WEIGHT	Maximum load weight of a container or a vehicle that can be used for shipping sales orders.
MINIMUM_FILL_PERCENT	MTL_SYSTEM_ITEMS	MINIMUM_FILL_PERCENT	Minimum fill condition under which the container or vehicle should be used.

Document Field	Oracle Applications Table/View Name	Column Name	Description
CONTAINER_TYPE_CODE	MTL_SYSTEM_ITEMS	CONTAINER_TYPE_CODE	Container type code for container items. Container types are user-definable.
INTERNAL_VOLUME	MTL_SYSTEM_ITEMS	INTERNAL_VOLUME	Internal volume for container items. Used by shipping to calculate container capacity restrictions.
WH_UPDATE_DATE	MTL_SYSTEM_ITEMS	WH_UPDATE_DATE	Warehouse update date for tracking changes relevant to data collected in Data Warehouse.

1.1. MTL_ITEM_CATEGORIES

Document Field	Oracle Applications Table/View Name	Column Name	Description
INV_ITEM_ORG_ID	None		Concatenated String INVENTORY_ITEM_ID '-' ORGANIZATION_ID.
CATEGORY_SET_NAME	MTL_ITEM_CATEGORIES_V	CATEGORY_SET_NAME	Category set name.
CATEGORY_NAME	MTL_ITEM_CATEGORIES_V	CATEGORY_NAME	Concatenated category name.
TRANSACTION_TYPE	Not Used	Not Used	Used for inbound transactions.

1.1.1. MTL_ITEM_REVISIONS

Document Field	Oracle Applications Table/View Name	Column Name	Description
REVISION	MTL_ITEM_REVISIONS	REVISION	Item revision code.
INV_ITEM_ORG_ID	None		Concatenated string INVENTORY_ITEM_ID '-' ORGANIZATION_ID.
CHANGE_NOTICE	MTL_ITEM_REVISIONS	CHANGE_NOTICE	

Document Field	Oracle Applications Table/View Name	Column Name	Description
ECN_INITIATION_DATE	MTL_ITEM_REVISIONS	ECN_INITIATION_DATE	Engineering change initiation date.
IMPLEMENTATION_DATE	MTL_ITEM_REVISIONS	IMPLEMENTATION_DATE	
IMPLEMENTED_SERIAL_NUMBER	MTL_ITEM_REVISIONS	IMPLEMENTED_SERIAL_NUMBER	
EFFECTIVITY_DATE	MTL_ITEM_REVISIONS	EFFECTIVITY_DATE	
REVISED_ITEM_SEQUENCE_ID	MTL_ITEM_REVISIONS	REVISED_ITEM_SEQUENCE_ID	Identifies multiple occurrences of the same item on an engineering change order.
DESCRIPTION	MTL_ITEM_REVISIONS	DESCRIPTION	

Send Manufacturing BOM Service

The name of this service is:

WmOAMFG107SC.billsOfMaterial107SC.fromOA.manufacturingBOM:sendManufacturingBOM

This service sends to the webMethods Integration Server information on the parent, component, and substitute Manufacturing Bills of Material items, as well as revisions.

Database Scripts

This service uses the following database scripts:

Script	Description
wm_install_from_mfgbom.sql	Runs the scripts listed below, except the uninstall script.
wm_from_mfgbom_vw.sql	Creates the following required view components for Manufacturing BOM outbound transactions: <ul style="list-style-type: none"> ■ WM_BOM_BILL_OF_MTLS_VW ■ WM_BOM_ITEM_REVISIONS_VW ■ WM_BOM_INVENTORY_COMPS_VW ■ WM_BOM_SUBSTITUTE_COMPS_VW ■ WM_BOM_REFERENCE_DESGS_VW ■ WM_BOM_BILL_OF_MTLS_QRY_VW
wm_from_mfgbom_trg.sql	Creates the following trigger components to be used for writing to the WM_TRACKCHANGES table to indicate a new or updated documents: <ul style="list-style-type: none"> ■ WM_BOM_BILL_OF_MTLS_IUD_TRG ■ WM_MTL_ITEM_REV_IUD_TRG ■ WM_BOM_INV_COMPS_IUD_TRG ■ WM_BOM_SUB_COMPS_IUD_TRG ■ WM_BOM_REF_DESGS_IUD_TRG
wm_disable_from_mfgbom.sql	Disables the triggers installed by wm_from_mfgbom_trg.sql.

Script	Description
wm_enable_from_mfgbom.sql	Re-enables the triggers installed by wm_from_mfgbom_trg.sql.
wm_drop_from_mfgbom.sql	Uninstalls all components created by wm_install_from_mfgbom.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definition:

- `getManufacturingBOMTxn107SC.txp`

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

You can send new or changed Manufacturing Bills of Material data using the following services:

- `specifyDefaultSettings` specifies the default parameter settings required for service execution. You should change these settings accordingly.
- `lockTxnCtrl` determines whether the `sendManufacturingBOM` service is ready for execution. That is, it determines whether no other instance of this service is already in process.
 - If `lockTxnCtrl` returns `False`, it indicates that another instance of this service is already in process. The service exits and waits for the next scheduled execution.
 - If `lockTxnCtrl` returns `True`, it indicates that the service is ready to execute, and the Manufacturing BOM row in the control table is locked and updated so that the status is changed to be `INPROCESS`. This prevents any other Manufacturing BOM service from executing.
- `getManufacturingBOMTxn` service queries the Oracle Applications database for any Manufacturing BOM Transaction to be processed. The number of records returned depends on the parameter value specified in the above map.
- `processBizDoc` is the customizable step that sends the business document to the Trading Partner. This step needs to be customized to receive a Success or an Error status of the document transfer. If an error occurs in a particular business document transfer, the error information must be sent back to the calling service (`sendManufacturingBOM` in this case). The error information passed back should

have the document identifiers. If a particular document is transferred successfully to the Trading Partner, no information needs to be sent back to the calling program (sendManufacturingBOM).

- Based on the Debug Mode specified during execution, it is determined to either purge or update the records in the WM_TRACKCHANGES custom table.
 - If the Debug Mode is TRUE, based on the purge criteria the records in the WM_TRACKCHANGES table are updated and the PROCESSED_FLAG is set to Y. This ensures that the same set of records is not picked up during next polling interval. The updateTrackChanges service updates the PROCESSED_FLAG in the WM_TRACKCHANGES table to Y so that same information is not picked up again during next polling instance.
 - If the Debug Mode is FALSE, based on the purge criteria the records in the WM_TRACKCHANGES table are deleted. The **purgeTrackChanges** service purges the records from the WM_TRACKCHANGES table.
- If an error occurs in transferring the Business document, the **insertTransferERRInfo** service inserts a new record in the WM_TRACKCHANGES table so that same document can be picked up during the next polling interval.
- Any document that is created and deleted in between two successive polling operations will not be delivered. Such records are not selected by getManufacturingBOMTxn; thus they are not updated as processed or purged in the WM_TRACKCHANGES table.
 - If the Debug Mode is TRUE, the **updateUnqualifiedRec** service recognizes such records and updates the corresponding PROCESSED_FLAG to Y.
 - If the Debug Mode is FALSE, the **purgeUnqualifiedRec** service recognizes such records and deletes them from the WM_TRACKCHANGES table.
- **unlockTxnCtrl** service releases the lock on the Custom Control table so that next polling instance of the sendManufacturingBOM service can begin.
- If the document transfer is successful, the flow exits.

For more details on send transactions, see [“Overview of Send Service Transaction Processing” on page 38](#).

Business Document Structure

This service uses the following business document structure:

- 1.0. BOM_BILL_OF_MTLS
 - 1.1. MTL_ITEM_REVISIONS

- 1.2. BOM_INVENTORY_COMPS

- 1.2.1. BOM_SUB_COMPS

- 1.2.2. BOM_REF_DESGS

1.0. BOM_BILL_OF_MTLS

Document Field	Oracle Applications Table/View Name	Column Name	Description
WEB_TRANSACTION_ID			
DOCUMENT_TYPE			MFGBOM will be populated in this field.
DOCUMENT_STATUS			UPDATE, INSERT or DELETE will be populated in this field.
BILL_SEQUENCE_ID	BOM_BILL_OF_MATERIALS	BILL_SEQUENCE_ID	Unique identifier for bill.
ITEM_NUMBER	MTL_SYSTEM_ITEMS_B_KFV	CONCATENATE_D_SEGMENTS	Concatenated segments.
ORGANIZATION_NAME	HR_ALL_ORGANIZATION_UNITS	NAME	
COMMON_ITEM_NUMBER	MTL_SYSTEM_ITEMS_B_KFV	CONCATENATE_D_SEGMENTS	Concatenates segments for common bill item.
COMMON_ORG_NAME	HR_ALL_ORGANIZATION_UNITS	NAME	Organization name for common bill.
ALTERNATE_BOM_DESIGNATOR	BOM_BILL_OF_MATERIALS	ALTERNATE_BOM_DESIGNATOR	Alternate Bill of Materials designator code.
COMMON_ALT_BOM_DESIGNATOR	BOM_BILL_OF_MATERIALS	COMMON_ALT_BOM_DESIGNATOR	Common alternate Bill of Materials designator code.
SPECIFIC_ASSEMBLY_COMMENT	BOM_BILL_OF_MATERIALS	SPECIFIC_ASSEMBLY_COMMENT	Specific Bill of Materials comment.
PENDING_FROM_ECN	BOM_BILL_OF_MATERIALS	PENDING_FROM_ECN	Change notice that created this Bill of Materials.
ASSEMBLY_TYPE	BOM_BILL_OF_MATERIALS	ASSEMBLY_TYPE	For Manufacturing/Engineering.

Document Field	Oracle Applications Table/View Name	Column Name	Description
DEMAND_SOURCE_LINE			Not used for Bills of Material outbound transactions.
SET_ID			Not used for Bills of Material outbound transactions.
DEMAND_SOURCE_TYPE			Not used for Bills of Material outbound transactions.
DEMAND_SOURCE_HEADER_ID			Not used for Bills of Material outbound transactions.
NEXT_EXPLODE_DATE	BOM_BILL_OF_MATERIALS	NEXT_EXPLODE_DATE	Next date when pre-explosion will be refreshed.
UNIT_OF_MEASURE	MTL_UNITS_OF_MEASURE_TL	UNIT_OF_MEASURE	
REVISION			Not used for Bills of Material outbound transactions.

1.1. MTL_ITEM_REVISIONS

Document Field	Oracle Applications Table/View Name	Column Name	Description
BILL_SEQUENCE_ID	BOM_INVENTORY_COMPONENTS	BILL_SEQUENCE_ID	Unique bill identifier.
ITEM_NUMBER	MTL_SYSTEM_ITEMS_B_KFV	CONCATENATED_SEGMENTS	Concatenated segments.
ORGANIZATION_NAME	HR_ALL_ORGANIZATION_UNITS	NAME	
REVISION	MTL_ITEM_REVISIONS	REVISION	Item revision code.
CHANGE_NOTICE	MTL_ITEM_REVISIONS	CHANGE_NOTICE	Engineering change order number.
ECN_INITIATION_DATE	MTL_ITEM_REVISIONS	ECN_INITIATION_DATE	Engineering change order initiation date.
IMPLEMENTATION_DATE	MTL_ITEM_REVISIONS	IMPLEMENTATION_DATE	Engineering change order implementation date.

Document Field	Oracle Applications Table/View Name	Column Name	Description
EFFECTIVITY_DATE	MTL_ITEM_REVISIONS	EFFECTIVITY_DATE	Revision effective date.
REVISED_ITEM_SEQUENCE_ID	MTL_ITEM_REVISIONS	REVISED_ITEM_SEQUENCE_ID	Revised item sequence ID.
TRANSACTION_TYPE			Not used for Bills of Material outbound transactions.

1.2. BOM_INVENTORY_COMPS

Document Field	Oracle Applications Table/View Name	Column Name	Description
COMPONENT_SEQUENCE_ID	BOM_INVENTORY_COMPONENTS	COMPONENT_SEQUENCE_ID	Unique component identifier.
BILL_SEQUENCE_ID	BOM_INVENTORY_COMPONENTS	BILL_SEQUENCE_ID	Unique bill identifier.
ASSEMBLY_ITEM_NUMBER	MTL_SYSTEM_ITEMS_B_KFV	CONCATENATE_D_SEGMENTS	Inventory item number of manufactured assembly.
ALTERNATE_BOM_DESIGNATOR	BOM_BILL_OF_MATERIALS	ALTERNATE_BOM_DESIGNATOR	Alternate Bill of Materials designator code.
ORGANIZATION_NAME	HR_ALL_ORGANIZATION_UNITS	NAME	
COMPONENT_ITEM_NUMBER	MTL_SYSTEM_ITEMS_B_KFV	CONCATENATE_D_SEGMENTS	
OPERATION_SEQ_NUM	BOM_INVENTORY_COMPONENTS	OPERATION_SEQ_NUM	Operation sequence number.
EFFECTIVITY_DATE	BOM_INVENTORY_COMPONENTS	EFFECTIVITY_DATE	Effective date.
LOCATION_NAME	MTL_ITEM_LOCATIONS	DESCRIPTION	Supply locator name.
SUBSTITUTE_COMPONENT_NUMBER			Not used for Bills of Material outbound transactions.

Document Field	Oracle Applications Table/View Name	Column Name	Description
ITEM_NUM	BOM_INVENTORY_COMPONENTS	ITEM_NUM	Item sequence within Bill of Materials structure.
COMPONENT_QUANTITY	BOM_INVENTORY_COMPONENTS	COMPONENT_QUANTITY	Quantity of components in Bill of Materials.
COMPONENT_YIELD_FACTOR	BOM_INVENTORY_COMPONENTS	COMPONENT_YIELD_FACTOR	Factor used to multiply component quantity with to obtain required component quantity.
COMPONENT_REMARKS	BOM_INVENTORY_COMPONENTS	COMPONENT_REMARKS	
CHANGE_NOTICE	BOM_INVENTORY_COMPONENTS	CHANGE_NOTICE	Engineering change order number.
IMPLEMENTATION_DATE	BOM_INVENTORY_COMPONENTS	IMPLEMENTATION_DATE	Date on which engineering change order was implemented.
DISABLE_DATE	BOM_INVENTORY_COMPONENTS	DISABLE_DATE	
PLANNING_FACTOR	BOM_INVENTORY_COMPONENTS	PLANNING_FACTOR	Factor used to multiply component quantity with to obtain planning quantity.
QUANTITY_RELATED	BOM_INVENTORY_COMPONENTS	QUANTITY_RELATED	Identifier to indicate if this component has quantity-related reference designators.
SO_BASIS	BOM_INVENTORY_COMPONENTS	SO_BASIS	Quantity basis used by Oracle Order Management to determine how many units of component to put on an order.
OPTIONAL	BOM_INVENTORY_COMPONENTS	OPTIONAL	Indicates whether component is optional in bill.

Document Field	Oracle Applications Table/View Name	Column Name	Description
MUTUALLY_EXCLUSIVE_OPTIONS	BOM_INVENTORY_COMPONENTS	MUTUALLY_EXCLUSIVE_OPTIONS	Indicates whether one or more children of component can be picked when taking an order.
INCLUDE_IN_COST_ROLLUP	BOM_INVENTORY_COMPONENTS	INCLUDE_IN_COST_ROLLUP	Indicates whether this component is to be used when rolling up costs.
CHECK_ATP	BOM_INVENTORY_COMPONENTS	CHECK_ATP	Indicates whether ATP check is required.
SHIPPING_ALLOWED	BOM_INVENTORY_COMPONENTS	SHIPPING_ALLOWED	Indicates whether component is allowed to ship.
REQUIRED_TO_SHIP	BOM_INVENTORY_COMPONENTS	REQUIRED_TO_SHIP	Indicates whether component is required to ship.
REQUIRED_FOR_REVENUE	BOM_INVENTORY_COMPONENTS	REQUIRED_FOR_REVENUE	Indicates whether component is required for revenue.
INCLUDE_ON_SHIP_DOCS	BOM_INVENTORY_COMPONENTS	INCLUDE_ON_SHIP_DOCS	Indicates whether component is displayed on shipping documents.
INCLUDE_ON_BILL_DOCS	BOM_INVENTORY_COMPONENTS	INCLUDE_ON_BILL_DOCS	Indicates whether component is displayed on billing documents.
LOW_QUANTITY	BOM_INVENTORY_COMPONENTS	LOW_QUANTITY	Minimum quantity allowed on an order.
HIGH_QUANTITY	BOM_INVENTORY_COMPONENTS	HIGH_QUANTITY	Maximum quantity allowed on an order.
ACD_TYPE	BOM_INVENTORY_COMPONENTS	ACD_TYPE	Add, change, or disable code for component on an engineering change order.
REVISED_ITEM_NUMBER	MTL_SYSTEM_ITEMS_B_KFV	CONCATENATED_SEGMENTS	
REVISION	MTL_ITEM_REVISIONS	REVISION	

Document Field	Oracle Applications Table/View Name	Column Name	Description
WIP_SUPPLY_TYPE	BOM_INVENTORY_COMPONENTS	WIP_SUPPLY_TYPE	
SUPPLY_SUBINVENTORY	BOM_INVENTORY_COMPONENTS	SUPPLY_SUBINVENTOR Y	
REFERENCE_DESIGNATOR			Not used for Bills of Material outbound transactions.
BOM_ITEM_TYPE	BOM_INVENTORY_COMPONENTS	BOM_ITEM_TYPE	Type of item (de-normalized from BOM Item Type in MTL_SYSTEM_ITEMS).
OPERATION_LEAD_TIME_PERCENT	BOM_INVENTORY_COMPONENTS	OPERATION_LEAD_TIME_PERCENT	Operation offset from first operation in routing.
UNIT_OF_MEASURE	MTL_UNITS_OF_MEASURE_TL	UNIT_OF_MEASURE	

1.2.1. BOM_SUB_COMPS

Document Field	Oracle Applications Table/View Name	Column Name	Description
BILL_SEQUENCE_ID	BOM_INVENTORY_COMPONENTS	BILL_SEQUENCE_ID	Unique bill identifier.
SUBSTITUTE_COMP_NUMBER	MTL_SYSTEM_ITEMS_B_KFV	CONCATENATE D_SEGMENTS	Substitute component number.
SUBSTITUTE_ITEM_QUANTITY	BOM_SUBSTITUTE_COMPONENTS	SUBSTITUTE_ITEM_QUANTITY	Substitute component quantity.
ACD_TYPE	BOM_SUBSTITUTE_COMPONENTS	ACD_TYPE	ADD or DELETE code on an engineering change order.
CHANGE_NOTICE	BOM_SUBSTITUTE_COMPONENTS	CHANGE_NOTICE	Engineering change order number.
ORGANIZATION_NAME	HR_ALL_ORGANIZATION_UNITS	NAME	

Document Field	Oracle Applications Table/View Name	Column Name	Description
ASSEMBLY_ITEM_NUMBER	MTL_SYSTEM_ITEMS_B_KFV	CONCATENATE_D_SEGMENTS	Inventory item number of manufactured assembly.
ALTERNATE_BOM_DESIGNATOR	BOM_BILL_OF_MATERIALS	ALTERNATE_BOM_DESIGNATOR	Alternate Bill of Materials designator code.
COMPONENT_ITEM_NUMBER	MTL_SYSTEM_ITEMS_B_KFV	CONCATENATE_D_SEGMENTS	Inventory item number of component.
OPERATION_SEQ_NUM	BOM_INVENTORY_COMPONENTS	OPERATION_SEQ_NUM	Manufacturing operation sequence number.
EFFECTIVITY_DATE	BOM_INVENTORY_COMPONENTS	EFFECTIVITY_DATE	Effective date.
UNIT_OF_MEASURE	MTL_UNITS_OF_MEASURE_TL	UNIT_OF_MEASURE	

1.2.2. BOM_REF_DESGS

Document Field	Oracle Applications Table/View Name	Column Name	Description
BILL_SEQUENCE_ID	BOM_INVENTORY_COMPONENTS	BILL_SEQUENCE_ID	Unique bill identifier.
COMPONENT_REFERENCE_DESIGNATOR	BOM_REFERENCE_DESIGNATORS	COMPONENT_REFERENCE_DESIGNATOR	Component reference designator.
ORGANIZATION_NAME	HR_ALL_ORGANIZATION_UNITS	NAME	
ASSEMBLY_ITEM_NUMBER	MTL_SYSTEM_ITEMS_B_KFV	CONCATENATE_D_SEGMENTS	
ALTERNATE_BOM_DESIGNATOR	BOM_BILL_OF_MATERIALS	ALTERNATE_BOM_DESIGNATOR	Alternate Bill of Materials identifier.
COMPONENT_ITEM_NUMBER	MTL_SYSTEM_ITEMS_B_KFV	CONCATENATE_D_SEGMENTS	

Document Field	Oracle Applications Table/View Name	Column Name	Description
OPERATION_SEQ_NUM	BOM_INVENTORY_COMPONENTS	OPERATION_SEQ_NUM	Manufacturing operation sequence number.
EFFECTIVITY_DATE	BOM_INVENTORY_COMPONENTS	EFFECTIVITY_DATE	Date on which functionality will be enabled.
REF_DESIGNATOR_COMMENT	BOM_REFERENCE_DESIGNATORS	REF_DESIGNATOR_COMMENT	Reference designator comment.
CHANGE_NOTICE	BOM_REFERENCE_DESIGNATORS	CHANGE_NOTICE	Engineering change order number.
ACD_TYPE	BOM_REFERENCE_DESIGNATORS	ACD_TYPE	ADD or DELETE code on an engineering change order.

Order Management Predefined Transaction Services

■ Overview	372
■ Query Pick Detail Service	373
■ Query Sales Order Service	374
■ Receive Sales Order Service	375
■ Send Pick Detail Service	387
■ Send Sales Order Service	394

Overview

This chapter describes the predefined transaction services provided in the Oracle Applications Adapter’s 10.7SC Order Management package.

The table below shows the predefined transaction services organized by Oracle Applications module. This chapter lists the transactions in alphabetical order.

Oracle Applications Module	Predefined Transactions
Order Management	<ul style="list-style-type: none">■ “Query Sales Order Service” on page 374■ “Receive Sales Order Service” on page 375■ “Send Sales Order Service” on page 394
Shipping	<ul style="list-style-type: none">■ “Query Pick Detail Service” on page 373■ “Send Pick Detail Service” on page 387

For more information about using the predefined transaction services, see [Chapter 1, “Predefined Transaction Services” on page 17](#).

Query Pick Detail Service

The name of this service is:

WmOAOMG107SC.shipping107SC.queryOA.pickDetail:queryPickDetail

This service queries for information about the inventory transfer when an item moves from one location to another.

Use this service to retrieve the pick details based on the following parameters:

- PICK_SLIP_RULE: The Pick Slip Rule used by Pick Release when Ship Confirm executes.
- SCHEDULED_SHIP_DATE_FROM: Scheduled ship date (from).
- SCHEDULED_SHIP_DATE_TO: Scheduled ship date (to).
- ORDER_NUMBER: The order number assigned on creation of order.
- ORDER_TYPE_NAME: Type of the order.
- FROM_ORGANIZATION_NAME: The organization from which the picking executed.

Database Scripts

This service uses the same database scripts as the Send Pick Detail service.



Note: If you use this service but you do *not* use the Send Pick Detail service, you should run the `wm_disable_from_pickdetails.sql` script to disable the triggers installed by the Send Pick Detail service.

For a detailed description of these database scripts, see [“Send Pick Detail Service” on page 387](#). For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service uses the following transaction definition:

- queryPickDetailTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow executes as follows:

- `specifyDefaultSettings` specifies the default parameter settings required for service execution. You should change these settings accordingly.
- `queryPickDetailTxn` queries the Oracle Applications database for any Pick Detail information matching the parameter values. The parameters are defined as the input to this service.

For more details on query transactions, see [“Using Query Services” on page 36](#).

Business Document Structure

This service uses the same business document structure as the Send Pick Detail service. For a detailed description of the business document structure, see [“Send Pick Detail Service” on page 387](#).

Query Sales Order Service

The name of this service is:

`WmOAOMG107SC.orderManagement107SC.queryOA.salesOrder:querySalesOrder`

This service queries new, changed, and deleted booked sales orders.

Database Scripts

This service uses the same database scripts as the Send Sales Order service.



Note: If you use this service but you do *not* use the Send Sales Order service, you should run the `wm_disable_from_order.sql` script to disable the triggers installed by the Send Sales Order service.

For a detailed description of these database scripts, see [“Send Sales Order Service” on page 394](#). For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service uses the following transaction definitions

- `querySalesOrderTxn107SC.txp`

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow executes as follows:

- `querySalesOrderTxn` queries for any SalesOrder information matching the parameter values. The parameters are defined as the input to this service.

For more details on query transactions, see [“Using Query Services” on page 36](#).

Business Document Structure

This service uses the same business document structure as the Send Sales Order service. For a detailed description of the business document structure, see [“Send Sales Order Service” on page 394](#).

Receive Sales Order Service

The name of this service is:

`WmOAOMG107SC.orderManagement107SC.intoOA.salesOrder:receiveSalesOrder`

Sales Order Import is an Order Management Open Interface that consists of open interface tables and provides a set of APIs. Order Import can import new, changed, and completed sales orders or returns from other applications, such as a legacy system. Orders may come from any source such as EDI transactions processed by the Oracle e-Commerce Gateway, or internal orders created for internal requisitions developed in Oracle Purchasing or returns.

Order Import features include validation and defaulting, processing constraint checks, applying and releasing order holds, and scheduling shipments to insert, update, or delete the orders in the base Order Management tables. Order Management checks all the data during the import process to ensure its validity within Order Management. Valid transactions convert into orders with lines, reservations, price adjustments, and sales credits in the base Order Management tables.

Database Scripts

This service uses the following database scripts:

Database Script	Description
wm_install_into_order.sql	Runs all the scripts listed below, except the uninstall script.
wm_into_order_pkg.sql	Installs WM_ORDER_IMP_HANDLER_PKG. WM_HANDLE_ORDER_IMPORT which calls the WM_CONC_REQUEST_PKG.WM_REQUEST_SUBMIT procedure to submit the sales order information.
wm_drop_into_order.sql	Uninstalls all components created by wm_install_into_order.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definitions:

- setSalesOrderTxn107SC.txp
- SalesOrderTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow executes as follows:

- **specifyDefaultSettings** specifies the default parameter settings required for service execution. You should change these settings accordingly.
- **specifyConcProgParams** specifies the default parameter settings required for concurrent program execution. You should change these settings accordingly.
- **bizDocMapping** maps the incoming business document structure to the required Oracle Applications data structures (interface tables).
 - **getInventoryItemIdfromCode** is a transformer for mapping the incoming business document to the interface table structure. This service selects the inventory item ID from WAREHOUSE_CODE and ITEM_CODE.

- **getCommitmentId** is a transformer for mapping incoming business document to the interface table structure. This service derives COMMITMENT_ID from the TRX_NUMBER in the business document.
- **getOrderSourceId** derives ORDER_SOURCE_ID from the ORDER_SOURCE in the business document.
- **getWarehouseId** derives ORGANIZATION_ID from the WAREHOUSE_CODE in the business document.
- **convertToDateObject** returns outDate as a date object corresponding to the supplied inDate which is in text format. This service is used as a transformer in bizDocMapping.
- **getOASystemDateObject** returns the Oracle Applications system date.
- **getOracleAppsUserId** is a transformer for mapping the business document IData structure to the interface table IData structure. It takes ORACLE_APPS_USER_NAME as the input parameter from the business document, and queries the table FND_USER to get the USER_ID. The USER_ID information is required for insertion into the interface tables.
- **setSalesOrderTxn** inserts data into the interface table. It extracts data from the IData structure resulted in the bizDocMapping service and puts the data into the interface table in OA for Sales Order.
- **importSalesOrder** imports data to the production table from the interface table. This then calls the execSalesOrderConcProg, checkSalesOrderImportStatus, and getSalesOrder_ERR services to execute the corresponding concurrent program that inserts data into the production table to generate the error/ acknowledgement message. If the status of the execution is SUCCESS (returned by the service execSalesOrderConcProg), it checks for the record that has the returned request ID in the error interface table. If there are any errors, it specifies an error during the import process. In this case, this service calls getSalesOrder_ERR to retrieve the errors. If no record is found, it executes out of flow and processes the data import successfully. If the status of the execution is FAILED it returns the appropriate error message and discontinues execution.
- **execSalesOrderConcProg** invokes the stored procedure WM_ORDER_IMPORT_HANDLER_PKG.WM_HANDLE_ORDER_IMPORT. This procedure calls the corresponding concurrent subroutine to execute the data import process for Sales Order into Oracle Applications. This service gives Status ID, Request ID, Execution Status Message for normal concurrent program completion and a database Stored Procedure error message if an exception occurs in Stored Procedure execution.
- **checkSalesOrderImportStatus** checks if the above step has returned any stored procedure exceptions. In case of stored procedure exceptions, the dbErrorMsg record list captures the information. If there are no stored procedure exceptions, it indicates normal completion of the concurrent program stored procedure and the stores the returned message in concProgMsg record list.

- `getSalesOrder_ERR` gets the error message that occurs during the data import to the production table from interface table. Based on the parameter `REQUEST_ID` it scans the table interfaces to get the `REQUEST_ID`.

For more details on receive transactions, see [“Overview of Receive Service Transaction Processing” on page 31](#).

Business Document Structure

This service uses the business document `SalesOrderBizDoc`. Its structure is as follows:

- 1.0 ORDER_HEADERS
 - 1.1 ORDER_LINES
 - 1.1.1PRICE_ADJ
 - 1.1.2 SALES_CREDITS
 - 1.1.3 DETAILS
 - 1.2 HEADER_PRICE_ADJ
 - 1.3 HEADER_SALES_CREDIT

1.0 ORDER_HEADERS (Maps to SO_HEADERS_INTERFACE)

Field Name	Maps to Column	Description
HEADER_ID		Header Identified. Not used for inbound transactions.
ENTRY_STATUS		Order Status. Not used for inbound transactions.
ORDER_TYPE	ORDER_TYPE	Required. Order Type.
ORGANIZATION_NAME		Organization Name. Not used for inbound transaction.
ORDER_NUMBER		Unique number assigned to the order. Not used for inbound transaction.
DATE_ORDERED	DATE_ORDERED	Required. Date of the Order
ORDER_SOURCE	ORDER_SOURCE_ID	Required. Order Source Name. Use <code>SO_ORDER_SOURCES.NAME</code> to get <code>ORDER_SOURCE_ID</code> .
ORIGINAL_SYSTEM_REFERENCE	ORIGINAL_SYSTEM_REFERENCE	Required. Original document reference for the order in external system.

Field Name	Maps to Column	Description
CUSTOMER_NAME	CUSTOMER_NAME	Name of the customer of the order. Either CUSTOMER_NUMBER or CUSTOMER_NAME must be given.
CUSTOMER_NUMBER	CUSTOMER_NUMBER	Number of the customer of the order. You must use CUSTOMER_NUMBER or CUSTOMER_NAME.
PRICE_LIST_NAME	PRICE_LIST_NAME	Price List Name.
CONVERSION_RATE	CONVERSION_RATE	Conversion Rate.
CONVERSION_DATE	CONVERSION_DATE	Conversion date.
CONVERSION_TYPE_CODE	CONVERSION_TYPE_CODE	Conversion Type Code.
CURRENCY_CODE	CURRENCY_CODE	Required. Currency.
SALESREP_NAME	SALESREP_NAME	Sales Representative Name. You must use SALESREP_NAME or SALESREP_NUMBER with Book an Order.
SALESREP_NUMBER	SALESREP_NUMBER	Sales Representative Number. You must use SALESREP_NAME or SALESREP_NUMBER with Book an Order.
SALES_CHANNEL_CODE	SALES_CHANNEL_CODE	Sales channel through which order was placed.
TAX_EXEMPT_FLAG	TAX_EXEMPT_FLAG	Tax Exemption Flag. Valid values: <ul style="list-style-type: none"> ■ S: Standard ■ E: Exempt ■ R: Required
TAX_EXEMPT_NUM	TAX_EXEMPT_NUM	Tax Exemption Number
TAX_EXEMPT_REASON_CODE	TAX_EXEMPT_REASON_CODE	Tax Exemption Reason Code.
AGREEMENT_NAME	AGREEMENT_NAME	Agreement Name.
INVOICING_RULE	INVOICING_RULE	Invoicing Rule Name.
ACCOUNTING_RULE	ACCOUNTING_RULE	Accounting Rule Name.
CONTEXT	CONTEXT	Context.
TERMS_NAME	TERMS_NAME	Payment Term.

Field Name	Maps to Column	Description
SHIPMENT_PRIORITY_CODE	SHIPMENT_PRIORITY_CODE	Shipment Priority Code.
SHIP_METHOD_CODE	SHIP_METHOD_CODE	Shipping Method Code.
FREIGHT_TERMS_CODE	FREIGHT_TERMS_CODE	Freight Terms Code.
FOB_CODE	FOB_CODE	FOB Code.
SHIPPING_INSTRUCTIONS	SHIPPING_INSTRUCTIONS	Shipping Instructions.
PACKING_INSTRUCTIONS	PACKING_INSTRUCTIONS	Packaging Instructions.
PURCHASE_ORDER_NUM	PURCHASE_ORDER_NUM	Purchase Order Number.
PAYMENT_TYPE_CODE	PAYMENT_TYPE_CODE	Type of payment for orders, such as cash, check, or credit.
PAYMENT_AMOUNT	PAYMENT_AMOUNT	Amount of payment.
CHECK_NUMBER	CHECK_NUMBER	Check number if payment type is check
CREDIT_CARD_CODE	CREDIT_CARD_CODE	Credit card name if payment type is credit card.
CREDIT_CARD_HOLDER_NAME	CREDIT_CARD_HOLDER_NAME	Credit card holder name if payment type is credit card.
CREDIT_CARD_NUMBER	CREDIT_CARD_NUMBER	Credit card number if payment type is credit card.
CREDIT_CARD_EXPIRATION_DATE	CREDIT_CARD_EXPIRATION_DATE	Credit card expiration date if payment type is credit card.
CREDIT_CARD_APPROVAL_CODE	CREDIT_CARD_APPROVAL_CODE	Credit card approval code if payment type is credit card.
DATE_SHIPPED	DATE_SHIPPED	Shipment date.
SHIP_TO_ADDRESS1	SHIP_ADDRESS1	Required. Ship to address1.
SHIP_TO_ADDRESS2	SHIP_ADDRESS2	Required. Ship to address2.
SHIP_TO_ADDRESS3	SHIP_ADDRESS3	Required. Ship to address3.
SHIP_TO_ADDRESS4	SHIP_ADDRESS4	Required. Ship to address4.
SHIP_TO_CITY	SHIP_CITY	Required. Ship to city name.
SHIP_TO_COUNTY	SHIP_COUNTY	Required. Ship to county.
SHIP_TO_STATE	SHIP_STATE	Required. Ship to state.
SHIP_TO_POSTAL_CODE	SHIP_POSTAL_CODE	Required. Ship to address postal code.
SHIP_TO_COUNTRY	SHIP_COUNTRY	Required. Ship to country.

Field Name	Maps to Column	Description
SHIP_TO_CUSTOMER	SHIP_TO_CUSTOMER	Required. Ship to customer name.
INVOICE_ADDRESS1	INVOICE_ADDRESS1	Required. Invoice to organization address1.
INVOICE_ADDRESS2	INVOICE_ADDRESS2	Required. Invoice to organization address2.
INVOICE_ADDRESS3	INVOICE_ADDRESS3	Required. Invoice to organization address3.
INVOICE_ADDRESS4	INVOICE_ADDRESS4	Required. Invoice to organization address4.
INVOICE_CITY	INVOICE_CITY	Required. Invoice To Organization address city.
INVOICE_COUNTY	INVOICE_COUNTY	Required. Invoice to organization address county.
INVOICE_STATE	INVOICE_STATE	Required. Invoice to organization address state.
INVOICE_POSTAL_CODE	INVOICE_POSTAL_CODE	Required. Invoice to organization address postal code.
INVOICE_COUNTRY	INVOICE_COUNTRY	Required. Invoice to organization address country.
INVOICE_CUSTOMER	INVOICE_CUSTOMER	Required. Invoicing customer name.
SHIP_TO_CONTACT_FIRST_NAME	SHIP_TO_CONTACT_FIRST_NAME	Ship to contact first name.
SHIP_TO_CONTACT_LAST_NAME	SHIP_TO_CONTACT_LAST_NAME	Ship to contact last name.
INVOICE_TO_CONTACT_FIRST_NAME	INVOICE_TO_CONTACT_FIRST_NAME	Invoice to contact first name.
INVOICE_TO_CONTACT_LAST_NAME	INVOICE_TO_CONTACT_LAST_NAME	Invoice to contact last name.
ORDERED_BY_CONTACT_FIRST_NAME	ORDERED_BY_CONTACT_FIRST_NAME	Ordered by contact first name.
ORDERED_BY_CONTACT_LAST_NAME	ORDERED_BY_CONTACT_LAST_NAME	Required. Ordered by contact last name.
OPERATION_CODE	OPERATION_CODE	Required. Operation Code. Valid values are INSERT, UPDATE and DELETE.
APPLY_STANDARD_NOTES	APPLY_STANDARD_NOTES	Apply standard notes.

Field Name	Maps to Column	Description
ENTERED_STATE_NAME	ENTERED_STATE_NAME	Required. Entered State Name. Valid values are Entered, Partial, and Booked.
ENTERED_STATE_DATE	ENTERED_STATE_DATE	Entered state Date.
DATE_REQUESTED_CURRENT	DATE_REQUESTED_CURRENT	Current requested date.
SCHEDULE_STATUS_CODE	SCHEDULE_STATUS_CODE	Schedule Status Code. Valid values are DEMANDED and RESERVED.
COMPLETE_FLAG	COMPLETE_FLAG	Set this flag to Y to import historical data.

1.1 ORDER_LINES (Maps to SO_LINES_INTERFACE)

Field Name	Maps to Column	Description
LINE_ID		Order Line Identifier. Not used for inbound transactions.
HEADER_ID		Order Header Identifier. Not used for inbound transactions.
ORIGINAL_SYSTEM_LINE_REFERENCE	ORIGINAL_SYSTEM_LINE_REFERENCE	Required. Identifier of line from a source system outside of Oracle Order Management.
ORDER_SOURCE	ORDER_SOURCE_ID	Required. Order Source Name. Use SO_ORDER_SOURCES.NAME to get ORDER_SOURCE_ID.
LINE_NUMBER	LINE_NUMBER	Required. Unique Line sequence number within the Oracle Order Management.
COMMITMENT	COMMITMENT_ID	Commitment for order line. Gets COMMITMENT_ID = CUSTOMER_TRX_ID for a given COMMITMENT when it is equal to RA_CUSTOMER_TRX_ALL_TRX_NUMBER.
ACCOUNTING_RULE	ACCOUNTING_RULE	Accounting rule.
INVOICING_RULE	INVOICING_RULE	Invoicing rule.
SHIP_TO_CUSTOMER	SHIP_TO_CUSTOMER	Required. Ship to customer Name.
SHIP_ADDRESS1	SHIP_ADDRESS1	Required. Ship to address1.
SHIP_ADDRESS2	SHIP_ADDRESS2	Required. Ship to address2.
SHIP_ADDRESS3	SHIP_ADDRESS3	Required. Ship to address3.
SHIP_ADDRESS4	SHIP_ADDRESS4	Required. Ship to address4.
SHIP_CITY	SHIP_CITY	Required. Ship to city name.

Field Name	Maps to Column	Description
SHIP_COUNTY	SHIP_COUNTY	Required. Ship to county.
SHIP_STATE	SHIP_STATE	Required. Ship to state.
SHIP_POSTAL_CODE	SHIP_POSTAL_CODE	Required. Ship to address postal code.
SHIP_COUNTRY	SHIP_COUNTRY	Required. Ship to country.
SHIP_TO_CONTACT_FIRST_NAME	SHIP_TO_CONTACT_FIRST_NAME	Required. Ship to contact first name.
SHIP_TO_CONTACT_LAST_NAME	SHIP_TO_CONTACT_LAST_NAME	Required. Ship to contact last name.
TAX_CODE	TAX_CODE	Tax Code. Required if the tax exemption flag in the order header is set to R.
LINK_TO_LINE_REF	LINK_TO_LINE_REF	Link to line reference.
PARENT_LINE_REF	PARENT_LINE_REF	Parent line reference.
SHIP_SET_NUMBER	SHIP_SET_NUMBER	Ship set number.
SHIPMENT_PRIORITY_CODE	SHIPMENT_PRIORITY_CODE	Shipment priority code.
WAREHOUSE_CODE	WAREHOUSE_ID	Warehouse. Use MTL_PARAMETERS.ORGANIZATION_CODE to get the ORGANIZATION_ID.
SHIP_METHOD_CODE	SHIP_METHOD_CODE	Shipping method code.
UNIT_CODE	UNIT_CODE	Required. Unit code.
ORDERED_QUANTITY	ORDERED_QUANTITY	Required. Quantity Ordered of this line item.
LIST_PRICE	LIST_PRICE	List Price. Required if CALCULATE_PRICE is set to N.
SELLING_PRICE	SELLING_PRICE	Selling Price. Required if CALCULATE_PRICE is set to N.
DATE_REQUESTED_CURRENT	DATE_REQUESTED_CURRENT	Date the customer requested receipt of the order.
ITEM	INVENTORY_ITEM_ID	Required. Line Item. Use MTL_SYSTEM_ITEMS_KFV.CONCATENATED_SEGMENTS for the current organization to get the INVENTORY_ITEM_ID.
SHIPPED_QUANTITY	SHIPPED_QUANTITY	Shipping Quantity.

Field Name	Maps to Column	Description
SCHEDULED_SHIPMENT_DATE	SCHEDULED_SHIPMENT_DATE	Schedule shipment date. Date must be later or the same as the date ordered in the order header.
PRICING_CONTEXT	PRICING_CONTEXT	Pricing Context.
ITEM_TYPE_CODE	ITEM_TYPE_CODE	Required. Item Type Code. Valid values are CLASS, KIT, MODEL, and STANDARD.
OPTION_FLAG	OPTION_FLAG	Optional flag that indicates whether the item on this line is an option item or option class. For base models or regular items, type N or leave this field null.
ORDER_CATEGORY	ORDER_CATEGORY	Order Category. Type R for regular order line or leave it null.
CONTEXT	CONTEXT	Context.
CALCULATE_PRICE	CALCULATE_PRICE	Required. Use N if the LIST_PRICE and SELLING_PRICE are provided; otherwise set to Y if you want the system to automatically price the order line.
OPERATION_CODE	OPERATION_CODE	Required. Operation Code. Valid values are INSERT, UPDATE, and DELETE.

1.1.1 PRICE_ADJ (Maps to SO_PRICE_ADJUSTMENTS_INTERFACE)

Field Name	Maps to Column	Description
HEADER_ID		Required. Order Header Identifier. Not used for inbound transactions.
LINE_ID		Order Line Identifier. Not used for inbound transactions.
ORDER_SOURCE	ORDER_SOURCE_ID	Required. Order Source name. Use SO_ORDER_SOURCES.NAME to get ORDER_SOURCE_ID.
DISCOUNT_NAME	DISCOUNT_NAME	Required. Discount Name.
PERCENT	PERCENT	Required. Percentage of discount. This percent must match the percentage defined for the discount if the discount cannot be overriden. If SO_LINES_INTERFACE.CALCUALTE_PRICE value is N, this discount on line must equal the difference between the list price and selling price.

Field Name	Maps to Column	Description
CONTEXT	CONTEXT	Context.
PRICING_CONTEXT	PRICING_CONTEXT	Pricing Context.
OPERATION_CODE	OPERATION_CODE	Required. Operation Code. Valid values are INSERT, UPDATE, and DELETE.

1.1.2 SALES_CREDITS (Maps to SO_SALES_CREDITS_INTERFACE)

Field Name	Maps to Column	Description
HEADER_ID		Order Header Identifier. Not used for inbound transactions.
LINE_ID		Order Line Identifier. Not used for inbound transactions.
ORDER_SOURCE	ORDER_SOURCE_ID	Required. Order Source Name. Use SO_ORDER_SOURCES.NAME to get ORDER_SOURCE_ID.
SALESREP_NAME	SALESREP_NAME	Sales person name. Required for booked orders.
SALES_CREDIT_TYPE	SALES_CREDIT_TYPE	Required. Sales credit type.
PERCENT	PERCENT	Required. Percentage. The total quota credit must equal 100 percent.
CONTEXT	CONTEXT	Context.
OPERATION_CODE	OPERATION_CODE	Required. Operation Code. Valid values are INSERT, UPDATE, and DELETE.

1.1.3 DETAILS (Maps to SO_LINE_DETAILS_INTERFACE)

Field Name	Maps to Column	Description
LINE_ID		Order Line Identifier. Not used for inbound transactions.
ORDER_SOURCE	ORDER_SOURCE_ID	Required. Order Source Name. Use SO_ORDER_SOURCES.NAME to get ORDER_SOURCE_ID.
QUANTITY	QUANTITY	Required. Quantity for the Order Line Detail.
SCHEDULE_DATE	SCHEDULE_DATE	Schedule date for the order line detail. It is required for booking.

Field Name	Maps to Column	Description
LOT_NUMBER	LOT_NUMBER	Lot number. Required if SO_HEADERS_INTERFACE.SCHEDULE_STATUS_CODE is set to 'Reserved'; otherwise this field should be null.
SUBINVENTORY	SUBINVENTORY	Subinventory used when releasing items from inventory. Required for booking orders.
CUSTOMER_REQUESTED_LOT_FLAG	CUSTOMER_REQUESTED_LOT_FLAG	Indicates whether the customer requested the specific subinventory, lot, or revision.
CONTEXT	CONTEXT	Context.
REVISION	REVISION	The revision number of the scheduled item. Set this value to Reserved if SO_HEADERS_INTERFACE.SCHEDULE_STATUS_CODE is set to Reserved; otherwise this field should be null.
WAREHOUSE	WAREHOUSE_ID	Warehouse from which items ship. Use MTL_PARAMETERS.ORGANIZATION_CODE to get the ORGANIZATION_ID.

1.2 HEADER_PRICE_ADJ (Maps to SO_PRICE_ADJUSTMENTS_INTERFACE)

Field Name	Maps to Column	Description
HEADER_ID		Order Header Identifier. Not used for inbound transactions.
ORDER_SOURCE	ORDER_SOURCE_ID	Required. Order Source Name. Use SO_ORDER_SOURCES.NAME to get ORDER_SOURCE_ID.
DISCOUNT_NAME	DISCOUNT_NAME	Required. Discount Name.
PERCENT	PERCENT	Required. Percentage of Discount. This percentage must match the percentage defined on the discount if the discount is not overridable. If SO_LINES_INTERFACE.CALCULATE_PRICE = N, then this discount must equal the difference between the list price and selling price.
CONTEXT	CONTEXT	Context.

Field Name	Maps to Column	Description
PRICING_CONTEXT	PRICING_CONTEXT	Pricing Context.
OPERATION_CODE	OPERATION_CODE	Required. Operation Code. Valid values are INSERT, UPDATE, and DELETE.

1.3. HEADER_SALES_CREDIT (Maps to SO_SALES_CREDITS_INTERFACE)

Field Name	Maps to Column	Description
HEADER_ID		Order header identifier. Not used for inbound transactions.
ORDER_SOURCE	ORDER_SOURCE_ID	Required. Order source name. Use SO_ORDER_SOURCES.NAME to get ORDER_SOURCE_ID.LINE_ID.
SALESREP_NAME	SALESREP_NAME	Sales person name.
SALES_CREDIT_TYPE	SALES_CREDIT_TYPE	Required. Sale credit type.
PERCENT	PERCENT	Required. Percentage. The total credit must equal 100 percent.
CONTEXT	CONTEXT	QUANTITY.
OPERATION_CODE	OPERATION_CODE	Required. LINE_ID.

Send Pick Detail Service

The name of this service is:

WmOAOMG107SC.shipping107SC.fromOA.pickDetail:sendPickDetail

This service provides information about the inventory transfer when the item moves from one location to another.

Database Scripts

This service uses the following database scripts:

Database Script	Description
wm_install_from_pickdetails.sql	Runs all the scripts listed below, except the uninstall script.
wm_from_pickdetails_vw.sql	Creates the following required view components for Pick Detail outbound transactions: <ul style="list-style-type: none">■ WM_SO_PICKING_HEADERS_VW■ WM_SO_PICKING_LINES_VW■ WM_SO_PICKING_HEADERS_QRY_VW
wm_from_pickdetails_trg.sql	Create the following trigger component to be used for writing to the WM_TRACKCHANGES table to indicate a new or updated document: <ul style="list-style-type: none">■ WM_SO_PICKING_HEADERS_U_TRG
wm_drop_from_pickdetails.sql	Uninstalls all components created by wm_install_from_pickdetails.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definition:

- getPickDetailTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow executes as follows:

- **specifyDefaultSettings** specifies the default parameter settings required for service execution. You should change these settings accordingly.
- **lockTxnCtrl** determines whether the sendPickDetail service is ready for execution. That is, it determines whether there is no other instance of this service already in process.
 - If lockTxnCtrl returns True, it means that the service is ready to execute, the PICKDETAIL row in the control table is locked and updated, and the status is

changed to INPROCESS. This prevents any other PickDetail service from executing.

- If lockTxnCtrl returns False, it means that another instance of this service is already in process. The service exits and waits for next scheduled execution.
- **getPickDetailTxn** service queries the Oracle Applications database for any Pick Detail Transaction to be processed. The number of records returned depends on the parameter value specified in the above map.
- **processBizDoc** is the customizable step that sends the business document to a trading partner. You must customize this step to receive a Success or an Error status of the document transfer. If an error occurs during a business document transfer, then the error information is sent back to the sendPickDetail calling service. The error information passed back should have the document identifiers. If a particular document is transferred successfully to the trading partner, no information needs to be sent back to the sendPickDetail calling service.
- Based on the Debug Mode specified during execution, it either purges or updates the records in the WM_TRACKCHANGES custom table.
 - If the Debug Mode is TRUE, then based on the purge criteria, the records in the WM_TRACKCHANGES table are updated and the PROCESSED_FLAG is set to Y. This ensures that same set of records is not picked up during next polling interval. The **updateTrackChanges** service updates the PROCESSED_FLAG in the WM_TRACKCHANGES table to Y so that the same information is not picked up again during next polling instance.
 - If the Debug Mode is FALSE, then based on the purge criteria the records in the WM_TRACKCHANGES table are deleted.
- **purgeTrackChanges** purges the records from the WM_TRACKCHANGES table.
- **insertTransferERRInfo** service inserts a new record in the WM_TRACKCHANGES table if an error occurs while transferring the Business document, so this document can be picked up during the next polling interval.
- **unlockTxnCtrl** service releases the lock on the Custom Control table so that the next polling instance of sendPickDetail service can begin.

For more details on send transactions, see [“Overview of Send Service Transaction Processing” on page 38](#).

Business Document Structure

This service use the following business document structure:

- 1.0 PICK_DETAILS
 - 1.1 PICKING_LINES

1.0 PICK_DETAILS

Document Field	Oracle Applications Table/View Name	Column Name	Description
WEB_TRANSACTION_ID			
DOCUMENT_TYPE			Value is PICKDETAIL.
DOCUMENT_STATUS			Valid values are INSERT and QUERY.
PICKING_HEADER_ID	SO_PICKING_HEADERS_ALL	PICKING_HEADER_ID	Unique identifier for Picking Header Information.
BATCH_NAME	SO_PICKING_BATCHES_ALL	NAME	Picking Batch Name used to Pick Release.
ORDER_NUMBER	SO_HEADERS_ALL	ORDER_NUMBER	User-visible number of the order.
ORDER_CATEGORY	SO_HEADERS_ALL	ORDER_CATEGORY	Order category. Use: I for Internal RMA for Regular S for Service P for Internal Sales Order
PICK_SLIP_RULE	WSH_PICK_SLIP_RULES	NAME	Pick Slip rule. Valid values are Carrier/Departure, Customer, Delivery, Order Number, Priority, and SubInventory.
ORDER_CURRENCY_CODE	SO_HEADERS_ALL	CURRENCY_CODE	Currency code of the order.
CUSTOMER_NAME	RA_CUSTOMERS	CUSTOMER_NAME	Name of customer in order.
FREIGHT_TERMS	SO_LOOKUPS	MEANING	Terms of Freight. Valid values are Paid, Due, and TBD (To be decided).
ORDER_TYPE_NAME	SO_ORDER_TYPES_ALL	NAME	Name of Order Type. Examples are Standard, Standard-Intl, and Quoted Standard.
FROM_ORGANIZATION_NAME	HR_ORGANIZATION_UNITS	NAME	Organization Name from which pick release executes.

Document Field	Oracle Applications Table/View Name	Column Name	Description
SHIP_TO_SITE_USE_LOCATION	RA_SITE_USES_ALL	LOCATION	Customer Ship To Site location.
STATUS_CODE	SO_PICKING_HEADERS_ALL	STATUS_CODE	Pick Slip Status Code. Valid values are Open and Closed.
PICK_SLIP_NUMBER	SO_PICKING_HEADERS_ALL	PICK_SLIP_NUMBER	Pick Slip Number assigned on Pick Release.
WAYBILL_NUM	SO_PICKING_HEADERS_ALL	WAYBILL_NUM	Waybill number.
PICKER_NAME	FND_USER	USER_NAME	Name of picker.
PACKER_NAME	FND_USER	USER_NAME	Name of the packer.
WEIGHT	SO_PICKING_HEADERS_ALL	WEIGHT	Weight of item.
WEIGHT_UNIT_CODE	SO_PICKING_HEADERS_ALL	WEIGHT_UNIT_CODE	Weight unit code.
NUMBER_OF_BOXES	SO_PICKING_HEADERS_ALL	NUMBER_OF_BOXES	Number of boxes as in Order.
SHIP_METHOD_CODE	SO_PICKING_HEADERS_ALL	SHIP_METHOD_CODE	Shipping method code.
SHIP_METHOD_NAME	ORG_FREIGHT	DESCRIPTION	Shipping method name.
INVOICED_LINES	Shp_Picking_Headers_Pkg.INVOICED_LINES	INVOICED_LINES	Indicates whether lines are invoiced.
DATE_RELEASED	SO_PICKING_HEADERS_ALL	DATE_RELEASED	Pick release date.
DATE_SHIPPED	SO_PICKING_HEADERS_ALL	DATE_SHIPPED	Shipment date.
DATE_CONFIRMED	SO_PICKING_HEADERS_ALL	DATE_CONFIRMED	Ship confirm date.
EXPECTED_ARRIVAL_DATE	SO_PICKING_HEADERS_ALL	EXPECTED_ARRIVAL_DATE	Expected shipment arrival date.

1.1 PICKING_LINES

Document Field	Oracle Applications Table/View Name	Column Name	Description
PICKING_LINE_ID	SO_PICKING_LINES_ALL	PICKING_LINE_ID	Unique identifier for Picking Line.
PICKING_HEADER_ID	SO_PICKING_LINES_ALL	PICKING_HEADER_ID	Unique identifier for Picking Header.
PICK_SLIP_NUMBER	SO_PICKING_HEADERS_ALL	PICK_SLIP_NUMBER	Pick Slip Number.
PICKING_RULE_NAME	MTL_PICKING_RULES	PICKING_RULE_NAME	Material Picking Rule name.
SEQUENCE_NUMBER	SO_PICKING_LINES_ALL	SEQUENCE_NUMBER	Sequence Number.
COMPONENT_CODE	SO_PICKING_LINES_ALL	COMPONENT_CODE	Component code for the picking line item.
COMPONENT_RATIO	SO_PICKING_LINES_ALL	COMPONENT_RATIO	Component Ratio.
REQUESTED_QUANTITY	SO_PICKING_LINES_ALL	REQUESTED_QUANTITY	Quantity requested to be shipped.
ITEM_DESCRIPTION	MTL_SYSTEM_ITEMS	DESCRIPTION	Inventory item description.
SUBINV_RESTRICTED_FLAG	MTL_SYSTEM_ITEMS	RESTRICT_SUBINVENTORIES_CODE	Indicates whether subinventories are restricted to pre-defined list.
REVISION_CONTROL_FLAG	MTL_SYSTEM_ITEMS	REVISION_QTY_CONTROL_CODE	Indicates whether item is under revision quantity control.
LOT_CONTROL_FLAG	MTL_SYSTEM_ITEMS	LOT_CONTROL_CODE	Indicate whether item has no lot control or full lot control.

Document Field	Oracle Applications Table/View Name	Column Name	Description
SERIAL_NUMBER_CONTROL_FLAG	MTL_SYSTEM_ITEMS	SERIAL_NUMBER_CONTROL_CODE	Indicates whether item has no serial number control, predefined serial numbers, or Dynamic entry at inventory receipt.
INCLUDED_ITEM_FLAG	SO_PICKING_LINES_ALL	INCLUDED_ITEM_FLAG	
RESERVABLE_FLAG	MTL_SYSTEM_ITEMS	RESERVABLE_TYPE	Indicates whether item is reservable.
DATE_REQUESTED	SO_PICKING_LINES_ALL	DATE_REQUESTED	Date when the line is requested.
ORIGINAL_REQUESTED_QUANTITY	SO_PICKING_LINES_ALL	ORIGINAL_REQUESTED_QUANTITY	Item originally requested in Picking Lines.
ORGANIZATION_NAME	HR_ORGANIZATION_UNITS	NAME	Organization name.
SHIPPED_QUANTITY	SO_PICKING_LINES_ALL	SHIPPED_QUANTITY	Quantity shipped after ship confirm.
CANCELLED_QUANTITY	SO_PICKING_LINES_ALL	CANCELLED_QUANTITY	Quantity cancelled in course of picking.
SHIPMENT_PRIORITY_CODE	SO_PICKING_LINES_ALL	SHIPMENT_PRIORITY_CODE	Code for ship priority. Valid values are Priority 1, Priority 2, and so forth.
SHIP_METHOD_CODE	SO_PICKING_LINES_ALL	SHIP_METHOD_CODE	Shipment Method Code.
DATE_CONFIRMED	SO_PICKING_LINES_ALL	DATE_CONFIRMED	Date on which ship confirmation is done.
INVOICED_QUANTITY	SO_PICKING_LINES_ALL	INVOICED_QUANTITY	Quantity invoiced at ship confirmation.
INVENTORY_STATUS	SO_PICKING_LINES_ALL	INVENTORY_STATUS	Inventory status at Pick Confirm.

Document Field	Oracle Applications Table/View Name	Column Name	Description
UNIT_CODE	SO_PICKING_LINES_ALL	UNIT_CODE	Unit Of Measure Code.
PRIMARY_UOM_CODE	MTL_SYSTEM_ITEMS	PRIMARY_UOM_CODE	Primary unit of measure code.
SCHEDULE_DATE	SO_PICKING_LINES_ALL	SCHEDULE_DATE	Scheduled ship date.
DEMAND_CLASS_CODE	SO_PICKING_LINES_ALL	DEMAND_CLASS_CODE	The class to which the demand is tied.
CONFIGURATION_ITEM_FLAG	SO_PICKING_LINES_ALL	CONFIGURATION_ITEM_FLAG	Flag to indicate configuration item demand type.
LATEST_ACCEPTABLE_DATE	SO_PICKING_LINES_ALL	LATEST_ACCEPTABLE_DATE	Latest acceptable date for shipment.

Send Sales Order Service

The name of this service is:

WmOMG107SC.orderManagement107SC.fromOA.salesOrder:sendSalesOrder

This service notifies and delivers sales order changes. This service selects Booked Order Records only.

Database Scripts

This service uses the following database scripts:

Database Script	Description
wm_install_from_order.sql	Runs all the scripts listed below, except the uninstall script.
wm_from_order_vw.sql	Creates the following required view components for Sales Order outbound transactions: <ul style="list-style-type: none"> ■ WM_ORDER_HEADERS_VW ■ WM_ORDER_LINES_VW ■ WM_LINES_DETAIL_VW ■ WM_ORDER_HEADER_QRY_VW ■ WM_LINE_SALES_CREDITS_VW ■ WM_HEADER_SALES_CREDIT_VW ■ WM_LINE_PRICE_ADJ_VW ■ WM_HRD_PRICE_ADJ_VW
wm_from_order_trg.sql	Creates the following trigger components to be used for writing to the WM_TRACKCHANGES table to indicate a new or updated document: <ul style="list-style-type: none"> ■ WM_ORDER_HEADERS_IU_TRG ■ WM_ORDER_LINES_IU_TRG ■ WM_SALES_CREDITS_IUD_TRG ■ WM_LINE_DETAILS_IUD_TRG
wm_disable_from_order.sql	Disables the triggers installed by wm_from_order_trg.sql.
wm_enable_from_order.sql	Re-enables the triggers installed by wm_from_order_trg.sql.
wm_drop_from_order.sql	Uninstalls all components created by wm_install_from_order.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definition:

- `getSalesOrderTxn107SC.txp`

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow executes as follows:

- `specifyDefaultSettings` specifies the default parameter settings required for service execution. You should change these settings accordingly.
- `lockTxnCtrl` determines whether the `sendSalesOrder` service is ready for execution. That is, it determines whether there is no other instance of this service already in process.
 - If `lockTxnCtrl` returns `False`, it means that another instance of this service is already in process. The service exits and waits for next scheduled execution.
 - If `lockTxnCtrl` returns `True`, it means that the service is ready to execute. The `SalesOrder` row in the control table is locked and updated so that the status is changed to `INPROCESS`, which prevents other `SalesOrder` services from executing.
- `getSalesOrderTxn` service queries the Oracle Applications database for any `SalesOrder` Transaction to be processed. The number of records returned depends on the parameter value specified in the above map.
- `processBizDoc` is the customizable step which sends the business document to the trading partner by looping against each document. You must customize this step to receive a `SUCCESS` or an `ERROR` status of the document transfer along with the error information. The transfer status and any error information are logged against each document.
- Loop against each document. Based on the `Debug Mode` specified during execution, it either purges or updates the records in the `WM_TRACKCHANGES` custom table.
 - If the `Debug Mode` is `TRUE`, the records in the `WM_TRACKCHANGES` table are updated and the `PROCESSED_FLAG` is set to `Y`. This ensures that same sets of records are not picked up during next polling interval. The `updateTrackChanges` service updates the `PROCESSED_FLAG` in the `WM_TRACKCHANGES` table to `Y` and updates `PROCESSED_DATE` to `sysdate` so that the same information is not picked up again during next polling instance.

- If the Debug Mode is FALSE, the records in the WM_TRACKCHANGES table are deleted. The `purgeTrackChanges` service purges the records from the WM_TRACKCHANGES table.
- Loop against each document. Based on the Transfer Status, the `insertTransferERRInfo` service inserts a new record in the WM_TRACKCHANGES table so that the same document can be picked up during the next polling interval.
- `unlockTxnCtrl` service releases the lock on the Custom Control table so that next polling instance of `sendVendor` service can begin.
- `getLastError` logs errors that occur in the above steps.
- `unlockTxnCtrl` releases the lock on the Custom Control table.

For more details on send transactions, see [“Overview of Send Service Transaction Processing” on page 38](#).

Business Document Structure

This service uses the following business document structure:

- 1.0 ORDER_HEADERS
 - 1.1. ORDER_LINES
 - 1.1.1 PRICE_ADJ
 - 1.1.2 SALES_CREDITS
 - 1.1.3. DETAILS
 - 1.2 HEADER_PRICE_ADJ
 - 1.3 HEADER_SALES_CREDIT

1.0 ORDER_HEADERS

Document Field	Oracle Applications Table/View Name	Column Name	Description
WEB_TRANSACTION_ID			
DOCUMENT_TYPE			Value is SALESORDER.
DOCUMENT_STATUS			Valid values are INSERT, UPDATE, and DELETE.
HEADER_ID	SO_HEADERS_ALL	HEADER_ID	
ENTRY_STATUS	SO_HEADERS_ALL	Function Used	Order Status.

Document Field	Oracle Applications Table/View Name	Column Name	Description
ORDER_TYPE	SO_ORDER_TYPES_ALL	NAME	Order type.
ORGANIZATION_NAME	HR_ALL_ORGANIZATION_UN ITS	NAME	Organization name.
ORDER_NUMBER	SO_HEADERS_ALL	ORDER_NUMBER	Unique number assigned to the order.
DATE_ORDERED	SO_HEADERS_ALL	DATE_ORDERED	Date of the order.
ORDER_CATEGORY	SO_LOOKUPS	MEANING	Order category name.
ORDER_SOURCE	SO_ORDER_SOURCES	NAME	Order source name.
ORIGINAL_SYSTEM_REFERENCE	SO_HEADERS_ALL	ORIGINAL_SYSTEM_REFEREN CE	Original system reference for the order in external system.
CUSTOMER_NAME	RA_CUSTOMERS	CUSTOMER_NAME	Name of the customer of the order.
CUSTOMER_NUMBER	RA_CUSTOMERS	CUSTOMER_NUMBER	Number of the customer of the order.
PRICE_LIST_NAME	SO_PRICE_LISTS	NAME	Price list name.
CONVERSION_RATE	SO_HEADERS_ALL	CONVERSION_RATE	Conversion rate.
CONVERSION_DATE	SO_HEADERS_ALL	CONVERSION_DATE	Conversion date.
CONVERSION_TYPE_CODE	SO_HEADERS_ALL	CONVERSION_TYPE_CODE	Conversion type code.
CURRENCY_CODE	SO_HEADERS_ALL	CURRENCY_CODE	Currency code.
SALESREP_NAME	RA_SALESREPS_ALL	NAME	Sales representative name.
SALESREP_NUMBER	RA_SALESREPS_ALL	SALESREP_NUMBER	Sales representative number.
SALES_CHANNEL_CODE	SO_HEADERS_ALL	SALES_CHANNEL_CODE	Sales channel through which order was placed.

Document Field	Oracle Applications Table/View Name	Column Name	Description
TAX_EXEMPT_FLAG	SO_HEADERS_ALL	TAX_EXEMPT_FLAG	Tax exemption flag. Use: S for Standard E for Exempt R for Required
TAX_EXEMPT_NUM	SO_HEADERS_ALL	TAX_EXEMPT_NUM	Tax exemption number.
TAX_EXEMPT_REASON_CODE	SO_HEADERS_ALL	TAX_EXEMPT_REASON_CODE	Tax exemption reason code.
AGREEMENT_NAME	SO_AGREEMENTS	NAME	Agreement name.
INVOICING_RULE	RA_RULES	NAME	Invoicing rule name.
ACCOUNTING_RULE	RA_RULES	NAME	Accounting rule name.
CONTEXT	SO_HEADERS_ALL	CONTEXT	Context.
TERMS_NAME	RA_TERMS	NAME	Payment term.
SHIPMENT_PRIORITY_CODE	SO_HEADERS_ALL	SHIPMENT_PRIORITY_CODE	Shipment priority code.
SHIP_METHOD_CODE	SO_HEADERS_ALL	SHIP_METHOD_CODE	Ship method code.
FREIGHT_TERMS_CODE	SO_HEADERS_ALL	FREIGHT_TERMS_CODE	Freight terms code.
FOB_CODE	SO_HEADERS_ALL	FOB_CODE	FOB code.
SHIPPING_INSTRUCTIONS	SO_HEADERS_ALL	SHIPPING_INSTRUCTIONS	Shipping instructions.
PACKING_INSTRUCTIONS	SO_HEADERS_ALL	PACKING_INSTRUCTIONS	Packaging instructions.
PURCHASE_ORDER_NUM	SO_HEADERS_ALL	PURCHASE_ORDER_NUM	Customer purchase order number.
PAYMENT_TYPE_CODE	SO_HEADERS_ALL	PAYMENT_TYPE_CODE	Type of payment for order, such as cash, check, or credit.
PAYMENT_AMOUNT	SO_HEADERS_ALL	PAYMENT_AMOUNT	Amount of payment.
CHECK_NUMBER	SO_HEADERS_ALL	CHECK_NUMBER	Check number if payment type is check.

Document Field	Oracle Applications Table/View Name	Column Name	Description
CREDIT_CARD_CODE	SO_HEADERS_ALL	CREDIT_CARD_CODE	Credit card name if payment type is credit card.
CREDIT_CARD_HOLDER_NAME	SO_HEADERS_ALL	CREDIT_CARD_HOLDER_NAME	Credit card holder name if payment type is credit card.
CREDIT_CARD_NUMBER	SO_HEADERS_ALL	CREDIT_CARD_NUMBER	Credit card number if payment type is credit card.
CREDIT_CARD_EXPIRATION_DATE	SO_HEADERS_ALL	CREDIT_CARD_EXPIRATION_DATE	Credit card expiration date if payment type is credit card.
CREDIT_CARD_APPROVAL_CODE	SO_HEADERS_ALL	CREDIT_CARD_APPROVAL_CODE	Credit card approval code if payment type is credit card.
DATE_SHIPPED			Not used for outbound transactions.
SHIP_TO_ADDRESS1	RA_ADDRESSES_ALL	ADDRESS1	Ship to Address1.
SHIP_TO_ADDRESS2	RA_ADDRESSES_ALL	ADDRESS2	Ship to Address2.
SHIP_TO_ADDRESS3	RA_ADDRESSES_ALL	ADDRESS3	Ship to Address3.
SHIP_TO_ADDRESS4	RA_ADDRESSES_ALL	ADDRESS4	Ship to Address4.
SHIP_TO_CITY	RA_ADDRESSES_ALL	CITY	Ship to city.
SHIP_TO_COUNTY	RA_ADDRESSES_ALL	COUNTY	Ship to county.
SHIP_TO_STATE	RA_ADDRESSES_ALL	STATE	Ship to state.
SHIP_TO_POSTAL_CODE	RA_ADDRESSES_ALL	POSTAL_CODE	Ship to postal code.
SHIP_TO_COUNTRY	RA_ADDRESSES_ALL	COUNTRY	Ship to country
SHIP_TO_CUSTOMER	RA_CUSTOMERS	CUSTOMER_NAME	Ship to Customer Name
INVOICE_ADDRESS1	RA_ADDRESSES_ALL	ADDRESS1	Invoice To Address1
INVOICE_ADDRESS2	RA_ADDRESSES_ALL	ADDRESS2	Invoice To Address2
INVOICE_ADDRESS3	RA_ADDRESSES_ALL	ADDRESS3	Invoice To Address3
INVOICE_ADDRESS4	RA_ADDRESSES_ALL	ADDRESS4	Invoice To Address4
INVOICE_CITY	RA_ADDRESSES_ALL	CITY	Invoice To city
INVOICE_COUNTY	RA_ADDRESSES_ALL	COUNTY	Invoice To county
INVOICE_STATE	RA_ADDRESSES_ALL	STATE	Invoice To state
INVOICE_POSTAL_CODE	RA_ADDRESSES_ALL	POSTAL_CODE	Invoice To postal Code

Document Field	Oracle Applications Table/View Name	Column Name	Description
INVOICE_COUNTRY	RA_ADDRESSES_ALL	COUNTRY	Invoice To country
INVOICE_CUSTOMER	RA_CUSTOMERS	CUSTOMER_NAME	Invoicing Customer Name
SHIP_TO_CONTACT_FIRST_NAME	RA_CONTACTS	FIRST_NAME	Ship To Contact first name
SHIP_TO_CONTACT_LAST_NAME	RA_CONTACTS	LAST_NAME	Ship To Contact last name
INVOICE_TO_CONTACT_FIRST_NAME	RA_CONTACTS	FIRST_NAME	Invoice To Contact first name
INVOICE_TO_CONTACT_LAST_NAME	RA_CONTACTS	LAST_NAME	Invoice to contact last name
ORDERED_BY_CONTACT_FIRST_NAME	RA_CONTACTS	FIRST_NAME	Ordered by Contact First Name.
ORDERED_BY_CONTACT_LAST_NAME	RA_CONTACTS	LAST_NAME	Ordered by Contact last Name.
OPERATION_CODE			Operation Code. Not used in Outbound.
APPLY_STANDARD_NOTES			Apply Standard Notes. Not used in Outbound.
ENTERED_STATE_NAME			Entered State Name. Not used in Outbound.
ENTERED_STATE_DATE			Entered State Date. Not used in Outbound.
DATE_REQUESTED_CURRENT	SO_HEADERS_ALL	DATE_REQUESTED_CURRENT	Current requested Date.
SCHEDULE_STATUS_CODE			Schedule Status Code. Not used in Outbound.
COMPLETE_FLAG			Complete Flag. Not used for outbound transactions.

1.1 ORDER_LINES

Document Field	Oracle Applications Table/View Name	Column Name	Description
LINE_ID	SO_LINES_ALL	LINE_ID	Order Line Identifier.
HEADER_ID	SO_LINES_ALL	HEADER_ID	Order Header Identifier. Same as HEADER_ID in Order above.
ORIGINAL_SYSTEM_LINE_REFERENCE	SO_LINES_ALL	ORIGINAL_SYSTEM_LINE_REFERENCE	Line identifier from a source system outside of Oracle Order Management.
ORIGINAL_SYSTEM_REFERENCE	SO_HEADERS_ALL	ORIGINAL_SYSTEM_REFERENCE	Original system reference for the order in external system.
ORDER_SOURCE	SO_ORDER_SOURCES	NAME	Order source name.
LINE_NUMBER	SO_LINES_ALL	LINE_NUMBER	Line sequence number within the order.
LINE_TYPE	SO_LINES_ALL	LINE_TYPE_CODE	Line type code.
COMMITMENT	RA_CUSTOMER_TRX_ALL	TRX_NUMBER	Commitment number.
ACCOUNTING_RULE	RA_RULES	MEANING	Accounting rule.
INVOICING_RULE	RA_RULES	MEANING	Invoicing rule.
SHIP_TO_CUSTOMER	RA_CUSTOMERS	CUSTOMER_NAME	Ship to Customer name.
SHIP_ADDRESS1	RA_ADDRESSES_ALL	ADDRESS1	Ship To Address1.
SHIP_ADDRESS2	RA_ADDRESSES_ALL	ADDRESS2	Ship To Address2.
SHIP_ADDRESS3	RA_ADDRESSES_ALL	ADDRESS3	Ship To Address3.
SHIP_ADDRESS4	RA_ADDRESSES_ALL	ADDRESS4	Ship To Address4.
SHIP_CITY	RA_ADDRESSES_ALL	CITY	Ship To City.
SHIP_COUNTY	RA_ADDRESSES_ALL	COUNTY	Ship To County.
SHIP_STATE	RA_ADDRESSES_ALL	STATE	Ship To State.
SHIP_POSTAL_CODE	RA_ADDRESSES_ALL	POSTAL_CODE	Ship To Postal Code.
SHIP_COUNTRY	RA_ADDRESSES_ALL	COUNTRY	Ship To Country.
SHIP_TO_CONTACT_FIRST_NAME	RA_CONTACTS	FIRST_NAME	Ship To Contact first name.

Document Field	Oracle Applications Table/View Name	Column Name	Description
SHIP_TO_CONTACT_LAST_NAME	RA_CONTACTS	LAST_NAME	Ship To Contact last name.
TAX_CODE	SO_LINES_ALL	TAX_CODE	Tax Code
LINK_TO_LINE_REF	SO_LINES_ALL	ORIGINAL_SYSTEM_LINE_REFERENCE	Link to line reference.
PARENT_LINE_REF	SO_LINES_ALL	ORIGINAL_SYSTEM_LINE_REFERENCE	Parent line reference.
SHIP_SET_NUMBER	SO_LINES_ALL	SHIP_SET_NUMBER	Ship set number.
SHIPMENT_PRIORITY_CODE	SO_LINES_ALL	SHIPMENT_PRIORITY_CODE	Shipment priority code.
SHIP_METHOD_CODE	SO_LINES_ALL	SHIP_METHOD_CODE	Shipping method code.
WAREHOUSE_CODE	MTL_PARAMETERS	ORGANIZATION_CODE	Warehouse.
UNIT_CODE	SO_LINES_ALL	UNIT_CODE	Unit code.
ORDERED_QUANTITY	SO_LINES_ALL	ORDERED_QUANTITY	Ordered quantity.
LIST_PRICE	SO_LINES_ALL	LIST_PRICE	List price.
SELLING_PRICE	SO_LINES_ALL	SELLING_PRICE	Selling price
DATE_REQUESTED_CURRENT	SO_LINES_ALL	DATE_REQUESTED_CURRENT	Date the customer requested receipt of the order.
ITEM	MTL_SYSTEM_ITEMS_KFV	CONCATENATED_SEGMENTS	Item code.
SHIPPED_QUANTITY	SO_LINES_ALL	SHIPPED_QUANTITY	Shipping quantity.
SCHEDULED_SHIPMENT_DATE	SO_LINES_ALL	SCHEDULE_DATE	Schedule date.
PRICING_CONTEXT	SO_LINES_ALL	PRICING_CONTEXT	Pricing context.
PRICING_METHOD_CODE	SO_LINES_ALL	PRICING_METHOD_CODE	Pricing method code.

Document Field	Oracle Applications Table/View Name	Column Name	Description
ITEM_TYPE_CODE	SO_LINES_ALL	ITEM_TYPE_CODE	Item type code.
OPTION_FLAG	SO_LINES_ALL	OPTION_FLAG	Indicates whether the item on this line is an option item or option class.
ORDER_CATEGORY	SO_LOOKUPS	MEANING	Order Category.
CONTEXT	SO_LINES_ALL	CONTEXT	Context.
CALCULATE_PRICE			Flag to indicate if price is to be calculated. Not Used in Outbound
OPERATION_CODE			Operation code. Not used for outbound transactions.

1.1.1 PRICE_ADJ

Document Field	Oracle Applications Table/View Name	Column Name	Description
HEADER_ID	SO_PRICE_ADJUSTMENTS	HEADER_ID	Order Header Identifier. Same as HEADER_ID in Order.
LINE_ID	SO_PRICE_ADJUSTMENTS	LINE_ID	Order line identifier. Same as LINE_ID in Order Lines above.
ORIGINAL_SYSTEM_REFERENCE	SO_HEADERS_ALL	ORIGINAL_SYSTEM_REFERENCE	Original system reference for the order in external system.
ORIGINAL_SYSTEM_LINE_REFERENCE	SO_LINES_ALL	ORIGINAL_SYSTEM_LINE_REFERENCE	Line identifier from a source system outside of Oracle Order Management.
ORDER_SOURCE	SO_ORDER_SOURCES	NAME	Order source name.
DISCOUNT_NAME	SO_DISCOUNTS	NAME	Discount name.
PERCENT	SO_PRICE_ADJUSTMENTS	PERCENT	Percentage.
CONTEXT	SO_PRICE_ADJUSTMENTS	CONTEXT	Context.

Document Field	Oracle Applications Table/View Name	Column Name	Description
PRICING_CONTEXT			Pricing context. Not used for outbound transactions.
OPERATION_CODE			Operation Code. Not used for outbound transactions.

1.1.2 SALES_CREDITS

Document Field	Oracle Applications Table/View Name	Column Name	Description
HEADER_ID	SO_SALES_CREDITS	HEADER_ID	Order Header Identifier. The same as the HEADER_ID in Order.
LINE_ID	SO_SALES_CREDITS	LINE_ID	Order Line Identifier. Same as LINE_ID in Order Lines above.
ORIGINAL_SYSTEM_REFERENCE	SO_HEADERS_ALL	ORIGINAL_SYSTEM_REFERENCE	Original system reference for the order in external system.
ORIGINAL_SYSTEM_LINE_REFERENCE	SO_LINES_ALL	ORIGINAL_SYSTEM_LINE_REFERENCE	Line identifier from a source system outside of Oracle Order Management.
ORDER_SOURCE	SO_ORDER_SOURCES	NAME	Order source name.
SALESREP_NAME	RA_SALESREPS_ALL	NAME	Sales person name.
SALES_CREDIT_TYPE	SO_SALES_CREDIT_TYPES	NAME	Sales credit type.
PERCENT	SO_SALES_CREDITS	PERCENT	Indicates Sales Credit percentage for a sales person.
CONTEXT	SO_SALES_CREDITS	CONTEXT	Context.
OPERATION_CODE			Operation Code. Not used for outbound transactions.

1.1.3 DETAILS

Document Field	Oracle Applications Table/View Name	Column Name	Description
LINE_ID	SO_LINE_DETAILS	LINE_ID	Order Line Identifier. Same as LINE_ID in Order Lines above.
ORIGINAL_SYSTEM_REFERENCE	SO_HEADERS_ALL	ORIGINAL_SYSTEM_REFERENCE	Original system reference for the order in the external system.
ORIGINAL_SYSTEM_LINE_REFERENCE	SO_LINES_ALL	ORIGINAL_SYSTEM_LINE_REFERENCE	Line identifier from a source system outside of Oracle Order Management.
ORDER_SOURCE	SO_ORDER_SOURCES	NAME	Order source name.
QUANTITY	SO_LINE_DETAILS	QUANTITY	Quantity.
SCHEDULE_DATE	SO_LINE_DETAILS	SCHEDULE_DATE	Schedule date.
LOT_NUMBER	SO_LINE_DETAILS	LOT_NUMBER	Lot number.
SUBINVENTORY	SO_LINE_DETAILS	SUBINVENTORY	Subinventory.
CUSTOMER_REQUESTED_LOT_FLAG	SO_LINE_DETAILS	CUSTOMER_REQUESTED_LOT_FLAG	Indicator to customer requested lot.
CONTEXT	SO_LINE_DETAILS	CONTEXT	Context.
REVISION	SO_LINE_DETAILS	REVISION	Revision.
WAREHOUSE	MTL_PARAMETERS	ORGANIZATION_CODE	Warehouse code.

1.2 HEADER_PRICE_ADJ

Document Field	Oracle Applications Table/View Name	Column Name	Description
HEADER_ID	SO_PRICE_ADJUSTMENTS	HEADER_ID	Order Header Identifier. Same as HEADER_ID in Order.
ORIGINAL_SYSTEM_REFERENCE	SO_HEADERS_ALL	ORIGINAL_SYSTEM_REFERENCE	Original system reference for the order in external system.
ORDER_SOURCE	SO_ORDER_SOURCES	NAME	Order Source Name.
DISCOUNT_NAME	SO_DISCOUNTS	NAME	Discount Name.

Document Field	Oracle Applications Table/View Name	Column Name	Description
PERCENT	SO_PRICE_ADJUSTMENTS	PERCENT	Percentage.
CONTEXT	SO_PRICE_ADJUSTMENTS	CONTEXT	Context.
PRICING_CONTEXT			Pricing context.Not used for outbound transactions.
OPERATION_CODE			Operation code.Not used for outbound transactions.

1.3 HEADER_SALES_CREDIT

Document Field	Oracle Applications Table/View Name	Column Name	Description
HEADER_ID	SO_SALES_CREDITS	HEADER_ID	Order Header Identifier. Same as HEADER_ID in Order.
ORIGINAL_SYSTEM_REFERENCE	SO_HEADERS_ALL	ORIGINAL_SYSTEM_REFERENCE	Original system reference for the order in external system.
ORDER_SOURCE	SO_ORDER_SOURCES	NAME	Order source name.
SALESREP_NAME	RA_SALESREPS_ALL	NAME	Sales person name.
SALES_CREDIT_TYPE	SO_SALES_CREDIT_TYPES	NAME	Sales credit type.
PERCENT	SO_SALES_CREDITS	PERCENT	Value to indicate sales credit percent for a sales person.
CONTEXT	SO_SALES_CREDITS	CONTEXT	Context.
OPERATION_CODE			Operation Code. Not used for outbound transactions.

Procurement Predefined Transaction Services

■ Overview	410
■ Query Product Catalog Service	411
■ Query Purchase Order Service	418
■ Query Purchase Order Status Service	420
■ Query Supplier Service	424
■ Receive Requisition Service	425
■ Send Purchase Order Service	434
■ Send Supplier Service	451

Overview

This chapter describes the predefined transaction services provided in the Oracle Applications Adapter’s 10.7SC Procurement package.

The table below shows the predefined transaction services organized by Oracle Applications module. This chapter lists the transactions in alphabetical order.

Oracle Applications Module	Predefined Service
Purchasing	<ul style="list-style-type: none">■ “Query Product Catalog Service” on page 411■ “Query Purchase Order Service” on page 418■ “Query Purchase Order Status Service” on page 420■ “Query Supplier Service” on page 424■ “Receive Requisition Service” on page 425■ “Send Purchase Order Service” on page 434■ “Send Supplier Service” on page 451

For more information about using the predefined transaction services, see [Chapter 1, “Predefined Transaction Services”](#) on page 17.

Query Product Catalog Service

The name of this service is:

WmOAPRC107SC.purchasing107SC.queryOA.productCatalog:queryProductCatalog

This service retrieves the Supplier Product/Item Catalog information to locate items and their source information to create purchase order and requisition lines.

You can use the following parameters to query the Product Catalog:

- ITEM_CATEGORY: Item category.
- VENDOR_NAME: Vendor name.
- VENDOR_SITE: Vendor site code for an item.
- COMMODITY: Unique identifier for an item.
- ITEM_REVISION: Item revision code.
- ORGANIZATION_NAME: Organization name.

Database Scripts

This service uses the following database scripts:

Script	Description
wm_install_from_product_catalog.sql	Runs the scripts listed below, except the uninstall script.
wm_from_product_catalog_qry_vw.sql	Creates the following required view components: <ul style="list-style-type: none">■ WM_PO_PRODUCT_CATALOG_QRY_VW■ WM_PO_SOURCING_DOCS_VW■ WM_PO_SOURCING_RULES_VW■ WM_PO_PRIOR_PURCHASES_VW■ WM_PO_NEGOTIATED_SOURCES_VW
wm_drop_from_product_catalog.sql	Uninstalls all components created by wm_install_from_product_catalog.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service uses the following transaction definition:

- queryProductCatalogTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow queryProductCatalog executes as follows:

- specifyDefaultSettings specifies the default parameter settings required for service execution. You should change these settings accordingly.
- queryProductCatalogTxn service queries the Oracle Applications database for any Product Catalog matching the parameter values. The parameters are defined as the input to this service.

For more details on query transactions, see [“Using Query Services” on page 36](#).

Business Document Structure

This service uses the following business document structure:

- 1.0. SUPPLIER_ITEMS
 - 1.1. NEGOTIATED_SOURCES
 - 1.2. PRIOR_PURCHASES
 - 1.3. SOURCING_RULES
 - 1.3.1. SOURCING_DOCUMENTS

1.0. SUPPLIER_ITEMS

Document Field	Oracle Applications Table/View Name	Column Name	Description
WEB_TRANSACTION_ID		Not used.	Populated from a sequence and used internally in the Flow. Will contain NULL value for queried product catalog data.
DOCUMENT_TYPE		Not used.	Use PRODUCTCAT.
DOCUMENT_STATUS		Not used.	Use QUERY.

Document Field	Oracle Applications Table/View Name	Column Name	Description
ORG_ID			Organization ID.
ORGANIZATION_NAME			
ITEM_CATEGORY	MTL_CATEGORIES	SEGMENT1 ' SEGMENT2	The Category ID stored in the PO_LINES_ALL table joins with Category ID in MTL_CATEGORIES table.
ITEM_NUM	MTL_SYSTEM_ITEMS_KFV	CONCATENATE D_SEGMENTS	The Item ID stored in the PO_LINES_ALL table joins with Item ID in MTL_SYSTEM_ITEMS_KFV table.
VENDOR_NAME	PO_VENDORS	VENDOR_NAME	Supplier name.
VENDOR_SITE	PO_VENDOR_SITES_ALL	VENDOR_SITE_CODE	Supplier site.
ITEM_DESCRIPTION	PO_LINES_ALL	ITEM_DESCRIPTION	
ITEM_REVISION	PO_LINES_ALL	ITEM_REVISION	Item revision stored in PO_LINES_ALL.
SUPPLIER_ITEM	PO_LINES_ALL	VENDOR_PRODUCT_NUM	Supplier product number.

1.1. NEGOTIATED_SOURCES

Field Name	Oracle Applications Table/View Name	Column Name	Description
VENDOR_NAME	PO_VENDORS	VENDOR_NAME	Supplier name.
VENDOR_SITE	PO_VENDOR_SITES_ALL	VENDOR_SITE_CODE	Supplier site.
ITEM_NUM	MTL_SYSTEM_ITEMS_KFV	CONCATENATE D_SEGMENTS	Item ID stored in PO_LINES_ALL table joins with Item ID in MTL_SYSTEM_ITEMS_KFV table.

Field Name	Oracle Applications Table/View Name	Column Name	Description
ITEM_CATEGORY	MTL_CATEGORIES	SEGMENT1 ' ' SEGMENT2	Category ID stored in PO_LINES_ALL table joins with Category ID in MTL_CATEGORIES table.
ITEM_DESCRIPTION	PO_LINES_ALL	ITEM_DESCRIPTION	
ITEM_REVISION	PO_LINES_ALL	ITEM_REVISION	Item revision stored in PO_LINES_ALL.
LINE_UOM	PO_LINES_ALL	UNIT_MEAS_LOOKUP_CODE	Unit of measure.
LINE_PRICE	PO_LINES_ALL	UNIT_PRICE	Unit price for the line.
SUPPLIER_ITEM	PO_LINES_ALL	VENDOR_PRODUCT_NUM	Supplier product number.
BREAK_QUANTITY	PO_LINE_LOCATIONS_ALL	QUANTITY - NVL(QUANTITY_CANCELLED, 0)	Quantity ordered minus quantity cancelled.
BREAK_PRICE	PO_LINE_LOCATIONS_ALL	PRICE_OVERRIDE	Order shipment price or break price for blanket purchase orders, Requests For Quotation (RFQs), and quotations.

1.2. PRIOR_PURCHASES

Document Field	Oracle Applications Table/View Name	Column Name	Description
ORDER_DATE	PO_HEADERS_ALL PO_RELEASES_ALL	CREATION_DATE	IF TYPE_LOOKUP_CODE = Standard then use CREATION_DATE from the PO_HEADERS_ALL table. For TYPE_LOOKUP_CODE = Planned or Blanket, then use the CREATION_DATE column in the table PO_RELEASES_ALL.
VENDOR_NAME	PO_VENDORS	VENDOR_NAME	Supplier name.
VENDOR_SITE	PO_VENDOR_SITES_ALL	VENDOR_SITE_CODE	Supplier site.
ITEM_NUM	MTL_SYSTEM_ITEMS_KFV	CONCATENATE D_SEGMENTS	Item ID stored in PO_LINES_ALL table joins with Item ID in MTL_SYSTEM_ITEMS_KFV table.
ITEM_CATEGORY	MTL_CATEGORIES	SEGMENT1 ' ' SEGMENT2	Category ID stored in PO_LINES_ALL table joins with Category ID in MTL_CATEGORIES table.
ITEM_DESCRIPTION	PO_LINES_ALL	ITEM_DESCRIPTION	
ITEM_REVISION	PO_LINES_ALL	ITEM_REVISION	Item revision stored in PO_LINES_ALL.
UOM	PO_LINES_ALL	UNIT_MEAS_LOOKUP_CODE	Unit of measure.
SUPPLIER_ITEM	PO_LINES_ALL	VENDOR_PRODUCT_NUM	Supplier product number.

Document Field	Oracle Applications Table/View Name	Column Name	Description
QUANTITY	PO_LINE_LOCATIONS_ALL	QUANTITY - NVL(QUANTITY_CANCELLED, 0)	Quantity ordered minus quantity cancelled.
PRICE	PO_LINES_ALL/ PO_LINE_LOCATIONS_ALL	PO_LINES_ALL.UNIT_PRICE/PO_LINE_LOCATIONS_ALL.PRICE_OVERRIDE	Unit price for standard PO, or the price override for a planned or blanket PO.

1.3. SOURCING_RULES

Document Field	Oracle Applications Table/View Name	Column Name	Description
ORG_ID	MRP_SOURCES_V	ORGANIZATION_ID	Organization identifier.
RULE	MRP_SOURCES_V	SOURCING_RULE_NAME	Rule name.
RULE_ID	MRP_SOURCES_V	SOURCING_RULE_ID	Rule identifier.
ITEM_NUM	MTL_SYSTEM_ITEMS_KFV	CONCATENATED_SEGMENTS	Item ID stored in PO_LINES_ALL table joins with Item ID in MTL_SYSTEM_ITEMS_KFV table.
ITEM_DESCRIPTION	PO_LINES_ALL	ITEM_DESCRIPTION	
FROM_DATE	MRP_SOURCES_V	EFFECTIVE_DATE	
TO_DATE	MRP_SOURCES_V	DISABLE_DATE	
VENDOR_NAME	PO_VENDORS	VENDOR_NAME	Supplier name.
VENDOR_SITE	PO_VENDOR_SITES_ALL	VENDOR_SITE_CODE	Supplier site.

Document Field	Oracle Applications Table/View Name	Column Name	Description
ORGANIZATION_CODE	MTL_PARAMETERS	ORGANIZATION_CODE	
VENDOR_ID	PO_VENDORS	VENDOR_ID	

1.3.1. SOURCING_DOCUMENTS

Field Name	Oracle Applications Table/View Name	Column Name	Description
ITEM_NUM	MTL_SYSTEM_ITEMS_KFV	CONCATENATED_SEGMENTS	Item number.
ITEM_CATEGORY	MTL_CATEGORIES	SEGMENT1 ' ' SEGMENT2	Category ID stored in PO_LINES_ALL table joins with Category ID in MTL_CATEGORIES table.
VENDOR_ID	PO_VENDORS	VENDOR_ID	Vendor identifier.
RULE_ID	PO_ASL_DOCUMENTS	ASL_ID	Rule identifier.
SEQUENCE_NUM	PO_ASL_DOCUMENTS	SEQUENCE_NUM	Sequence number.
VENDOR_PRODUCT_NUM	PO_LINES_ALL	VENDOR_PRODUCT_NUM	Supplier product number.
LINE_UOM	PO_LINES_ALL	UNIT_MEAS_LOOKUP_CODE	Unit of measure.
LINE_PRICE	PO_LINES_ALL	UNIT_PRICE	Unit price for the line.
BREAK_QUANTITY	PO_LINE_LOCATIONS_ALL	QUANTITY - NVL(QUANTITY_CANCELLED, 0)	Quantity ordered minus quantity cancelled.
BREAK_PRICE	PO_LINE_LOCATIONS_ALL	PRICE_OVERRIDE	Order shipment price or break price for blanket purchase orders, Requests For Quotation (RFQs), and quotations.

Query Purchase Order Service

The name of this service is:

WmOAPRC107SC.purchasing107SC.queryOA.PO:queryPO

This service retrieves PO data.

The following parameters are used for querying PO data:

- AGENT_NAME
- DOCUMENT_NUM
- SHIP_TO_ADDRESS_LINE_1, SHIP_TO_ADDRESS_LINE_2, SHIP_TO_ADDRESS_LINE_3
- SHIP_TO_TOWN_OR_CITY
- SHIP_TO_REGION_1, SHIP_TO_REGION_2, SHIP_TO_REGION_3
- SHIP_TO_COUNTRY
- SHIP_TO_POSTAL_CODE
- BILL_TO_ADDRESS_LINE_1, BILL_TO_ADDRESS_LINE_2, BILL_TO_ADDRESS_LINE_3
- BILL_TO_TOWN_OR_CITY
- BILL_TO_REGION_1, BILL_TO_REGION_2, BILL_TO_REGION_3
- BILL_TO_COUNTRY
- BILL_TO_POSTAL_CODE
- ADDRESS_LINE1, ADDRESS_LINE2, ADDRESS_LINE3
- CITY
- STATE
- ZIP
- COUNTRY
- VENDOR_NAME
- ORGANIZATION_NAME

Database Scripts

This service uses the same database scripts as the Send Purchase Order service.



Note: If you use this service but you do *not* use the Send Purchase Order service, you should run the `wm_disable_from_po.sql` script to disable the triggers installed by the Send Purchase Order service.

For a detailed description of these database scripts, see [“Send Purchase Order Service” on page 434](#). For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service uses the following transaction definition:

- `queryPOTxn107SC.txp`

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow `queryPO` executes as follows:

- `specifyDefaultSettings` specifies the default parameter settings required for service execution. You should change these settings accordingly.
- `queryPOTxn` queries the Oracle Applications database for any purchase order information matching the parameter values. The parameters are defined as the input to this service.

For more details on query transactions, see [“Using Query Services” on page 36](#).

Business Document Structure

This service uses the same business document structure as the Send Purchase Order service. For a detailed description of the business document’s structure, see [“Send Purchase Order Service” on page 434](#).

Query Purchase Order Status Service

The name of this service is:

WmOAPRC107SC.purchasing107SC.queryOA.POStatus:queryPOStatus

This service retrieves the purchase order status based on the following parameters:

- BUYER_NAME
- PO_DOCUMENT_NUM
- ORGANIZATION_NAME
- SHIP_TO_ADDRESS_LINE_1, SHIP_TO_ADDRESS_LINE_2, SHIP_TO_ADDRESS_LINE_3
- SHIP_TO_TOWN_OR_CITY
- SHIP_TO_REGION_1, SHIP_TO_REGION_2, SHIP_TO_REGION_3
- SHIP_TO_COUNTRY
- SHIP_TO_POSTAL_CODE
- BILL_TO_ADDRESS_LINE_1, BILL_TO_ADDRESS_LINE_2, BILL_TO_ADDRESS_LINE_3
- BILL_TO_TOWN_OR_CITY
- BILL_TO_REGION_1, BILL_TO_REGION_2, BILL_TO_REGION_3
- BILL_TO_COUNTRY
- BILL_TO_POSTAL_CODE
- SUPPLIER_ADDRESS_LINE1, SUPPLIER_ADDRESS_LINE2, SUPPLIER_ADDRESS_LINE3
- SUPPLIER_CITY
- SUPPLIER_STATE
- SUPPLIER_ZIP
- SUPPLIER_COUNTRY
- SUPPLIER_NAME

Database Scripts

This service uses the following database scripts:

Script	Description
wm_install_from_orderstatus.sql	Runs the scripts listed below, except the uninstall script.
wm_from_orderstatus_vw.sql	Creates the required view component WM_OS_QRY_VW.
wm_from_orderstatus_pkg.sql	Installs the WM_ORDER_STATUS_PKG.get_po_status procedure to retrieve the Purchase Order Status.
wm_drop_from_orderstatus.sql	Uninstalls all components created by wm_install_from_orderstatus.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service uses the following transaction definition:

- queryPOStatusTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

You can query order status data using the following services:

- specifyDefaultSettings specifies the default parameter settings required for service execution. You should change these settings accordingly.
- queryOrderStatusTxn queries the Oracle Applications database for any Order Status matching the parameter values. The parameters are defined as the input to this service.

For more details on query transactions, see [“Using Query Services” on page 36](#).

Business Document Structure

This service uses the following business structure:

■ PURCHASE_ORDER

PURCHASE_ORDER

Document Field	Oracle Applications Table/View Name	Column Name	Description
PO_DOCUMENT_NUM	PO_HEADERS_ALL	SEGMENT1	Purchase order number.
REVISION_NUM	PO_HEADERS_ALL	REVISION_NUM	Purchase order revision number.
TYPE_LOOKUP_CODE	PO_HEADERS_ALL	TYPE_LOOKUP_CODE	Required. Use STANDARD, BLANKET, PLANNED, or CONTRACT. Maps directly.
CREATION_DATE	PO_HEADERS_ALL	CREATION_DATE	Purchase order creation date.
ORGANIZATION_NAME	PO_HEADERS_ALL	ORG_ID	Organization name for which the purchase order is generated.
SUPPLIER_NAME	PO_HEADERS_ALL	VENDOR_ID	Supplier name.
SUPPLIER_SITE_CODE	PO_HEADERS_ALL	VENDOR_SITE_ID	Supplier site code.
SUPPLIER_CONTACT	PO_HEADERS_ALL	VENDOR_CONTACT_ID	Supplier contact name.
CURRENCY_CODE	PO_HEADERS_ALL	CURRENCY_CODE	
AGENT_NAME	PO_HEADERS_ALL	AGENT_ID	Buyer name.
PO_STATUS	PO_HEADERS_ALL	STATUS_LOOKUP_CODE	Purchase order status.
BLANKET_TOTAL_AMOUNT	PO_HEADERS_ALL	BLANKET_TOTAL_AMOUNT	Purchase order amount.
COMMENTS	PO_HEADERS_ALL	COMMENTS	

Document Field	Oracle Applications Table/View Name	Column Name	Description
SHIP_TO_ADDRESS_LINE_1 SHIP_TO_ADDRESS_LINE_2 SHIP_TO_ADDRESS_LINE_3 SHIP_TO_TOWN_OR_CITY SHIP_TO_REGION_1 SHIP_TO_REGION_2 SHIP_TO_REGION_3 SHIP_TO_COUNTRY SHIP_TO_POSTAL_CODE	PO_HEADERS_ ALL	SHIP_TO_ LOCATION_ID	Shipping location details.
BILL_TO_ADDRESS_LINE_1 BILL_TO_ADDRESS_LINE_2 BILL_TO_ADDRESS_LINE_3 BILL_TO_TOWN_OR_CITY BILL_TO_REGION_1 BILL_TO_REGION_2 BILL_TO_REGION_3 BILL_TO_COUNTRY BILL_TO_POSTAL_CODE	PO_HEADERS_ ALL	BILL_TO_ LOCATION_ID	Billing location details.
SUPPLIER_ADDRESS_LINE1 SUPPLIER_ADDRESS_LINE2 SUPPLIER_ADDRESS_LINE3 SUPPLIER_CITY SUPPLIER_STATE SUPPLIER_ZIP SUPPLIER_COUNTRY	PO_HEADERS_ ALL	VENDOR_SITE_ ID	Purchase order vendor site details.

Query Supplier Service

The name of this service is:

WmOAPRC107SC.purchasing107SC.queryOA.supplier:querySupplier

This service retrieves active supplier data. It extracts information for a supplier only, and not for individual employees who are also set up as suppliers. Because suppliers cannot be deleted, the document status of DELETE does not apply in this case.

You can use the following parameters to query supplier data:

- SUPPLIER_NUMBER: Unique supplier identifier in Oracle Applications that is assigned to the supplier upon creation.
- SUPPLIER_SITE_CODE: Unique site identifier for a supplier. The same site name can exist for multiple suppliers.
- INACTIVE_DATE_FROM: Beginning date when the supplier is inactive.
- INACTIVE_DATE_TO: Ending date when the supplier is inactive.
- TAXPAYER_ID: Supplier taxpayer ID.
- TAX_REGISTRATION_NUMBER: Tax registration number.
- VENDOR_NAME: Vendor name.
- ORGANIZATION_NAME: Organization name.

Database Scripts

This service uses the same database scripts as the Send Supplier service.



Note: If you use this service but you do *not* use the Send Supplier service, you should run the `wm_disable_from_supplier.sql` script to disable the triggers installed by the Send Supplier service.

For a detailed description of these database scripts, see [“Send Supplier Service” on page 451](#). For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service uses the following transaction definition:

- `getSupplierTxn107SC.txp`

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow querySupplier executes as follows:

- `specifyDefaultSettings` specifies the default parameter settings required for service execution. You should change these settings accordingly.
- `querySupplierTxn` queries the Oracle Applications database for any supplier matching the parameter values. The parameters are defined as the input to this service.

For more details on query transactions, see [“Using Query Services” on page 36](#).

Business Document Structure

This service uses the same business document structure as the Send Supplier service. For a detailed description of the business document’s structure, see [“Send Supplier Service” on page 451](#).

Receive Requisition Service

The name of this service is:

`WmOAPRC107SC.purchasing107SC.intoOA.requisition:receiveRequisition`

This service loads requisition documents into Oracle Applications production tables. You can load multiple types of requisition documents, such as Internal and Purchase. This service handles multiple requisition records in a single document, and a single document can contain any combination of requisition types.

Each record in the business document is inserted, updated, or cancelled according to the production data. The actual business operation (such as INSERT or UPDATE) is determined by the underlying Oracle Applications APIs and the data in the individual columns of the business document.

Data is loaded into the interface tables for both single and multiple distributions. The custom API calls the Oracle Applications API with a default parameter for multiple distributions.

While submitting the Import Requisition concurrent program to load the requisition records from interface tables to production tables, it is not possible to restrict the processing to only those records uploaded by the IS flow instance. The program will process all unprocessed records and error records from both the previous load and the interface table at the time of execution. While showing errors, it might display an error related to previously failed records, even if the current import process was successful. It is suggested to keep the interface table free of error records to minimize ambiguity.

Database Scripts

This service uses the following database scripts:

Script	Description
wm_install_into_req.sql	Runs the scripts listed below, except the uninstall script.
wm_into_req_pkg.sql	Installs WM_REQ_IMP_HANDLER_PKG.WM_HANDLE_REQ, which calls the WM_CONC_REQUEST.WM_REQUEST_SUBMIT procedure to submit the Requisition import process.
wm_into_req_seq.sql	Creates the WM_REQ_BATCH_S component, which creates the Req_Batch_sequence_id sequence.
wm_drop_into_req.sql	Uninstalls all components created by wm_install_into_req.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definitions:

- setRequisitionTxn107SC.txp
- RequisitionTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

This service calls the following services to import inventory transactions:

- **specifyDefaultSettings** specifies the default parameter settings required for service execution. You should change these settings accordingly.
- **specifyConcProgParams** specifies the default parameter settings required for concurrent program execution. You should change these settings accordingly.
- **pickSequence** generates sequence values. This flow are sequence values for header sequence, line sequence, distribution sequence, and custom batch sequence.
- **bizDocMapping** maps the incoming business data structure to the required data structure, which is similar to the interface table. This service internally uses the following services to provide required data feed during the mapping.

- **getOrgId** is used as a transformer while mapping the business doc Idata structure to the interface table Idata structure. It takes ORGANIZATION_NAME as the input parameter, queries the table ORG_ORGANIZATION_DEFINITIONS, and gets the ORGANIZATION_ID corresponding to the ORGANIZATION_NAME.
- **getVendorSiteId** is a transformer while mapping the business doc Idata structure to the interface table Idata structure. It takes ADDRESS_LINE1, ADDRESS_LINE2, ADDRESS_LINE3, CITY, STATE, ZIP, and COUNTRY as the input parameters, queries the table PO_VENDOR_SITES_ALL, and gets the VENDOR_SITE_ID corresponding to the passed values.
- **getCategoryId** is a transformer while mapping the business doc Idata structure to the interface table Idata structure. It takes CATEGORY as the input parameter, and returns the CATEGORY_ID by querying MTL_CATEGORIES_KFV table.
- **getItemNumber** is a transformer while mapping the business doc Idata structure to the interface table Idata structure. It takes ITEM as the input parameter, and returns the ITEM_NUMBER by querying MTL_SYSTEM_ITEMS table.
- **getWIPEntityId** is a transformer while mapping the business doc Idata structure to the interface table Idata structure. It takes WIP_ENTITY_NAME as the input parameters, and gets the WIP_ENTITY_ID by querying WIP_ENTITIES table. The WIP_ENTITY_ID returned is specific to the ORGANIZATION_NAME specified in the business document.
- **getWIPLineId** is a transformer while mapping the business doc Idata structure to the interface table Idata structure. It takes WIP_LINE_CODE as the input parameters, and gets the WIP_LINE_ID corresponding to the input values by querying WIP_LINES table.
- **getCodeCombinationId** is a transformer while mapping the business doc Idata structure to the interface table Idata structure. It takes CONCATINATED_SEGMENTS and SET_OF_BOOKS_NAME as the input parameters, and gets the CODE_COMBINATION_ID corresponding to the input values. This service is used in multiple places in bizDocMapping.
- **getInventoryItemId** returns the INVENTORY_ITEM_ID corresponding to the supplied ITEM_CODE and the ORGANIZATION_NAME. This service is used as a transformer in bizDocMapping.
- **getLocationId** is a transformer while mapping the business doc Idata structure to the interface table Idata structure. This is a general service to get the LOCATION_ID from HR_LOCATIONS, given the ADDRESS_LINE_1, ADDRESS_LINE_2, ADDRESS_LINE_3, TOWN_OR_CITY, COUNTRY, POSTAL_CODE, REGION_1, and REGION_2.
- **getProjectAndTaskId** is a transformer while mapping the business doc Idata structure to the interface table Idata structure. It takes Project Name and Task Name as input, and returns Project ID and Task Id.

- **getResourceId** is a transformer while mapping the business doc Idata structure to the interface table Idata structure. It takes BOM_RESOURCE_NAME as input and returns BOM_RESOURCE_ID.
- **convertToDateObject** returns the outDate as a date object corresponding to the supplied inDate, which is in text format. This service is used as a transformer in bizDocMapping.
- **setRequisitionTxn** inserts data into the interface table. It takes data from the Idata structure resulted in the bizDocMapping service and puts the data into the interface tables, namely PO_REQUISITIONS_INTERFACE_ALL in Oracle Applications.
- **importRequisition** imports data to the production table and calls following services:
 - **execRequisitionConcProg** calls the underlying OracleApps concurrent program with the set of parameters set in the previous MAP, and returns the V_REQUEST_ID, V_STATUS, O_MESSAGE and V_ERRMSG if errors occur in the process.
 - **checkRequisitionImportStatus** checks the status of the concurrent program execution by checking the error Interface Table for any rejected record corresponding to the current REQUEST_ID. If the query does not return any rows, it indicates a successful import. If the query returns a row, it indicates that the concurrent program could not import data successfully in the production tables of Oracle Applications.
 - **getRequisitionImport_ERR** gets the data import error for the specific REQUEST ID. This service takes REQUEST_ID as an input, retrieves data import error from PO_INTERFACE_ERRORS table, and appends the errorsDoc record list.

For more details on receive transactions, see [“Overview of Receive Service Transaction Processing” on page 31](#).

Business Document Structure

This service uses the business document requisitionBizDoc. Its structure is as follows:

- 1.0. PO_REQUISITIONS
 - 1.1. PO_REQ_LINES
 - 1.1.1. PO_REQ_DISTRIBUTIONS

1.0. PO_REQUISITIONS (Maps to PO_REQUISITIONS_INTERFACE_ALL)

Field Name	Maps to Column	Description
INTERFACE_SOURCE_CODE	INTERFACE_SOURCE_CODE	Required. Use values such as ICX, INV, MRP, MSC, ORDER ENTRY, WIP
REQUISITION_TYPE	REQUISITION_TYPE	Use values such as BLANKET, PLANNED, SCHEDULED, STANDARD, INTERNAL, EXPENSE
AUTHORIZATION_STATUS	AUTHORIZATION_STATUS	Required. Use values such as APPROVED, CANCELLED, IN PROCESS, INCOMPLETE, PRE-APPROVED, REJECTED, REQUIRES REAPPROVAL, RETURNED
PREPARER_NAME	PREPARER_NAME	Required. Name of the person who prepared the requisition
APPROVER_NAME	APPROVER_NAME	Name of the person who approved the requisition
HEADER_DESCRIPTION	HEADER_DESCRIPTION	Description of the requisition
NOTE_TO_APPROVER	NOTE_TO_APPROVER	
USSGL_TRANSACTION_CODE	USSGL_TRANSACTION_CODE	United States standard general ledger transaction code
ORGANIZATION_NAME	ORG_ID	Organization for which a requisition has been raised.
SET_OF_BOOKS_NAME		Required for retrieving the Account ID using the transformer. See the commonOA107SC.utils: getCodeCombinationId transformer.

1.1. PO_REQ_LINES (Maps to PO_REQUISITIONS_INTERFACE_ALL)

Field Name	Maps to Column	Description
DESTINATION_TYPE_CODE	DESTINATION_TYPE_CODE	Required. Requisition destination type. Values can be EXPENSES, INVENTORY, or SHOP FLOOR.
SOURCE_TYPE_CODE	SOURCE_TYPE_CODE	Conditionally required. Requisition source. Can have values such as INVENTORY or VENDOR.
URGENT_FLAG	URGENT_FLAG	Indicates whether requisition is urgent.
RFQ_REQUIRED_FLAG	RFQ_REQUIRED_FLAG	Indicates whether a Request For Quotation (RFQ) is required prior to placement on a purchase order.
JUSTIFICATION	JUSTIFICATION	Purchase justification.
NOTE_TO_BUYER	NOTE_TO_BUYER	
NOTE_TO_RECEIVER	NOTE_TO_RECEIVER	
ITEM	ITEM_ID	Conditionally required. Item for which requisition is placed.
ITEM_DESCRIPTION	ITEM_DESCRIPTION	
ITEM_REVISION	ITEM_REVISION	
CATEGORY	CATEGORY_ID	
QUANTITY	QUANTITY	
UNIT_PRICE	UNIT_PRICE	
UNIT_OF_MEASURE	UNIT_OF_MEASURE	
UOM_CODE	UOM_CODE	Unit of measurement code.
LINE_TYPE	LINE_TYPE	Requisition line type.
UN_NUMBER	UN_NUMBER	
HAZARD_CLASS	HAZARD_CLASS	
REFERENCE_NUM	REFERENCE_NUM	Reference number.
SOURCE_ORGANIZATION_NAME	SOURCE_ORGANIZATION_ID	Conditionally required.
SOURCE_SUBINVENTORY	SOURCE_SUBINVENTORY	Conditionally required.

Field Name	Maps to Column	Description
DESTINATION_ORGANIZATION_NAME	DESTINATION_ORGANIZATION_ID	
DESTINATION_SUBINVENTORY	DESTINATION_SUBINVENTORY	
DELIVER_TO_LOCATION_CODE	DELIVER_TO_LOCATION_CODE	
DELIVER_TO_LOCATION_ADDRESS_LINE_1 DELIVER_TO_LOCATION_ADDRESS_LINE_2 DELIVER_TO_LOCATION_ADDRESS_LINE_3 DELIVER_TO_LOCATION_TOWN_OR_CITY DELIVER_TO_LOCATION_COUNTY DELIVER_TO_LOCATION_STATE DELIVER_TO_LOCATION_POSTAL_CODE DELIVER_TO_LOCATION_COUNTRY	DELIVER_TO_LOCATION_ID	Required. Address for the delivery location.
DELIVER_TO_REQUESTOR_NAME	DELIVER_TO_REQUESTOR_NAME	Conditionally required.
AUTOSOURCE_FLAG	AUTOSOURCE_FLAG	Conditionally required. Value validated against PO_AUTOSOURCE_DOCUMENTS_ALL.
AUTOSOURCE_DOC_LINE_NUMER	AUTOSOURCE_DOC_LINE_NUM	Conditionally required.
DOCUMENT_TYPE_CODE	DOCUMENT_TYPE_CODE	Conditionally required.
SUGGESTED_BUYER_NAME	SUGGESTED_BUYER_NAME	Conditionally required.
SUGGESTED_VENDOR_NAME	SUGGESTED_VENDOR_NAME	Conditionally required.

Field Name	Maps to Column	Description
SUGGESTED_VENDOR_SITE_ADDRESS_1	SUGGESTED_VENDOR_SITE_ID	Conditionally required. Supplier's address
SUGGESTED_VENDOR_SITE_ADDRESS_2		
SUGGESTED_VENDOR_SITE_ADDRESS_3		
SUGGESTED_VENDOR_SITE_TOWN_OR_CITY		
SUGGESTED_VENDOR_SITE_COUNTY		
SUGGESTED_VENDOR_SITE_STATE		
SUGGESTED_VENDOR_SITE_POSTAL_CODE		
SUGGESTED_VENDOR_SITE_COUNTRY		
SUGGESTED_VENDOR_PHONE	SUGGESTED_VENDOR_PHONE	Conditionally required. Supplier's phone.
SUGGESTED_VENDOR_CONTACT	SUGGESTED_VENDOR_CONTACT	Conditionally required. Supplier's contact person.
SUGGESTED_VENDOR_ITEM	SUGGESTED_VENDOR_ITEM_NUM	Supplier's item.
NEED_BY_DATE	NEED_BY_DATE	Required. Date by which the item is needed.
CURRENCY_CODE	CURRENCY_CODE	
CURRENCY_UNIT_PRICE	CURRENCY_UNIT_PRICE	Conditionally required.
RATE	RATE	Conditionally required.
RATE_DATE	RATE_DATE	Conditionally required.
RATE_TYPE	RATE_TYPE	Conditionally required.
WIP_ENTITY_NAME	WIP_ENTITY_ID	Conditionally required.
WIP_LINE_CODE	WIP_LINE_ID	
WIP_OPERATION_SEQ_NUM	WIP_OPERATION_SEQ_NUM	WIP operation sequence number within a routing.
WIP_RESOURCE_SEQ_NUM	WIP_RESOURCE_SEQ_NUM	WIP resource sequence number.

Field Name	Maps to Column	Description
WIP_REPETITIVE_SCHEDULE	WIP_REPETITIVE_SCHEDULE_ID	Conditionally required.
BOM_RESOURCE	BOM_RESOURCE_ID	Conditionally required. Bill of materials resource.
TRANSACTION_REASON_CODE	TRANSACTION_REASON_CODE	

1.1.1. PO_REQ_DISTRIBUTIONS (Maps to PO_REQUISITIONS_INTERFACE_ALL)

Field Name	Maps to Column	Description
PROJECT_ACCOUNTING_CONTEXT	PROJECT_ACCOUNTING_CONTEXT	Indicates whether project accounting is required.
EXPENDITURE_ORGANIZATION_NAME	EXPENDITURE_ORGANIZATION_ID	Conditionally required. Project accounting expenditure organization.
PROJECT_NAME	PROJECT_ID	Conditionally required. Project accounting project.
TASK_NAME	TASK_ID	Conditionally required. Project accounting task.
EXPENDITURE_ITEM_DATE	EXPENDITURE_ITEM_DATE	Project accounting expenditure date.
GL_DATE	GL_DATE	General Ledger date.
EXPENDITURE_TYPE	EXPENDITURE_TYPE	Conditionally required.
QUANTITY	QUANTITY	Required. Quantity to be distributed.
CHARGE_ACCOUNT	CHARGE_ACCOUNT_ID	Required. General Ledger charge account number.
ACCRUAL_ACCOUNT	ACCRUAL_ACCOUNT_ID	Conditionally required. General ledger accrual account number.
VARIANCE_ACCOUNT	VARIANCE_ACCOUNT_ID	Conditionally required. General Ledger variance account number.
BUDGET_ACCOUNT	BUDGET_ACCOUNT_ID	Conditionally required. General Ledger budget account number.

Send Purchase Order Service

The name of this service is:

WmOAPRC107SC.purchasing107SC.fromOA.PO:sendPO

This service retrieves all new or changed *approved* PO data, and presents it in a logical structure (a business document) that can be sent to a trading partner.

POs, once defined in Oracle Applications, cannot be deleted. Thus, business documents with the document status of INSERT or UPDATE (but not DELETE) will be created.

Approved POs may be updated and re-approved, then closed or cancelled. The Document Status, Status (PO Status), and Authorization Status defines the actual status of the PO at the polling instance. The decision to transfer the PO to a trading partner or to an internal organization must be made at implementation, based on the three statuses.

For example: A PO is created, approved, and then cancelled. One record of INSERT and one record of UPDATE are added to the Track Changes table. The sendPO service will retrieve the business document that has Authorization_status = APPROVED, Status = APPROVED, CLOSED, CANCELLED, and Transaction_status = INSERT. This document cannot not be sent to the trading partner.

Database Scripts

This service uses the following database scripts:

Script	Description
wm_install_from_po.sql	Runs the scripts listed below, except the uninstall script.
wm_from_po_vw.sql	Creates the following required view components for PO outbound transactions: <ul style="list-style-type: none"> ■ WM_PO_DISTRIBUTIONS_VW ■ WM_PO_LINE_LOCATIONS_VW ■ WM_PO_LINES_VW ■ WM_PO_HEADERS_VW ■ WM_PO_QRY_VW
wm_from_po_trg.sql	Creates the following trigger component to be used for writing to the WM_TRACKCHANGES table to indicate a new or updated document: <ul style="list-style-type: none"> ■ WM_PO_HEADERS_ALL_IU_TRG
wm_disable_from_po.sql	Disables the triggers installed by wm_from_po_trg.sql.

Script	Description
wm_enable_from_po.sql	Re-enables the triggers installed by wm_from_po_trg.sql.
wm_drop_from_po.sql	Uninstalls all components created by wm_install_from_po.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definition:

- getPOTxn107SC.txp

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow sendPO executes as follows:

- **specifyDefaultSettings** specifies the default parameter settings required for service execution. You should change these settings accordingly.
- **lockTxnCtrl** determines whether the sendPO service is ready for execution. That is, it determines whether there is no other instance of this service already in process.
 - If **lockTxnCtrl** returns False, it indicates that another instance of this service is already in process. The service exits and waits for next scheduled execution.
 - If **lockTxnCtrl** returns True, it indicates that the service is ready to execute. The PO row in the control table is locked and updated so that the status is changed to be INPROCESS. This prevents any other PO service from executing.
- **getPOTxn** queries the Oracle Applications database for any PO Transaction to be processed. The number of records returned depends on the parameter value specified in the above map.
- **processBizDoc** is the customizable step that sends the business document to the trading partner by looping against each document. This step needs to be customized to receive a SUCCESS or an ERROR status of the document transfer along with the error information. The transfer status and any error information are logged against each document.

- Loops against each document. Based on the Debug Mode specified during execution, it either purges or updates the records in the WM_TRACKCHANGES custom table.
 - If Debug Mode is TRUE, the records in the WM_TRACKCHANGES table are updated, and the PROCESSED_FLAG is set to Y. This ensures that the same sets of records are not picked up during next polling interval. The `updateTrackChanges` service updates the PROCESSED_FLAG in the WM_TRACKCHANGES table to Y, and updates Processed_Date to sysdate so that the same information is not picked up again during next polling instance.
 - If Debug Mode is FALSE, the records in the WM_TRACKCHANGES table are deleted. The `purgeTrackChanges` service purges the records from the WM_TRACKCHANGES table.
- Loops against each document. Based on the transfer status, the `insertTransferERRInfo` service is used to insert a new record into the WM_TRACKCHANGES table so that the same document can be picked up during the next polling interval.
- `unlockTxnCtrl` releases the lock on the Custom Control table so that the next polling instance of `sendVendor` service can begin.
- Any errors that occur in the above steps are logged using the `getLastError` service. The `unlockTxnCtrl` service executes to release the lock on the Custom Control table.

For more details on send transactions, see [“Overview of Send Service Transaction Processing” on page 38](#).

Business Document Structure

This service uses the following business document structure:

- 1.0. PO_HEADERS
 - 1.1. PO_LINES
 - 1.1.1. PO_LINE_LOCATIONS
 - 1.1.1.1. PO_DISTRIBUTIONS

1.0. PO_HEADERS

Field Name	Oracle Applications Table/View Name	Column Name	Description
WEB_TRANSACTION_ID			
DOCUMENT_TYPE			Use PO.
DOCUMENT_STATUS			Use INSERT or UPDATE.
ORGANIZATION_NAME	HR_ORGANIZATION_UNITS	NAME	Organization name for which the PO is generated.

Field Name	Oracle Applications Table/View Name	Column Name	Description
DOC_SUBTYPE	PO_DOCUMENT_TYPES_ALL	DOCUMENT_SUBTYPE	PO document subtype.
DOCUMENT_NUM	PO_HEADERS_ALL	SEGMENT1	PO number.
REVISION_NUM	PO_HEADERS_ALL	REVISION_NUM	PO revision number.
CREATION_DATE	PO_HEADERS_ALL	CREATION_DATE	Date PO was created.
VENDOR_NAME	PO_VENDORS	VENDOR_NAME	Supplier name.
STATUS	PO_HEADERS_ALL	AUTHORIZATION_STATUS	Status of the PO.
AUTHORIZATION_STATUS	PO_HEADERS_ALL	AUTHORIZATION_STATUS	Approval status of the PO.
COMMENTS	PO_HEADERS_ALL	COMMENTS	Comments on the PO.
ACCEPTANCE_REQUIRED_FLAG	PO_HEADERS_ALL	ACCEPTANCE_REQUIRED_FLAG	Indicates whether acceptance from the supplier is required.
ACCEPTANCE_DUE_DATE	PO_HEADERS_ALL	ACCEPTANCE_DUE_DATE	Last date when the supplier should accept the PO.
FIRM_STATUS_LOOKUP_CODE	PO_HEADERS_ALL	FIRM_STATUS_LOOKUP_CODE	Indicates whether the PO should be prevented from being automatically rescheduled by the manufacturing application.
FROZEN_FLAG	PO_HEADERS_ALL	FROZEN_FLAG	Indicates whether the PO is frozen.
AMOUNT_LIMIT	PO_HEADERS_ALL	AMOUNT_LIMIT	Maximum amount that can be released against the PO.
MIN_RELEASE_AMOUNT	PO_HEADERS_ALL	MIN_RELEASE_AMOUNT	Minimum amount that can be released against the PO.
APPROVAL_REQUIRED_FLAG	PO_HEADERS_ALL	APPROVAL_REQUIRED_FLAG	Indicates whether a PO requires approval.

Field Name	Oracle Applications Table/View Name	Column Name	Description
CLOSED_CODE	PO_HEADERS_ALL	CLOSED_CODE	Indicates whether the PO has a closed status.
SHIP_VIA_LOOKUP_CODE	PO_HEADERS_ALL	SHIP_VIA_LOOKUP_CODE	Freight carrier.
FREIGHT_TERMS_LOOKUP_CODE	PO_HEADERS_ALL	FREIGHT_TERMS_LOOKUP_CODE	Type of freight terms for the PO.
RATE_TYPE	PO_HEADERS_ALL	RATE_TYPE	Rate type for the PO.
RATE	PO_HEADERS_ALL	RATE	Rate applicable for the PO.
FROM_TYPE_LOOKUP_CODE	PO_HEADERS_ALL	FROM_TYPE_LOOKUP_CODE	PO used to auto create.
END_DATE	PO_HEADERS_ALL	END_DATE	Expiration date of the document.
REVISED_DATE	PO_HEADERS_ALL	REVISED_DATE	Date when the PO was revised.
NOTE_TO_VENDOR	PO_HEADERS_ALL	NOTE_TO_VENDOR	Note sent to the vendor.
PRINT_COUNT	PO_HEADERS_ALL	PRINT_COUNT	Print count.
REPLY_DATE	PO_HEADERS_ALL	REPLY_DATE	Date when the supplier replied to the PO.
SHIP_TO_ADDRESS_LINE_1	HR_LOCATIONS	ADDRESS_LINE_1	Address that will receive the PO items. Derives the SHIP_TO_LOCATION ID.
SHIP_TO_ADDRESS_LINE_2	HR_LOCATIONS	ADDRESS_LINE_2	
SHIP_TO_ADDRESS_LINE_3	HR_LOCATIONS	ADDRESS_LINE_3	
SHIP_TO_TOWN_OR_CITY	HR_LOCATIONS	TOWN_OR_CITY	
SHIP_TO_COUNTRY	HR_LOCATIONS	COUNTRY	
SHIP_TO_POSTAL_CODE	HR_LOCATIONS	POSTAL_CODE	

Field Name	Oracle Applications Table/View Name	Column Name	Description
SHIP_TO_REGION_1	HR_LOCATIONS	REGION_1	
SHIP_TO_REGION_2	HR_LOCATIONS	REGION_2	
SHIP_TO_REGION_3	HR_LOCATIONS	REGION_3	
BILL_TO_ADDRESS_LINE_1	HR_LOCATIONS	ADDRESS_LINE_1	Address that will receive the bill for the PO items. Derives the BILL_TO_LOCATION ID
BILL_TO_ADDRESS_LINE_2	HR_LOCATIONS	ADDRESS_LINE_2	
BILL_TO_ADDRESS_LINE_3	HR_LOCATIONS	ADDRESS_LINE_3	
BILL_TO_TOWN_OR_CITY	HR_LOCATIONS	TOWN_OR_CITY	
BILL_TO_COUNTRY	HR_LOCATIONS	COUNTRY	
BILL_TO_POSTAL_CODE	HR_LOCATIONS	POSTAL_CODE	
BILL_TO_REGION_1	HR_LOCATIONS	REGION_1	
BILL_TO_REGION_2	HR_LOCATIONS	REGION_2	
BILL_TO_REGION_3	HR_LOCATIONS	REGION_3	
FOB_LOOKUP_CODE	PO_HEADERS_ALL	FOB_LOOKUP_CODE	Type of Free On Board terms for the shipment.
CURRENCY_CODE	PO_HEADERS_ALL	CURRENCY_CODE	
RATE_DATE	PO_HEADERS_ALL	RATE_DATE	
START_DATE	PO_HEADERS_ALL	START_DATE	Effective date of the price break.
BLANKET_TOTAL_AMOUNT	PO_HEADERS_ALL	BLANKET_TOTAL_AMOUNT	Amount agreed for the PO.

Field Name	Oracle Applications Table/View Name	Column Name	Description
APPROVED_DATE	PO_HEADERS_ALL	APPROVED_DATE	
NOTE_TO_RECEIVER	PO_HEADERS_ALL	NOTE_TO_RECEIVER	
PRINTED_DATE	PO_HEADERS_ALL	PRINTED_DATE	Date when PO is printed.
CONFIRMING_ORDER_FLAG	PO_HEADERS_ALL	CONFIRMING_ORDER_FLAG	Indicates whether the PO is a confirmed order.
REPLY_METHOD_LOOKUP_CODE	PO_HEADERS_ALL	REPLY_METHOD_LOOKUP_CODE	Method the supplier should reply to the PO.
TYPE_LOOKUP_CODE	PO_HEADERS_ALL	TYPE_LOOKUP_CODE	Required. Use STANDARD, BLANKET, or QUOTATION. Maps directly.
USSGL_TRANSACTION_CODE	PO_HEADERS_ALL	USSGL_TRANSACTION_CODE	USA standard general ledger transaction code.
CLOSED_DATE	PO_HEADERS_ALL	CLOSED_DATE	Date when the PO closes.
RFQ_CLOSE_DATE	PO_HEADERS_ALL	RFQ_CLOSE_DATE	Date when the Request For Quotation (RFQ) closes.
QUOTE_WARNING_DELAY	PO_HEADERS_ALL	QUOTE_WARNING_DELAY	Number of days before the quotation expires.
QUOTE_VENDOR_QUOTE_NUMBER	VENDOR_DOC_NUM	QUOTE_VENDOR_QUOTE_NUMBER	Supplier document number.
AGENT_NAME	PER_ALL_PEOPLE_F	NAME	Buyer name.
ADDRESS_LINE1	PO_VENDOR_SITES_ALL	ADDRESS_LINE1	Supplier's address.
ADDRESS_LINE2	PO_VENDOR_SITES_ALL	ADDRESS_LINE2	
ADDRESS_LINE3	PO_VENDOR_SITES_ALL	ADDRESS_LINE3	

Field Name	Oracle Applications Table/View Name	Column Name	Description
CITY	PO_VENDOR_SITES_ALL	CITY	
STATE	PO_VENDOR_SITES_ALL	STATE	
ZIP	PO_VENDOR_SITES_ALL	ZIP	
COUNTRY	PO_VENDOR_SITES_ALL	COUNTRY	
PHONE	PO_VENDOR_SITES_ALL	PHONE	
FAX	PO_VENDOR_SITES_ALL	FAX	
VENDOR_CONTACT	PO_VENDOR_CONTACTS	LAST_NAME, FIRST_NAME	Supplier contact name.
PAYMENT_TERMS	AP_TERMS	NAME	
RELEASE_NUM	PO_RELEASES_ALL	RELEASE_NUM	PO release number.
INTERFACE_SOURCE_CODE	PO_HEADERS_ALL	INTERFACE_SOURCE_CODE	Indicates the source where Oracle Applications gets the PO.
PO_HEADER_ID	PO_HEADERS_ALL	PO_HEADER_ID	Required for outbound transactions. Unique identifier for PO header.

1.1. PO_LINES

Field Name	Oracle Applications Table/View Name	Column Name	Description
LINE_NUM	PO_LINES_ALL	LINE_NUM	PO line number.
LINE_TYPE	PO_LINE_TYPES	LINE_TYPE	PO line type.
ITEM_REVISION	PO_LINES_ALL	ITEM_REVISION	PO item revision.
ITEM_DESCRIPTION	PO_LINES_ALL	ITEM_DESCRIPTION	
UNIT_MEAS_LOOKUP_CODE	PO_LINES_ALL	UNIT_MEAS_LOOKUP_CODE	Unit of measure for the items.

Field Name	Oracle Applications Table/View Name	Column Name	Description
COMMITTED_AMOUNT	PO_LINES_ALL	COMMITTED_AMOUNT	Supplier's agreed amount for the items.
ALLOW_PRICE_OVERRIDE_FLAG	PO_LINES_ALL	ALLOW_PRICE_OVERRIDE_FLAG	Indicates whether the price can be changed for a PO release.
NOT_TO_EXCEED_PRICE	PO_LINES_ALL	NOT_TO_EXCEED_PRICE	Maximum price limit for a PO release.
LIST_PRICE_PER_UNIT	PO_LINES_ALL	LIST_PRICE_PER_UNIT	
UNIT_PRICE	PO_LINES_ALL	UNIT_PRICE	
QUANTITY	PO_LINES_ALL	QUANTITY	Quantity of each item ordered.
VENDOR_PRODUCT_NUM	PO_LINES_ALL	VENDOR_PRODUCT_NUM	Supplier's item number.
UN_NUMBER	PO_UN_NUMBERS	UN_NUMBER	
HAZARD_CLASS	PO_HAZARD_CLASSES	HAZARD_CLASS	Class of the hazard item.
MIN_ORDER_QUANTITY	PO_LINES_ALL	MIN_ORDER_QUANTITY	Minimum order quantity for the PO.
MAX_ORDER_QUANTITY	PO_LINES_ALL	MAX_ORDER_QUANTITY	Maximum order quantity for the PO.
QTY_RCV_TOLERANCE	PO_LINES_ALL	QTY_RCV_TOLERANCE	Maximum over-receipt tolerance percentage for the quantity of the items received.
OVER_TOLERANCE_ERROR_FLAG	PO_LINES_ALL	OVER_TOLERANCE_ERROR_FLAG	A shipment is automatically closed for invoicing if this tolerance percentage is reached when billed.
MARKET_PRICE	PO_LINES_ALL	MARKET_PRICE	Market price for the item ordered in the PO.
FIRM_STATUS_LOOKUP_CODE	PO_LINES_ALL	FIRM_STATUS_LOOKUP_CODE	Indicates whether the PO should be prevented from being automatically rescheduled by the manufacturing application.

Field Name	Oracle Applications Table/View Name	Column Name	Description
NOTE_TO_VENDOR	PO_LINES_ALL	NOTE_TO_VENDOR	Note to the supplier.
TAXABLE_FLAG	PO_LINES_ALL	TAXABLE_FLAG	Indicates whether the item is taxable.
TAX_NAME	AP_TAX_CODES_ALL	NAME	Name of the tax applied to the item.
TYPE_1099	PO_LINES_ALL	TYPE_1099	The 1099 type for the purchase order item.
CAPITAL_EXPENSE_FLAG	PO_LINES_ALL	CAPITAL_EXPENSE_FLAG	Indicates whether the item cost is a capital expense.
NEGOTIATED_BY_PREPARER_FLAG	PO_LINES_ALL	NEGOTIATED_BY_PREPARER_FLAG	Indicates whether the buyer negotiated the price.
MIN_RELEASE_AMOUNT	PO_LINES_ALL	MIN_RELEASE_AMOUNT	
PRICE_TYPE_LOOKUP_CODE	PO_LINES_ALL	PRICE_TYPE_LOOKUP_CODE	Default price type for a PO.
PRICE_BREAK_LOOKUP_CODE	PO_LINES_ALL	PRICE_BREAK_LOOKUP_CODE	Default price break type for catalog.
TRANSACTION_REASON_CODE	PO_LINES_ALL	TRANSACTION_REASON_CODE	
USSGL_TRANSACTION_CODE	PO_LINES_ALL	USSGL_TRANSACTION_CODE	USA standard general ledger transaction code.
ITEM	MTL_SYSTEM_ITEMS	SEGMENT1	
UOM_CODE	MTL_UNITS_OF_MEASURE	UOM_CODE	Unit of measurement.
CATEGORY	MTL_CATEGORIES_KFV	CONCATENATE_D_SEGMENTS	Category of the item in the PO.
FROM_HEADER_NUM	PO_HEADERS_ALL	SEGMENT1	Unique header number to automatically create the PO.
UNIT_WEIGHT	MTL_SYSTEM_ITEMS	UNIT_WEIGHT	Units used for weight.

Field Name	Oracle Applications Table/View Name	Column Name	Description
WEIGHT_UOM_CODE	MTL_SYSTEM_ITEMS	WEIGHT_UOM_CODE	Weight unit of measurement.
VOLUME_UOM_CODE	MTL_SYSTEM_ITEMS	VOLUME_UOM_CODE	Volume unit of measurement.
UNIT_VOLUME	MTL_SYSTEM_ITEMS	UNIT_VOLUME	Units used for volume.
SOURCING_RULE_NAME	MRP_SOURCING_RULES	SOURCING_RULE_NAME	Name of the sourcing rule created by purchasing documents.
ACTION			Action to be completed on the purchase order line for inbound transactions. Use: NEW for a new PO, or ADDED to add a line to an existing PO.
PO_LINE_ID	PO_LINES_ALL	PO_LINE_ID	Required for outbound transactions. Unique identifier for PO lines.
PO_HEADER_ID	PO_HEADERS_ALL	PO_HEADER_ID	Required for outbound transactions. Unique identifier for PO headers.
CANCEL_FLAG	PO_LINES_ALL	CANCEL_FLAG	Indicates whether PO was cancelled.
CANCEL_REASON	PO_LINES_ALL	CANCEL_REASON	
CANCEL_DATE	PO_LINES_ALL	CANCEL_DATE	
CLOSED_FLAG	PO_LINES_ALL	CLOSED_FLAG	Indicates whether a PO is closed.
CLOSED_REASON	PO_LINES_ALL	CLOSED_REASON	
CLOSED_DATE	PO_LINES_ALL	CLOSED_DATE	

1.1.1. PO_LINE_LOCATIONS

Field Name	Oracle Applications Table/View Name	Column Name	Description
SHIPMENT_NUM	PO_LINE_LOCATIONS	SHIPMENT_NUM	PO shipment number.
SHIP_TO_ORGANIZATION_NAME	ORG_ORGANIZATION_DEFS	ORGANIZATION_NAME	Organization that will receive PO items.
SHIP_TO_LOCATION	HR_LOCATIONS	DESCRIPTION	Description of location to ship the PO items.
SHIP_TO_LOC_ADDRESS_LINE_1	HR_LOCATIONS	ADDRESS_LINE_1	Address of location to ship the PO items.
SHIP_TO_LOC_ADDRESS_LINE_2	HR_LOCATIONS	ADDRESS_LINE_2	
SHIP_TO_LOC_ADDRESS_LINE_3	HR_LOCATIONS	ADDRESS_LINE_3	
SHIP_TO_LOC_TOWN_OR_CITY	HR_LOCATIONS	TOWN_OR_CITY	
SHIP_TO_LOCATION_COUNTRY	HR_LOCATIONS	COUNTRY	
SHIP_TO_LOCATION_POSTAL_CODE	HR_LOCATIONS	POSTAL_CODE	
SHIP_TO_LOCATION_REGION_1	HR_LOCATIONS	REGION_1	
SHIP_TO_LOCATION_REGION_2	HR_LOCATIONS	REGION_2	
SHIP_TO_LOCATION_REGION_3	HR_LOCATIONS	REGION_3	
NEED_BY_DATE	PO_LINE_LOCATIONS	NEED_BY_DATE	Due date when the PO item needs to be shipped.
PROMISED_DATE	PO_LINE_LOCATIONS	PROMISED_DATE	Date promised by the supplier when the item would be delivered.
PRICE_DISCOUNT	PO_LINE_LOCATIONS	PRICE_DISCOUNT	Discount percentage for price break.
EFFECTIVE_START_DATE	PO_LINE_LOCATIONS	START_DATE	Effective date for price break.

Field Name	Oracle Applications Table/View Name	Column Name	Description
EXPIRATION_END_DATE	PO_LINE_LOCATIONS	END_DATE	Expiration date for price break.
LEAD_TIME	PO_LINE_LOCATIONS	LEAD_TIME	Lead time for price break.
LEAD_TIME_UNIT	PO_LINE_LOCATIONS	LEAD_TIME_UNIT	Unit of measurement for the lead time.
INSPECTION_REQUIRED_FLAG	PO_LINE_LOCATIONS	INSPECTION_REQUIRED_FLAG	Indicates whether the shipment should be inspected before the invoice should be paid.
RECEIPT_REQUIRED_FLAG	PO_LINE_LOCATIONS	RECEIPT_REQUIRED_FLAG	Indicates whether the shipment must be received before the invoice is paid.
RECEIVE_CLOSE_TOLERANCE	PO_LINE_LOCATIONS	RECEIVE_CLOSE_TOLERANCE	Percentage tolerance within which a shipment is automatically closed for receiving.
INVOICE_CLOSE_TOLERANCE	PO_LINE_LOCATIONS	INVOICE_CLOSE_TOLERANCE	Percentage tolerance within which a shipment is automatically closed for invoicing before billing.
QTY_RCV_EXCEPTION_CODE	PO_LINE_LOCATIONS	QTY_RCV_EXCEPTION_CODE	Describes the action taken when the quantity-received tolerance is exceeded.
DAYS_EARLY_RECEIPT_ALLOWED	PO_LINE_LOCATIONS	DAYS_EARLY_RECEIPT_ALLOWED	Maximum acceptable number of days that items can be received early.
DAYS_LATE_RECEIPT_ALLOWED	PO_LINE_LOCATIONS	DAYS_LATE_RECEIPT_ALLOWED	Minimum acceptable number of days that items can be received late.
ENFORCE_SHIP_TO_LOCATION_CODE	PO_LINE_LOCATIONS	ENFORCE_SHIP_TO_LOCATION_CODE	Describes the action taken when the receiving location differs from the Ship To location.
RECEIPT_DAYS_EXCEPTION_CODE	PO_LINE_LOCATIONS	RECEIPT_DAYS_EXCEPTION_CODE	Describes the action taken when items are received earlier or later than the allowed number of days.

Field Name	Oracle Applications Table/View Name	Column Name	Description
ACCRUE_ON_RECEIPT_FLAG	PO_LINE_LOCATIONS	ACCRUE_ON_RECEIPT_FLAG	Indicates whether items are accrued upon receipt.
ALLOW_SUBSTITUTE_RECEIPTS_FLAG	PO_LINE_LOCATIONS	ALLOW_SUBSTITUTE_RECEIPTS_FLAG	Indicates whether substitute items can be received in place of ordered items.
LINE_LOCATION_ID	PO_LINE_LOCATIONS	LINE_LOCATION_ID	Required for outbound transactions. Unique identifier for PO Line Locations.
PO_LINE_ID	PO_LINES_ALL	PO_LINE_ID	Required for outbound transactions. Unique identifier for PO lines.
CANCEL_FLAG	PO_LINES_ALL	CANCEL_FLAG	Indicates whether the PO was cancelled.
CANCEL_REASON	PO_LINES_ALL	CANCEL_REASON	
CANCEL_DATE	PO_LINES_ALL	CANCEL_DATE	
CLOSED_FLAG	PO_LINES_ALL	CLOSED_FLAG	Indicates whether the PO was closed.
CLOSED_REASON	PO_LINES_ALL	CLOSED_REASON	
CLOSED_DATE	PO_LINES_ALL	CLOSED_DATE	

1.1.1.1. PO_DISTRIBUTIONS

Field Name	Oracle Applications Table/View Name	Column Name	Description
RATE_DATE	PO_DISTRIBUTIONS_ALL	RATE_DATE	Currency conversion date.
AMOUNT_BILLED	PO_DISTRIBUTIONS_ALL	AMOUNT_BILLED	Amount invoiced against distribution.
ENCUMBERED_AMOUNT	PO_DISTRIBUTIONS_ALL	ENCUMBERED_AMOUNT	Distribution encumbered amount.

Field Name	Oracle Applications Table/View Name	Column Name	Description
GL_ENCUMBERED_PERIOD_NAME	PO_DISTRIBUTIONS_ALL	GL_ENCUMBERED_PERIOD_NAME	Period in which the distribution was encumbered.
WIP_RESOURCE_SEQ_NUM	PO_DISTRIBUTIONS_ALL	WIP_RESOURCE_SEQ_NUM	Work In Process resource sequence number.
DESTINATION_CONTEXT	PO_DISTRIBUTIONS_ALL	DESTINATION_CONTEXT	Destination details description.
USSGL_TRANSACTION_CODE	PO_DISTRIBUTIONS_ALL	USSGL_TRANSACTION_CODE	USA standard general ledger transaction code.
EXPENDITURE_TYPE	PO_DISTRIBUTIONS_ALL	EXPENDITURE_TYPE	Project accounting expenditure type.
PROJECT_ACCOUNTING_CONTEXT	PO_DISTRIBUTIONS_ALL	PROJECT_ACCOUNTING_CONTEXT	Project accounting context.
GL_CLOSED_DATE	PO_DISTRIBUTIONS_ALL	GL_CLOSED_DATE	Date when the distribution is closed.
ACCRUE_ON_RECEIPT_FLAG	PO_DISTRIBUTIONS_ALL	ACCRUE_ON_RECEIPT_FLAG	Indicates whether the items are accrued on receipt.
EXPENDITURE_ITEM_DATE	PO_DISTRIBUTIONS_ALL	EXPENDITURE_ITEM_DATE	Project accounting expenditure item date.
QUANTITY_ORDERED	PO_DISTRIBUTIONS_ALL	QUANTITY_ORDERED	Quantity ordered on the distribution.
QUANTITY_DELIVERED	PO_DISTRIBUTIONS_ALL	QUANTITY_DELIVERED	Quantity delivered against the distribution.
QUANTITY_CANCELLED	PO_DISTRIBUTIONS_ALL	QUANTITY_CANCELLED	Quantity cancelled for the distribution.

Field Name	Oracle Applications Table/View Name	Column Name	Description
QUANTITY_BILLED	PO_DISTRIBUTIONS_ALL	QUANTITY_BILLED	Quantity invoiced against the distribution.
REQ_HEADER_REFERENCE_NUM	PO_DISTRIBUTIONS_ALL	REQ_HEADER_REFERENCE_NUM	Requisition number of the requisition line placed on the distribution.
REQ_LINE_REFERENCE_NUM	PO_DISTRIBUTIONS_ALL	REQ_LINE_REFERENCE_NUM	Paper requisition line number.
RATE	PO_DISTRIBUTIONS_ALL	RATE	Currency conversion rate.
ACCRUED_FLAG	PO_DISTRIBUTIONS_ALL	ACCRUED_FLAG	Indicates whether distribution was accrued.
ENCUMBERED_FLAG	PO_DISTRIBUTIONS_ALL	ENCUMBERED_FLAG	Distribution encumbered amount.
UNENCUMBERED_QUANTITY	PO_DISTRIBUTIONS_ALL	UNENCUMBERED_QUANTITY	Quantity unencumbered on the distribution.
UNENCUMBERED_AMOUNT	PO_DISTRIBUTIONS_ALL	UNENCUMBERED_AMOUNT	Amount unencumbered on the distribution.
GL_ENCUMBERED_DATE	PO_DISTRIBUTIONS_ALL	GL_ENCUMBERED_DATE	Date the distribution was encumbered.
GL_CANCELLED_DATE	PO_DISTRIBUTIONS_ALL	GL_CANCELLED_DATE	Date the distribution was cancelled.
DESTINATION_TYPE_CODE	PO_DISTRIBUTIONS_ALL	DESTINATION_TYPE_CODE	Final destination name of the purchase item.
DESTINATION_SUBINVENTORY	PO_DISTRIBUTIONS_ALL	DESTINATION_SUBINVENTORY	Final destination subinventory for inventory purchases.

Field Name	Oracle Applications Table/View Name	Column Name	Description
WIP_OPERATION_SEQ_NUM	PO_DISTRIBUTIONS_ALL	WIP_OPERATION_SEQ_NUM	Work In Process operations sequence number with routing.
DISTRIBUTION_NUM	PO_DISTRIBUTIONS_ALL	DISTRIBUTION_NUM	Document distribution number.
PREVENT_ENCUMBRANCE_FLAG	PO_DISTRIBUTIONS_ALL	PREVENT_ENCUMBRANCE_FLAG	Indicates whether distribution should be encumbered.
DELIVERY_TO_LOCATION	HR_LOCATIONS	LOCATION_CODE	Location for the delivery.
DELIVERY_TO_PERSON_FULL_NAME	PER_PEOPLE_F	FULL_NAME	Person who will receive delivery.
DESTINATION_ORGANIZATION	HR_ALL_ORGANIZATION_UNITS	NAME	Organization that will receive the distribution.
SET_OF_BOOKS	GL_SET_OF_BOOKS	NAME	Set of general ledger books.
CHARGE_ACCOUNT	GL_CODE_COMBINATIONS_KFV	CONCATENATE_D_SEGMENTS	Charge account number.
BUDGET_ACCOUNT	GL_CODE_COMBINATIONS_KFV	CONCATENATE_D_SEGMENTS	Budget account number.
ACCURAL_ACCOUNT	GL_CODE_COMBINATIONS_KFV	CONCATENATE_D_SEGMENTS	Accrual account number.
VARIANCE_ACCOUNT	GL_CODE_COMBINATIONS_KFV	CONCATENATE_D_SEGMENTS	Variance account number.
WIP_ENTITY	WIP_ENTITIES	WIP_ENTITY_NAME	Work In Process job.
WIP_REPETITIVE_SCHEDULE	WIP_REPETITIVE_SCHEDULES	DESCRIPTION	Work In Process repetitive schedule.
WIP_LINE_CODE	WIP_LINES	LINE_CODE	Work In Process line.

Field Name	Oracle Applications Table/View Name	Column Name	Description
BOM_RESOURCE	BOM_RESOURCES	RESOURCE_CODE	Bill of Material resource.
PROJECT_NAME	PA_PROJECTS_ALL	NAME	
TASK_NAME	PA_TASKS	TASK_NAME	
EXPENDITURE	PA_EXPENDITURE_TYPES	DESCRIPTION	Project accounting expenditure.
EXPENDITURE_ORG_NAME	HR_ALL_ORGANIZATION_UNITS	NAME	Project accounting expenditure organization name.
PROJECT_RELATED_FLAG	PO_DISTRIBUTIONS_ALL	EXPENDITURE_TYPE	Project accounting's project related flag that uses N when NULL.
LINE_LOCATION_ID	PO_LINE_LOCATIONS_ALL	LINE_LOCATION_ID	Required for outbound transactions. Unique identifier for PO line locations.

Send Supplier Service

The name of this service is:

WmOAPRC107SC.purchasing107SC.fromOA.supplier:sendSupplier

This service notifies and delivers supplier changes. It extracts information for a supplier only, and not for individual employees who are also set up as suppliers. Because suppliers cannot be deleted, the document status of DELETE does not apply in this case.

Database Scripts

This service uses the following database scripts:

Script	Description
wm_install_from_supplier.sql	Runs the scripts listed below, except the uninstall script.
wm_from_supplier_vw.sql	Creates the following required view components for PO outbound transactions: <ul style="list-style-type: none"> ■ WM_PO_SUPPLIER_VW ■ WM_PO_SUPPLIER_SITES_ALL_VW ■ WM_PO_SUPPLIER_CONTACTS_VW ■ WM_BANK_ACC_USE_SUPP_SITES_VW ■ WM_BANK_ACCOUNT_USES_SUPP_VW ■ WM_PO_SUPPLIER_QRY_VW
wm_from_supplier_trg.sql	Creates the following triggers to be used for writing to the WM_TRACKCHANGES table to indicate a new or updated document: <ul style="list-style-type: none"> ■ WM_PO_SUPPLIER_IU_TRG ■ WM_PO_SUPP_SITES_ALL_IU_TRG ■ WM_PO_SUPP_CONTACTS_IUD_TRG ■ WM_AP_BK_SUP_AC_USE_ALL_IU_TRG
wm_disable_from_supplier.sql	Disables the triggers installed by wm_from_supplier_trg.sql.
wm_enable_from_supplier.sql	Re-enables the triggers installed by wm_from_supplier_trg.sql.
wm_drop_from_supplier.sql	Uninstalls all components created by wm_install_from_supplier.sql.

For more information about using database scripts, see [“Database Scripts” on page 26](#).

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definition:

- `getSupplierTxn107SC.txp`

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow `sendSupplier` executes as follows:

- `specifyDefaultSettings` specifies the default parameter settings required for service execution. You should change these settings accordingly.
- `lockTxnCtrl` checks whether another instance of this service is already in process. If no instance is already in process, the service takes control of processing in a single atomic action.
 - If `lockTxnCtrl` returns `FALSE`, it indicates that another instance of this service is already in process the service exits and waits for next scheduled execution.
 - If `lockTxnCtrl` returns `TRUE`, it indicates that the service is ready to execute, the Supplier row in the control table is locked and updated to `INPROCESS`. This prevents any other Supplier service from executing.
- `getSupplierTxn` queries the Oracle Applications database for any supplier transaction to be processed. The number of records returned depends on the parameter value specified in the above map.
- `processBizDoc` is the customizable step that sends the business document to the recipient (such as a trading partner) by looping against each document. This step needs to be customized to receive a `SUCCESS` or an `ERROR` status of the document transfer along with the Error information. The transfer status and any error information are logged against each document.
- Loops against each document. Based on the Debug Mode specified during execution, it is determined to either purge or update the records in the `WM_TRACKCHANGES` table.
 - If Debug Mode is `TRUE`, the records in the `WM_TRACKCHANGES` table are updated and the `PROCESSED_FLAG` is set to `Y`. This ensures that the same sets of records are not picked up during next polling interval. The `updateTrackChanges` service updates the `PROCESSED_FLAG` in `WM_TRACKCHANGES` table to `Y`, and updates `Processed_Date` to `sysdate` so that same information is not picked up again during next polling instance.

- If Debug Mode is FALSE, the records in the WM_TRACKCHANGES table are deleted. The `purgeTrackChanges` service purges the records from the WM_TRACKCHANGES table.
- Loops against each document. Based on the Transfer Status, the `insertTransferERRInfo` service inserts a new record in the WM_TRACKCHANGES table so that same document can be picked up during the next polling interval.
- `unlockTxnCtrl` releases the lock on the Custom Control table so that next polling instance of `sendSupplier` service can begin.
- `getLastError` logs any errors that occur in the above steps.
- `unlockTxnCtrl` releases the lock on the Custom Control table.

For more details on send transactions, see [“Overview of Send Service Transaction Processing” on page 38](#).

Business Document Structure

This service uses the following business document structure:

- 1.0. SUPPLIER
 - 1.1. BANK_ACCOUNTS
 - 1.2. SUPPLIER_SITES
 - 1.2.1. SITE_BANK_ACCOUNTS
 - 1.2.2. SUPPLIER_CONTACTS

1.0. SUPPLIER

Field Name	Oracle Applications Table/View Name	Column Name	Description
WEB_TRANSACTION_ID			Populated from a sequence and used internally in the IS flow. Will contain NULL value for queried supplier data.
DOCUMENT_TYPE			Use SUPPLIER.
DOCUMENT_STATUS			Use INSERT, UPDATE, or QUERY.
VENDOR_ID	PO_VENDORS	VENDOR_ID	Unique supplier identifier in Oracle Applications.
VENDOR_NUMBER	PO_VENDORS	SEGMENT1	Supplier number.

Field Name	Oracle Applications Table/View Name	Column Name	Description
VENDOR_NAME	PO_VENDORS	VENDOR_NAME	Name of the supplier.
ALTERNATE_VENDOR_NAME	PO_VENDORS	VENDOR_NAME_ALT	Alternate name for the supplier.
VENDOR_TYPE	PO_VENDORS	VENDOR_TYPE_LOOKUP_CODE	
CUSTOMER_NUMBER	PO_VENDORS	CUSTOMER_NUM	Customer number as set up on supplier's system.
ONE_TIME_FLAG	PO_VENDORS	ONE_TIME_FLAG	Indicates whether supplier is one time.
PARENT_VENDOR_NUMBER	PO_VENDORS	SEGMENT1	Parent supplier number.
PARENT_VENDOR_NAME	PO_VENDORS	VENDOR_NAME	Parent supplier name.
MINIMUM_ORDER_AMOUNT	PO_VENDORS	MIN_ORDER_AMOUNT	Minimum order amount to order goods from the supplier.
BILL_LOC_ADDRESS_LINE1	HR_LOCATIONS	ADDRESS_LINE_1	REGION1 is the County, REGION2 is the State Code, BILL_LOC is the Billing Location, and SHIP_LOC is the Shipping Location.
BILL_LOC_ADDRESS_LINE2	HR_LOCATIONS	ADDRESS_LINE_2	
BILL_LOC_ADDRESS_LINE3	HR_LOCATIONS	ADDRESS_LINE_3	
BILL_LOC_TOWN_OR_CITY	HR_LOCATIONS	TOWN_OR_CITY	
BILL_LOC_REGION2	HR_LOCATIONS	REGION_2	
BILL_LOC_POSTAL_CODE	HR_LOCATIONS	POSTAL_CODE	
BILL_LOC_COUNTRY	HR_LOCATIONS	COUNTRY	
BILL_LOC_REGION1	HR_LOCATIONS	REGION_1	

Field Name	Oracle Applications Table/View Name	Column Name	Description
SHIP_LOC_ADDRESS_LINE1	HR_LOCATIONS	ADDRESS_LINE_1	
SHIP_LOC_ADDRESS_LINE2	HR_LOCATIONS	ADDRESS_LINE_2	
SHIP_LOC_ADDRESS_LINE3	HR_LOCATIONS	ADDRESS_LINE_3	
SHIP_LOC_TOWN_OR_CITY	HR_LOCATIONS	TOWN_OR_CITY	
SHIP_LOC_REGION2	HR_LOCATIONS	REGION_2	
SHIP_LOC_POSTAL_CODE	HR_LOCATIONS	POSTAL_CODE	
SHIP_LOC_COUNTRY	HR_LOCATIONS	COUNTRY	
SHIP_LOC_REGION1	HR_LOCATIONS	REGION_1	
SHIP_VIA	PO_VENDORS	SHIP_VIA_LOOKUP_CODE	Shipping code.
FREIGHT_TERMS	PO_VENDORS	FREIGHT_TERMS_LOOKUP_CODE	Freight terms code.
FOB	PO_VENDORS	FOB_LOOKUP_CODE	Free On Board code.
TERMS	AP_TERMS_TL	NAME	Term name.
SET_OF_BOOKS_NAME	GL_SETS_OF_BOOKS	NAME	Set of books name as in Oracle Applications.
SET_OF_BOOKS_SHORT_NAME	GL_SETS_OF_BOOKS	SHORT_NAME	Set of books short name.
CREDIT_STATUS	PO_VENDORS	CREDIT_STATUS_LOOKUP_CODE	Credit status of the supplier.
CREDIT_LIMIT	PO_VENDORS	CREDIT_LIMIT	
ALWAYS_DISCOUNT_FLAG	PO_VENDORS	ALWAYS_TAKE_DISC_FLAG	Discount flag.

Field Name	Oracle Applications Table/View Name	Column Name	Description
PAY_DATE_BASIS	PO_VENDORS	PAY_DATE_BASIS_LOOKUP_CODE	Type of payment date basis.
PAY_GROUP	PO_VENDORS	PAY_GROUP_LOOKUP_CODE	
PAYMENT_PRIORITY	PO_VENDORS	PAYMENT_PRIORITY	
INVOICE_CURRENCY_CODE	PO_VENDORS	INVOICE_CURRENCY_CODE	
PAYMENT_CURRENCY_CODE	PO_VENDORS	PAYMENT_CURRENCY_CODE	
INVOICE_AMOUNT_LIMIT	PO_VENDORS	INVOICE_AMOUNT_LIMIT	
EXCHANGE_DATE_CODE	PO_VENDORS	EXCHANGE_DATE_LOOKUP_CODE	
HOLD_ALL_PAYMENTS	PO_VENDORS	HOLD_ALL_PAYMENTS_FLAG	
HOLD_FUTURE_PAYMENTS	PO_VENDORS	HOLD_FUTURE_PAYMENTS_FLAG	
HOLD_REASON	PO_VENDORS	HOLD_REASON	
DISTRIBUTION_SET_NAME	AP_DISTRIBUTION_SETS	DISTRIBUTION_SET_NAME	
SUPPLIER LIABILITY_ACCOUNT	GL_CODE_COMBINATION_S_KFV	CONCATENATE_D_SEGMENTS	Concatenated general ledger code combination.
NUMBER_1099	PO_VENDORS	NUM_1099	Taxpayer ID.
TYPE_1099	PO_VENDORS	TYPE_1099	Type of 1099.
WITHHOLDING_STATUS	PO_VENDORS	WITHHOLDING_STATUS_LOOKUP_CODE	

Field Name	Oracle Applications Table/View Name	Column Name	Description
WITHHOLDING_START_DATE	PO_VENDORS	WITHHOLDING_START_DATE	
IRS_ORGANIZATION_TYPE	PO_VENDORS	ORGANIZATION_TYPE_LOOKUP_CODE	
VAT_CODE	PO_VENDORS	VAT_CODE	
START_DATE_ACTIVE	PO_VENDORS	START_DATE_ACTIVE	Active starting date.
END_DATE_ACTIVE	PO_VENDORS	END_DATE_ACTIVE	Active ending date.
MINORITY_GROUP	PO_VENDORS	MINORITY_GROUP_LOOKUP_CODE	
PAYMENT_METHOD	PO_VENDORS	PAYMENT_METHOD_LOOKUP_CODE	
BANK_ACCOUNT_NAME	PO_VENDORS	BANK_ACCOUNT_NAME	Supplier's bank account name.
BANK_ACCOUNT_NUMBER	PO_VENDORS	BANK_ACCOUNT_NUM	Supplier's bank account number.
BANK_NUMBER	PO_VENDORS	BANK_NUM	Supplier's bank number.
BANK_ACCOUNT_TYPE	PO_VENDORS	BANK_ACCOUNT_TYPE	Type of bank account.
WOMEN_OWNED	PO_VENDORS	WOMEN_OWNED_FLAG	Indicates whether the supplier is a woman-owned business.
SMALL_BUSINESS	PO_VENDORS	SMALL_BUSINESS_FLAG	Indicates whether the supplier is a small business.
STANDARD_INDUSTRY_CLASS	PO_VENDORS	STANDARD_INDUSTRY_CLASS	Standard industry classification number.
HOLD_FLAG	PO_VENDORS	HOLD_FLAG	

Field Name	Oracle Applications Table/View Name	Column Name	Description
PURCHASING_HOLD_REASON	PO_VENDORS	PURCHASING_HOLD_REASON	Reason for placing the supplier on purchasing hold.
HOLD_BY_EMPLOYEE_NUMBER	PER_ALL_PEOPLE_F	EMPLOYEE_NUMBER	
PER_HOLD_BY_FULL_NAME	PER_ALL_PEOPLE_F	FULL_NAME	
HOLD_DATE	PO_VENDORS	HOLD_DATE	Date the supplier was placed on purchasing hold.
TERMS_DATE_BASIS	PO_VENDORS	TERMS_DATE_BASIS	Type of invoice payment schedule basis.
PRICE_TOLERANCE	PO_VENDORS	PRICE_TOLERANCE	
INSPECTION_REQUIRED	PO_VENDORS	INSPECTION_REQUIRED_FLAG	Indicates whether inspection is required.
RECEIPT_REQUIRED_FLAG	PO_VENDORS	RECEIPT_REQUIRED_FLAG	Indicates whether shipment must be received before the invoice is paid.
QUANTITY_RECEIVED_TOLERANCE	PO_VENDORS	QTY_RCV_TOLERANCE	Quantity received tolerance percentage.
DAYS_EARLY_RECEIPT_ALLOWED	PO_VENDORS	DAYS_EARLY_RECEIPT_ALLOWED	Maximum acceptable number of days items can be received early.
DAYS_LATE_RECEIPT_ALLOWED	PO_VENDORS	DAYS_LATE_RECEIPT_ALLOWED	Maximum acceptable number of days items can be received late.
ALLOW_SUBSTITUTE_RECEIPTS	PO_VENDORS	ALLOW_SUBSTITUTE_RECEIPTS_FLAG	Indicates whether substitute items can be received in place of the ordered items.
ALLOW_UNORDERED_RECEIPTS	PO_VENDORS	ALLOW_UNORDERED_RECEIPTS_FLAG	Indicates whether unordered receipts are allowed.

Field Name	Oracle Applications Table/View Name	Column Name	Description
HOLD_UNMATCHED_INVOICES	PO_VENDORS	HOLD_UNMATCHED_INVOICES_FLAG	Indicates whether unmatched invoices should be put on hold.
EXCLUSIVE_PAYMENT	PO_VENDORS	EXCLUSIVE_PAYMENT_FLAG	
TAX_VERIFICATION_DATE	PO_VENDORS	TAX_VERIFICATION_DATE	
STATE_REPORTABLE	PO_VENDORS	STATE_REPORTABLE_FLAG	
FEDERAL_REPORTABLE	PO_VENDORS	FEDERAL_REPORTABLE_FLAG	
OFFSET_VAT	PO_VENDORS	OFFSET_VAT_CODE	Offset Vat Code.
VAT_REGISTRATION_NUMBER	PO_VENDORS	VAT_REGISTRATION_NUM	Tax registration number.
AUTO_CALCULATE_INTEREST	PO_VENDORS	AUTO_CALCULATE_INTEREST_FLAG	Indicates whether interest is to be automatically calculated.
VALIDATION_NUMBER	PO_VENDORS	VALIDATION_NUMBER	
EXCLUDE_FREIGHT_FROM_DISCOUNT	PO_VENDORS	EXCLUDE_FREIGHT_FROM_DISCOUNT	Exclude supplier freight from discount amount.
TAX_REPORTING_NAME	PO_VENDORS	TAX_REPORTING_NAME	Tax reporting method name.
CHECK_DIGITS	PO_VENDORS	CHECK_DIGITS	Check number used by payables.
BANK_NUM	PO_VENDORS	BANK_NUM	Bank number for Accounts Payable department.

Field Name	Oracle Applications Table/View Name	Column Name	Description
AUTO_TAX_CALCULATION	PO_VENDORS	AUTO_TAX_CALC_FLAG	
AUTO_TAX_CALCULATION_OVERRIDE	PO_VENDORS	AUTO_TAX_CALC_OVERRIDE	Allow calculation level override.
AMOUNT_INCLUDES_TAX	PO_VENDORS	AMOUNT_INCLUDES_TAX_FLAG	Distribution amounts include tax.
BANK_CHARGE_BEARER	PO_VENDORS	BANK_CHARGE_BEARER	Indicates whether the supplier bears bank charges.
BANK_BRANCH_TYPE	PO_VENDORS	BANK_BRANCH_TYPE	Identifies the list that contains the bank routing number. Use ABA, CHIPS, SWIFT, or OTHER.

1.1. BANK_ACCOUNTS

Field Name	Oracle Applications Table/View Name	Column Name	Description
ROW_ID		ROW_ID	Unique row identifier.
VENDOR_ID	AP_BANK_ACCOUNTS_ALL	VENDOR_ID	Unique vendor identifier in Oracle Applications.
VENDOR_SITE_ID	AP_BANK_ACCOUNT_USES_ALL	VENDOR_SITE_ID	Unique vendor site identifier.
BANK_ACCOUNT_NAME	AP_BANK_ACCOUNTS_ALL	BANK_ACCOUNT_NAME	
BANK_ACCOUNT_NUM	AP_BANK_ACCOUNTS_ALL	BANK_ACCOUNT_NUM	
CURRENCY_CODE	AP_BANK_ACCOUNTS_ALL	CURRENCY_CODE	
BANK_NAME	AP_BANK_BRANCHES	BANK_NAME	

Field Name	Oracle Applications Table/View Name	Column Name	Description
BANK_NUMBER	AP_BANK_BRANCHES	BANK_NUMBER	
BANK_BRANCH_NAME	AP_BANK_BRANCHES	BANK_BRANCH_NAME	
BANK_NUM	AP_BANK_BRANCHES	BANK_NUM	
END_DATE	AP_BANK_ACCOUNTS_USES_ALL	END_DATE	
START_DATE	AP_BANK_ACCOUNTS_USES_ALL	START_DATE	
PRIMARY	AP_BANK_ACCOUNT_USES_ALL	PRIMARY_FLAG	Indicates primary bank account.
ORG_ID	AP_BANK_ACCOUNT_USES_ALL	ORG_ID	
ORGANIZATION_CODE	ORG_ORGANIZATION_DEFINITIONS	ORGANIZATION_CODE	
ORGANIZATION_NAME	ORG_ORGANIZATION_DEFINITIONS	ORGANIZATION_NAME	

1.2. SUPPLIER_SITES

Field Name	Oracle Applications Table/View Name	Column Name	Description
VENDOR_ID	PO_VENDOR_SITES_ALL	VENDOR_ID	Unique vendor identifier in Oracle Applications.
VENDOR_SITE_ID	PO_VENDOR_SITES_ALL	VENDOR_SITE_ID	Unique supplier site identifier.
VENDOR_SITE_CODE	PO_VENDOR_SITES_ALL	VENDOR_SITE_CODE	Supplier site name.

Field Name	Oracle Applications Table/View Name	Column Name	Description
PURCHASING_SITE	PO_VENDOR_SITES_ALL	PURCHASING_SITE_FLAG	Indicates whether you can purchase from this site.
RFQ_ONLY_SITE	PO_VENDOR_SITES_ALL	RFQ_ONLY_SITE_FLAG	Indicates whether you can only send a Request For Quotation (RFQ) to this site.
PAY_SITE	PO_VENDOR_SITES_ALL	PAY_SITE_FLAG	Indicates whether you can send payments to this site.
ATTENTION_AR	PO_VENDOR_SITES_ALL	ATTENTION_AR_FLAG	Indicates whether the payments should be sent to the Account Receivables department.
ADDRESS_LINE1	PO_VENDOR_SITES_ALL	ADDRESS_LINE1	Supplier site address line 1.
ADDRESS_LINE2	PO_VENDOR_SITES_ALL	ADDRESS_LINE2	Supplier site address line 2.
ADDRESS_LINE3	PO_VENDOR_SITES_ALL	ADDRESS_LINE3	Supplier site address line 3.
CITY	PO_VENDOR_SITES_ALL	CITY	
STATE	PO_VENDOR_SITES_ALL	STATE	
ZIP	PO_VENDOR_SITES_ALL	ZIP	
PROVINCE	PO_VENDOR_SITES_ALL	PROVINCE	
COUNTRY	PO_VENDOR_SITES_ALL	COUNTRY	
AREA_CODE	PO_VENDOR_SITES_ALL	AREA_CODE	
PHONE	PO_VENDOR_SITES_ALL	PHONE	
CUSTOMER_NUMBER	PO_VENDOR_SITES_ALL	CUSTOMER_NUMBER	Customer number with the supplier.

Field Name	Oracle Applications Table/View Name	Column Name	Description
BILL_TO_LOCATION_CODE	HR_LOCATIONS	LOCATION_CODE	REGION1 is the County, REGION2 is the State Code, BILL_LOC is the Billing Location, and SHIP_LOC is the Shipping Location.
BILL_LOC_ADDRESS_LINE1	HR_LOCATIONS	ADDRESS_LINE_1	
BILL_LOC_ADDRESS_LINE2	HR_LOCATIONS	ADDRESS_LINE_2	
BILL_LOC_ADDRESS_LINE3	HR_LOCATIONS	ADDRESS_LINE_3	
BILL_LOC_TOWN_OR_CITY	HR_LOCATIONS	TOWN_OR_CITY	
BILL_LOC_REGION2	HR_LOCATIONS	REGION_2	
BILL_LOC_POSTAL_CODE	HR_LOCATIONS	POSTAL_CODE	
BILL_LOC_COUNTRY	HR_LOCATIONS	COUNTRY	
BILL_LOC_REGION1	HR_LOCATIONS	REGION_1	
SHIP_TO_LOCATION_CODE	HR_LOCATIONS	LOCATION_CODE	
SHIP_LOC_ADDRESS_LINE1	HR_LOCATIONS	ADDRESS_LINE_1	
SHIP_LOC_ADDRESS_LINE2	HR_LOCATIONS	ADDRESS_LINE_2	
SHIP_LOC_ADDRESS_LINE3	HR_LOCATIONS	ADDRESS_LINE_3	
SHIP_LOC_TOWN_OR_CITY	HR_LOCATIONS	TOWN_OR_CITY	
SHIP_LOC_REGION2	HR_LOCATIONS	REGION_2	
SHIP_LOC_POSTAL_CODE	HR_LOCATIONS	POSTAL_CODE	

Field Name	Oracle Applications Table/View Name	Column Name	Description
SHIP_LOC_COUNTRY	HR_LOCATIONS	COUNTRY	
SHIP_LOC_REGION1	HR_LOCATIONS	REGION_1	
SHIP_VIA	PO_VENDOR_SITES_ALL	SHIP_VIA_LOOKUP_CODE	Shipping code.
FREIGHT_TERMS	PO_VENDOR_SITES_ALL	FREIGHT_TERMS_LOOKUP_CODE	Freight terms code.
FOB	PO_VENDOR_SITES_ALL	FOB_LOOKUP_CODE	Default Free On Board type.
INACTIVE_DATE	PO_VENDOR_SITES_ALL	INACTIVE_DATE	Inactive date for record.
FAX	PO_VENDOR_SITES_ALL	FAX	
FAX_AREA_CODE	PO_VENDOR_SITES_ALL	FAX_AREA_CODE	
TELEX	PO_VENDOR_SITES_ALL	TELEX	
PAYMENT_METHOD	PO_VENDOR_SITES_ALL	PAYMENT_METHOD_LOOKUP_CODE	
BANK_ACCOUNT_NAME	PO_VENDOR_SITES_ALL	BANK_ACCOUNT_NAME	Supplier's bank account name.
BANK_ACCOUNT_NUMBER	PO_VENDOR_SITES_ALL	BANK_ACCOUNT_NUMBER	Supplier's bank account number.
BANK_NUM	PO_VENDOR_SITES_ALL	BANK_NUM	Bank branch number.
BANK_ACCOUNT_TYPE	PO_VENDOR_SITES_ALL	BANK_ACCOUNT_TYPE	
TERMS_DATE_BASIS	PO_VENDOR_SITES_ALL	TERMS_DATE_BASIS	Type of payment date basis.

Field Name	Oracle Applications Table/View Name	Column Name	Description
VAT_CODE	PO_VENDOR_SITES_ALL	VAT_CODE	
DISTRIBUTION_SET_NAME	AP_DISTRIBUTION_SETS	DISTRIBUTION_SET_NAME	
SUPPLIER_LIABILITY_ACCOUNT	GL_CODE_COMBINATION_S_KFV	CONCATENATE_D_SEGMENTS	Concatenated General Ledger code combination.
PREPAY_ACCOUNT	GL_CODE_COMBINATION_S_KFV	CONCATENATE_D_SEGMENTS	Concatenated General Ledger code combination.
PAY_GROUP	PO_VENDOR_SITES_ALL	PAY_GROUP_LOOKUP_CODE	
PAYMENT_PRIORITY	PO_VENDOR_SITES_ALL	PAYMENT_PRIORITY	
TERMS	AP_TERMS_TL	NAME	
INVOICE_AMOUNT_LIMIT	PO_VENDOR_SITES_ALL	INVOICE_AMOUNT_LIMIT	
PAY_DATE_BASIS	PO_VENDOR_SITES_ALL	PAY_DATE_BASIS_LOOKUP_CODE	Type of payment date basis.
ALWAYS_DISCOUNT_FLAG	PO_VENDOR_SITES_ALL	ALWAYS_TAKE_DISC_FLAG	Indicates whether discount is applicable for the site.
INVOICE_CURRENCY_CODE	PO_VENDOR_SITES_ALL	INVOICE_CURRENCY_CODE	
PAYMENT_CURRENCY_CODE	PO_VENDOR_SITES_ALL	PAYMENT_CURRENCY_CODE	
HOLD_ALL_PAYMENTS	PO_VENDOR_SITES_ALL	HOLD_ALL_PAYMENTS_FLAG	
HOLD_FUTURE_PAYMENTS	PO_VENDOR_SITES_ALL	HOLD_FUTURE_PAYMENTS_FLAG	

Field Name	Oracle Applications Table/View Name	Column Name	Description
HOLD_REASON	PO_VENDOR_SITES_ALL	HOLD_REASON	
HOLD_UNMATCHED_INVOICES	PO_VENDOR_SITES_ALL	HOLD_UNMATCHED_INVOICES_FLAG	Indicates whether unmatched invoices should be put on hold.
EXCLUSIVE_PAYMENT	PO_VENDOR_SITES_ALL	EXCLUSIVE_PAYMENT_FLAG	
TAX_REPORTING_SITE	PO_VENDOR_SITES_ALL	TAX_REPORTING_SITE_FLAG	
VALIDATION_NUMBER	PO_VENDOR_SITES_ALL	VALIDATION_NUMBER	
EXCLUDE_FREIGHT_FROM_DISCOUNT	PO_VENDOR_SITES_ALL	EXCLUDE_FREIGHT_FROM_DISCOUNT	Exclude supplier freight from discount amount.
VAT_REGISTRATION_NUMBER	PO_VENDOR_SITES_ALL	VAT_REGISTRATION_NUM	
OFFSET_VAT	PO_VENDOR_SITES_ALL	OFFSET_VAT_CODE	Offset VAT code.
ORGANIZATION_NAME	ORG_ORGANIZATION_DEFINITIONS	ORGANIZATION_NAME	Name of organization corresponding to the operating unit.
ORGANIZATION_CODE	ORG_ORGANIZATION_DEFINITIONS	ORGANIZATION_CODE	Organization code for operating unit.
CHECK_DIGITS	PO_VENDOR_SITES_ALL	CHECK_DIGITS	Check number used by Account Payables department.
BANK_NUMBER	PO_VENDOR_SITES_ALL	BANK_NUM	Supplier's bank number.
ADDRESS_LINE4	PO_VENDOR_SITES_ALL	ADDRESS_LINE4	Supplier site address line 4.

Field Name	Oracle Applications Table/View Name	Column Name	Description
COUNTY	PO_VENDOR_SITES_ALL	COUNTY	
ADDRESS_STYLE	PO_VENDOR_SITES_ALL	ADDRESS_STYLE	
ALLOW_AWT	PO_VENDOR_SITES_ALL	ALLOW_AWT_FLAG	Allow withholding tax.
WITHHOLDING_TAX_GROUP	AP_AWT_GROUPS	NAME	Withholding tax group name.
ALTERNATE_VENDOR_SITE_CODE	PO_VENDOR_SITES_ALL	VENDOR_SITE_CODE_ALT	
ALTERNATE_ADDRESS_LINE	PO_VENDOR_SITES_ALL	ADDRESS_LINES_ALT	
AP_TAX_ROUNDING_RULE	PO_VENDOR_SITES_ALL	AP_TAX_ROUNDING_RULE	Tax rounding rule.
AUTO_TAX_CALCULATION	PO_VENDOR_SITES_ALL	AUTO_TAX_CALC_FLAG	
AUTO_TAX_CALCULATION_OVERRIDE	PO_VENDOR_SITES_ALL	AUTO_TAX_CALC_OVERRIDE	
AMOUNT_INCLUDES_TAX	PO_VENDOR_SITES_ALL	AMOUNT_INCLUDES_TAX_FLAG	
BANK_CHARGE_BEARER	PO_VENDOR_SITES_ALL	BANK_CHARGE_BEARER	Indicates whether this supplier bears bank charges.
BANK_BRANCH_TYPE	PO_VENDOR_SITES_ALL	BANK_BRANCH_TYPE	Identifies the list that contains the bank routing number. Use ABA, CHIPS, SWIFT, or OTHER.
PAY_ON_CODE	PO_VENDOR_SITES_ALL	PAY_ON_CODE	
PAY_ON_RECEIPT_SUMMARY_CODE	PO_VENDOR_SITES_ALL	PAY_ON_RECEIPT_SUMMARY_CODE	Identifies how to consolidate receipts to create invoices.

1.2.1. SITE_BANK_ACCOUNTS

Field Name	Oracle Applications Table/View Name	Column Name	Description
ROW_ID		ROW_ID	Unique row identifier.
VENDOR_ID	AP_BANK_ACCOUNTS_ALL	VENDOR_ID	Unique vendor identifier in Oracle Applications.
VENDOR_SITE_ID	AP_BANK_ACCOUNT_USES_ALL	VENDOR_SITE_ID	Vendor site unique identifier.
BANK_ACCOUNT_NAME	AP_BANK_ACCOUNTS_ALL	BANK_ACCOUNT_NAME	
BANK_ACCOUNT_NUM	AP_BANK_ACCOUNTS_ALL	BANK_ACCOUNT_NUM	
CURRENCY_CODE	AP_BANK_ACCOUNTS_ALL	CURRENCY_CODE	
BANK_NAME	AP_BANK_BRANCHES	BANK_NAME	
BANK_NUMBER	AP_BANK_BRANCHES	BANK_NUMBER	
BANK_BRANCH_NAME	AP_BANK_BRANCHES	BANK_BRANCH_NAME	
BANK_NUM	AP_BANK_BRANCHES	BANK_NUM	
END_DATE	AP_BANK_ACCOUNT_USES_ALL	START_DATE	
START_DATE	AP_BANK_ACCOUNT_USES_ALL	END_DATE	
PRIMARY	AP_BANK_ACCOUNT_USES_ALL	PRIMARY_FLAG	Indicates primary bank account.

Field Name	Oracle Applications Table/View Name	Column Name	Description
ORG_ID	AP_BANK_ACCOUNT_USES_ALL	ORG_ID	Organization identifier.
ORGANIZATION_CODE	ORG_ORGANIZATION_DEFINITIONS	ORGANIZATION_CODE	
ORGANIZATION_NAME	ORG_ORGANIZATION_DEFINITIONS	ORGANIZATION_NAME	

1.2.2. SUPPLIER_CONTACTS

Field Name	Oracle Applications Table/View Name	Column Name	Description
VENDOR_SITE_ID	PO_VENDOR_CONTACTS	VENDOR_SITE_ID	Unique vendor site identifier.
FIRST_NAME	PO_VENDOR_CONTACTS	FIRST_NAME	
MIDDLE_NAME	PO_VENDOR_CONTACTS	MIDDLE_NAME	
LAST_NAME	PO_VENDOR_CONTACTS	LAST_NAME	
PREFIX	PO_VENDOR_CONTACTS	PREFIX	
TITLE	PO_VENDOR_CONTACTS	TITLE	
MAIL_STOP	PO_VENDOR_CONTACTS	MAIL_STOP	
AREA_CODE	PO_VENDOR_CONTACTS	AREA_CODE	
PHONE	PO_VENDOR_CONTACTS	PHONE	
INACTIVE_DATE	PO_VENDOR_CONTACTS	INACTIVE_DATE	Date from which the contact is inactive.

Project Predefined Transaction Services

■ Overview	472
■ Receive Labor Service	473

Overview

This chapter describes the predefined transaction services provided in the Oracle Applications Adapter’s 10.7SC Project package.

The table below shows the predefined transaction services organized by Oracle Applications module. This chapter presents the transactions in alphabetical order.

Oracle Applications Module	Predefined Transactions
Project Accounting	■ “Receive Labor Service” on page 473

For more information about using the predefined transaction services, see [Chapter 1, “Predefined Transaction Services” on page 17](#).

Receive Labor Service

The name of this service is:

WmOAPRJ107SC.projectAccounting107SC.intoOA.labor:receiveLabor

This service imports expenditures from external cost collection systems, which have the following expenditure type classes:

- Straight Time (Labor Hours)
- Expense Reports
- Usages
- Inventory
- Work in Progress
- Miscellaneous
- Supplier Invoices

You can load items as costed (that only have a quantity) or uncosted (that have a quantity and raw cost). You can also load them as unaccounted (when General Ledger accounts are undetermined) or accounted (when costs will not be transferred to the General Ledger).

Database Scripts

This service uses the following database scripts:

Database Script	Description
wm_install_into_labor.sql	Runs all the scripts listed below, except the uninstall script.
wm_into_labor_pkg.sql	Installs WM_LABOR_IMP_HANDLER_PKG. WM_HANDLE_LABORTXN, which calls the WM_CONC_REQUEST.WM_REQUEST_SUBMIT procedure to submit the labor import process. The process in the Project Accounting module is PRC: Transaction Import.
wm_into_labor_seq.sql	Installs WM_PA_GROUP_S sequence, which generates BATCH_ID.
wm_drop_into_labor.sql	Uninstalls all components created by wm_install_into_labor.sql.

For more information about using database scripts, see [“Database Scripts”](#) on page 26.

Supporting Transaction Definitions

This service was built from the configured services defined in the following transaction definition:

- `setLaborTxn107SC.txp`

For information about using the transaction definition files to customize this service, see [“Transaction Definitions” on page 25](#).

Flow Control

The main flow executes as follows:

- `specifyDefaultSettings` specifies the default parameter settings required for service execution. You should change these settings accordingly.
- `specifyConcProgParams` specifies the default parameter settings required for concurrent program execution. You should change these settings accordingly.
- `generateBatchName` gets the `BATCH_NAME` for the current set of Labor records.
 - `pickSequence` generates the current sequence number from the sequence, `WM_PA_GROUP_S`.
- `bizDocMapping` maps the incoming business document structure to the required Oracle Applications data structures (interface tables).
 - `getOrgId` is a transformer for mapping the business document `IData` structure to the interface table `IData` structure. It takes `ORGANIZATION_NAME` as the input parameter and queries the table `ORG_ORGANIZATION_DEFINITIONS` and gets the `ORGANIZATION_ID` corresponding to the `ORGANIZATION_NAME`.
 - `getCodeCombinationId` is a transformer for mapping the business document `IData` structure to the interface table `IData` structure. This service gets `CODE_COMBINATION_ID` from `GL_CODE_COMBINATIONS_KFV` given the `CONCATENATED_SEGMENTS`.
 - `getSystemLinkage` is a transformer for mapping the business document `IData` structure to the interface table `IData` structure. It takes `EXPENDITURE_CLASS` as the input parameter and queries the table `PA_SYSTEM_LINKAGES` to get `SYSTEM_LINKAGE`.
- `setLaborTxn` inserts data into the interface table. It extracts data from the `IData` structure resulted in the `bizDocMapping` service and puts the data into the interface table in Oracle Applications for Labor.
- `importLabor` imports data to the production table from the interface table. It calls the `execLaborConcProg`, `checkLaborImportStatus`, and `getLaborImport_ERR` services to execute the corresponding concurrent program that inserts data into the production table and to generate the error or acknowledgement message. If the execution of

executeConcProgram returns the status of SUCCESS, it checks for the record with the current BATCH_NAME in the interface table. If any are found, it then indicates an error during import. In this case, this service calls getLaborImport_ERR to retrieve the errors. If no record is found, it comes out of the flow indicating success of the data import process.

- **execLaborConcProg** invokes the stored procedure WM_LABORTXN_IMP_HANDLER_PKG.WM_HANDLE_LABORTXN. This procedure calls the corresponding concurrent subroutine to execute the data import process for Labor into Oracle Applications. This service produces Status ID, Request ID, Execution Status Message for normal concurrent program completion, and a database Stored Procedure error message (if an exception occurs in the Stored Procedure execution).
- **checkLaborImportStatus** checks execution status by checking the interface table for any rejected records corresponding to the current BATCH_NAME. If it does not return any rows, the import was successful. If it returns a row, the concurrent program could not import data successfully in the Oracle Applications production tables.
- **getLaborImport_ERR** gets the error message that occurs during data import to the production table from the interface table. Based on the parameter BATCH_NAME, it scans the PA_TRANSACTION_INTERFACE_ALL table to get the corresponding message matches to the BATCH_NAME.

For more details on receive transactions, see [“Overview of Receive Service Transaction Processing” on page 31](#).

Business Document Structure

This service uses the business document LaborBizDoc. Its structure is as follows:

- LABOR_BIZDOC

LABOR_BIZDOC (Maps to PA_TRANSACTION_INTERFACE_ALL)

Field Name	Maps to Column	Description
TRANSACTION_SOURCE	TRANSACTION_SOURCE	Required. Define the Source of the transaction in the Transaction Sources window in Oracle Projects. Examples values are Default Expenditure Type Class, GL Account, Costed, or Uncosted.
EXPENDITURE_ENDING_DATE	EXPENDITURE_ENDING_DATE	Required. Last day of the expenditure week.

Field Name	Maps to Column	Description
EMPLOYEE_NUMBER	EMPLOYEE_NUMBER	Required for Expenditure Type class as Labor and Expense Reports. Employee number of the employee who incurred charges.
CHARGE_ORGANIZATION_NAME	ORGANIZATION_NAME	Name of the organization that incurred the charge for this transaction
EXPENDITURE_ITEM_DATE	EXPENDITURE_ITEM_DATE	Required. Date when the transaction occurred.
PROJECT_NUMBER	PROJECT_NUMBER	Required. Project number that incurred the expenditure.
TASK_NUMBER	TASK_NUMBER	Required. Task number for the project.
EXPENDITURE_TYPE	EXPENDITURE_TYPE	Required. Expenditure that classifies this transaction. The expenditure type and expenditure type class must have a valid combination in the PA_EXPEND_TYP_SYS_LINKS table.
NON_LABOR_RESOURCE	NON_LABOR_RESOURCE	Non-labor resource for the transaction.
NON_LABOR_RESOURCE_ORG_NAME	NON_LABOR_RESOURCE_ORG_NAME	Name of the organization owning the non-labor resource.
QUANTITY	QUANTITY	Required. Number of units for the transaction based on the expenditure type.
RAW_COST	RAW_COST	Raw cost amount
EXPENDITURE_COMMENT	EXPENDITURE_COMMENT	Description for the transaction.
TRANSACTION_STATUS_CODE		
ORIG_TRANSACTION_REFERENCE	ORIG_TRANSACTION_REFERENCE	Required. Unique reference to identify this record.
RAW_COST_RATE	RAW_COST_RATE	Raw cost rate of the costed transaction

Field Name	Maps to Column	Description
UNMATCHED_NEGATIVE_TXN_FLAG	UNMATCHED_NEGATIVE_TXN_FLAG	Use Y to indicate that the transaction is an unmatched negative transaction and is not adjusted against an expenditure item.
ORGANIZATION_NAME	ORG_ID	Organization name for the transaction. Populates the ORG_ID for a multi-organization setup. The getOrgId service is invoked to get the ORGANIZATION_ID.
DR_CODE_COMBINATION	DR_CODE_COMBINATION_ID	Populates the DR_CODE_COMBINATION_ID for accounted transactions. The getCodeCombinationID service is invoked to get the code combination ID.
CR_CODE_COMBINATION	CR_CODE_COMBINATION_ID	Populates the CR_CODE_COMBINATION_ID for accounted transactions. The getCodeCombinationID service is invoked to get the code combination ID.
CDL_SYSTEM_REFERENCE1	CDL_SYSTEM_REFERENCE1	Use the external system to drill down to the transaction in the original system.
CDL_SYSTEM_REFERENCE2	CDL_SYSTEM_REFERENCE2	
CDL_SYSTEM_REFERENCE3	CDL_SYSTEM_REFERENCE3	
GL_DATE	GL_DATE	General ledger date if the transaction has already been accounted.
BURDENED_COST_RATE	BURDENED_COST_RATE	Burdened cost multiplier that Oracle Applications uses for reporting purposes.

Field Name	Maps to Column	Description
EXPENDITURE_CLASS	SYSTEM_LINKAGE	Fetches the System Linkage function from PA_SYSTEM_LINKAGES using the Meaning column as the expenditure class. It also populates SYSTEM_LINKAGE column with this value. The combination of the Expenditure Type and System Linkage function must be defined in Oracle Projects. The getSystemLinkage service is invoked to get the expenditure class.
USER_TRANSACTION_SOURCE	USER_TRANSACTION_SOURCE	Must be defined in PA_TRANSACTION_USER.USER_TRANSACTION_SOURCE.
RECEIPT_CURRENCY_AMOUNT	RECEIPT_CURRENCY_AMOUNT	Amount of the expenditure in original currency.
RECEIPT_CURRENCY_CODE	RECEIPT_CURRENCY_CODE	Currency Code for receipt currency.
RECEIPT_EXCHANGE_RATE	RECEIPT_EXCHANGE_RATE	Exchange rate to convert from the receipt currency to the transaction currency (reimbursement currency).
DENOM_CURRENCY_CODE	DENOM_CURRENCY_CODE	Currency Code for the transaction currency.
DENOM_RAW_COST	DENOM_RAW_COST	Raw cost amount in the transaction currency.
DENOM_BURDENED_COST	DENOM_BURDENED_COST	Burdened cost amount in the transaction currency.
ACCT_RATE_DATE	ACCT_RATE_DATE	Exchange rate for converting to the functional currency.
ACCT_RATE_TYPE	ACCT_RATE_TYPE	Conversion type for converting to the functional currency.
ACCT_EXCHANGE_RATE	ACCT_EXCHANGE_RATE	Exchange rate for converting to the functional currency.

Field Name	Maps to Column	Description
ACCT_RAW_COST	ACCT_RAW_COST	Raw cost in functional currency. For accounted transactions, the import transaction compares this value to the value calculated from DENOM_RAW_COST to ensure it is within the ACCT_EXCHANGE_ROUNDING_LIMIT.
ACCT_BURDENED_COST	ACCT_BURDENED_COST	The burdened cost in the functional currency.
ACCT_EXCHANGE_ROUNDING_LIMIT	ACCT_EXCHANGE_ROUNDING_LIMIT	Functional currency rounding limit. If null, then the rounding limit is 0.
PROJECT_RATE_DATE	PROJECT_RATE_DATE	Exchange rate date for converting to project currency.
PROJECT_RATE_TYPE	PROJECT_RATE_TYPE	Conversion rate type used to convert to the project currency.
PROJECT_EXCHANGE_RATE	PROJECT_EXCHANGE_RATE	The exchange rate for converting to project currency.
ORIG_EXP_TXN_REFERENCE1	ORIG_EXP_TXN_REFERENCE1	Expenditure identifier in the external system.
ORIG_EXP_TXN_REFERENCE2	ORIG_EXP_TXN_REFERENCE2	Columns provided for additional grouping.
ORIG_EXP_TXN_REFERENCE3	ORIG_EXP_TXN_REFERENCE3	Used for additional grouping.
ORIG_USER_EXP_TXN_REFERENCE	ORIG_USER_EXP_TXN_REFERENCE	Expenditure identifier in the external system (User Reference).
VENDOR_NUMBER	VENDOR_NUMBER	For supplier invoices.
OVERRIDE_TO_ORGANIZATION_NAME	OVERRIDE_TO_ORGANIZATION_NAME	Overrides ORGANIZATION_NAME.
REVERSED_ORIG_TXN_REFERENCE	REVERSED_ORIG_TXN_REFERENCE	Reference identifier of the original transaction that this transaction reverses.
BILLABLE_FLAG	BILLABLE_FLAG	Billable or capitalizable flag.

Index

A

- Accounts Payable Invoice
 - query service 53
 - receive service 65
 - send service 131
- Accounts Payable Payment
 - query service 54
 - send service 139
- Accounts Receivable Transactions
 - query service 55
 - send service 148
- adapter package
 - name 19
- Auto Invoice
 - receive service 71

B

- Bank Statement
 - receive service 100
- Bills of Material
 - query service 243, 247
 - receive service 252
- Bills of Material module, list of services 233
- Bills of Material Revisions
 - query service 234
- Bills of Material routing
 - query service 236
- Bills of Material, Engineering
 - send service 325
- Bills of Material, Manufacturing
 - send service 359
- BOM Revisions
 - query service 234
- BOM routing
 - query service 236
- BOMs
 - query service 247
 - receiving bills and routing 252

- BOMs, Engineering
 - query service 243
 - send service 325
- BOMs, Manufacturing
 - send service 359
- built-in services
 - description of 19
- business documents
 - for predefined transaction services 30

C

- Cash Management module, list of services 51
- Catalog, Product
 - query service 411
- connection
 - default 33, 36, 42
- conventions used in this document 15
- Cross Reference, Customer Item
 - receive service 271
- custom tables
 - for predefined transaction services 43
- Customer
 - receive service 104
 - send service 179
- Customer Item
 - receive service 266
- Customer Item Cross Reference
 - receive service 271

D

- database scripts
 - description of 26
 - where they are installed 26
- default connection 33, 36, 42
- documentation
 - additional 16
 - conventions used 15
 - feedback 16

E

Employee
 receive service 223
Engineering Bills of Material
 query service 243
 send service 325
Engineering BOMS
 send service 325
Engineering BOMs
 query service 243
Engineering module, list of services 233

F

FA Budget
 receive service 115
Financials package, list of services 51
Fixed Assets module, list of services 51

G

General Ledger Balance
 query service 59
General Ledger Budget
 receive service 118
General Ledger module, list of services 51

H

Human Resources module, list of services 222

I

installing Oracle Applications components for predefined
 transaction services 28
Inventory
 Customer Item Cross Reference
 receive service 271
 Customer Item, receive service 266
 Inventory Transaction, receive service 275
 Item Category, query service 245
 Item, query service 244
 Item, receive service 284
 Item, send service 336
 On-hand Quantity, query service 249

Replenishment Request, receive service 314
Transaction, receive service 275

Inventory module, list of services 233

Inventory Transaction
 receive service 275

Item

 On-hand Quantity, query service 249
 query service 244
 receive service 284
 send service 336

Item Category
 query service 245

Item, Customer
 receive service 266

Item, Customer Cross Reference
 receive service 271

Item, Moved
 receive service 306

J

Journal
 query service 62
 receive service 121
 send service 199

L

Labor
 receive service 473
location of database scripts 26

M

Manufacturing Bills of Material
 query service 247
 send service 359
Manufacturing BOMs
 query service 247
 send service 359
Manufacturing package, list of services 233
Mass Addition
 receive service 126
Master Scheduling
 receive service 303

Master Scheduling module, list of services 233
modifying business documents for predefined transaction services 30
modifying transaction definitions 26
Move Transaction
 receive service 306

N

name of adapter package 19
naming conventions
 for predefined transaction services 23

O

On-hand Quantity
 query service 249
optimizing
 receive services 34
Oracle Applications database components
 description of 26
Oracle Applications modules
 folders in Developer 23
Order Management module, list of services 372

P

packages
 for predefined transaction services, description of 19
 Oracle Applications Adapter
 description of 19
packages for predefined transaction services 20
Payables module, list of services 52
Pick Details
 query service 373
 send service 387
predefined transaction service packages 20
predefined transaction services
 business documents for 30
 custom tables for 43
 description of 18
 installing Oracle Applications components 28
 naming conventions 23
 packages 19
 removing Oracle Applications components 29
 transaction definitions for 25

 types of 21
 uninstalling Oracle Applications components 29
Procurement package, list of services 410
Product Catalog
 query service 411
program code conventions in this document 15
Project Accounting module, list of services 472
Purchase Order
 query service 418, 420
 send service 434
Purchasing
 Product Catalog, query service 411
 Purchase Order Status, query service 420
 Purchase Order, query service 418
 Purchase Order, send service 434
 Requisition, receive service 425
 Supplier, query service 424
 Supplier, send service 451
Purchasing module
 list of services 410

Q

Quality Collection
 receive service 310
Quality Collection module, list of services 233
Quality results
 receive service 310
query and send predefined transaction services
 defined 18
query services
 business documents 30
 description of 20, 35
 transaction processing for 35
 using 36
queryAPIInvoice service 53
queryAPPayment service 54
queryARTransaction service 55
queryBOMRevision service 234
queryBOMRouting service 236
queryEngineeringBOM service 243
queryGLBalance service 59
queryItem service 244
queryItemCategory service 245

- queryJournal service 62
- queryManufacturingBOM service 247
- queryOnHandQuantity service 249
- queryPickDetail service 373
- queryPO service 418
- queryPOStatus service 420
- queryProductCatalog service 411
- querySalesOrder service 374
- querySupplier service 424
- queryVendor service 64

R

- Receivables module, list of services 52
- receive predefined transaction services
 - defined 18
- receive services
 - business documents 30
 - description of 19, 31
 - optimizing 34
 - transaction processing for 31
 - using 33
- receiveAPIInvoice service 65
- receiveAutoInvoice service 71
- receiveBankStatement service 100
- receiveBillsAndRoute service 252
- receiveCICrossRef service 271
- receiveCustomer service 104
- receiveCustomerItem service 266
- receiveEmployee service 223
- receiveFABudget service 115
- receiveGLBudget service 118
- receiveInvTransaction service 275
- receiveItem service 284
- receiveJournal service 121
- receiveLabor service 473
- receiveMassAdditions service 126
- receiveMasterSchedule service 303
- receiveMoveTransaction service 306
- receiveQualityCollection service 310
- receiveReplenishment service 314
- receiveRequisition service 425
- receiveResourceTransaction service 320
- receiveSalesOrder service 375

- removing Oracle Applications components for predefined transaction services 29
- Replenishment Request
 - receive service 314
- Requisition
 - receive service 425
- Resource Transaction
 - receive service 320
- Revision, Bills of Material
 - query service 234

S

- Sales Order
 - query service 374
 - receive service 375
 - send service 394
- Scheduling, Master
 - receive service 303
- send services
 - business documents 30
 - description of 20, 37
 - transaction processing for 38
 - using 42
 - using in debug mode 43
- sendAPIInvoice service 131
- sendAPPayment service 139
- sendARTransaction service 148
- sendCustomer service 179
- sendEngineeringBOM service 325
- sendItem service 336
- sendJournal service 199
- sendManufacturingBOM service 359
- sendPickDetail service 387
- sendPO service 434
- sendSalesOrder service 394
- sendSupplier service 451
- sendVendor service 204
- services
 - default connection 33, 36, 42
- Shipping module, list of resources 372
- Supplier
 - query service 424
 - send service 451

T

transaction definitions

- definition of 25
- modifying 26
- where they are installed 26

transaction processing

- for query services 35
- for receive services 31
- for send services 38

transaction services, predefined

- description of 18
- Oracle Applications versions supported 18
- packages 19

troubleshooting information 16

types of predefined transaction services 21

typographical conventions in this document 15

U

uninstalling Oracle Applications components for predefined

- transaction services 29

updating transaction definitions 26

using

- query services 36
- receive services 33
- send services 42
- send services in debug mode 43

V

Vendor

- query service 64
- send service 204

W

WM_CONTROL table

- description of 47

WM_TRACKCHANGES table

- description of 43

Work in Process

- moved items, receive service 306
- Resource Transaction, receive service 320

Work in Process module, list of services 233

