# IBM Maximo Enterprise Adapter for Oracle Applications System Administration Guide

Version 7.6

**Note**

Before using this information and the product it supports, read the information in “Notices” on page 207

This edition applies to version 7, release 6, modification 0 of IBM Maximo Enterprise Adapter for Oracle Applications and to all subsequent releases and modifications until otherwise indicated in new editions.

# Contents

# Contents

# Contents

# Contents

# Contents

# Contents

# Integration framework architecture

# 1

IBM® Maximo® Enterprise Adapter for Oracle Applications is a set of applications and predefined components that help you to integrate Maximo Asset Management with Oracle E-Business Suite. With this adapter, you can create business flows between Maximo Asset Management and your other business applications.

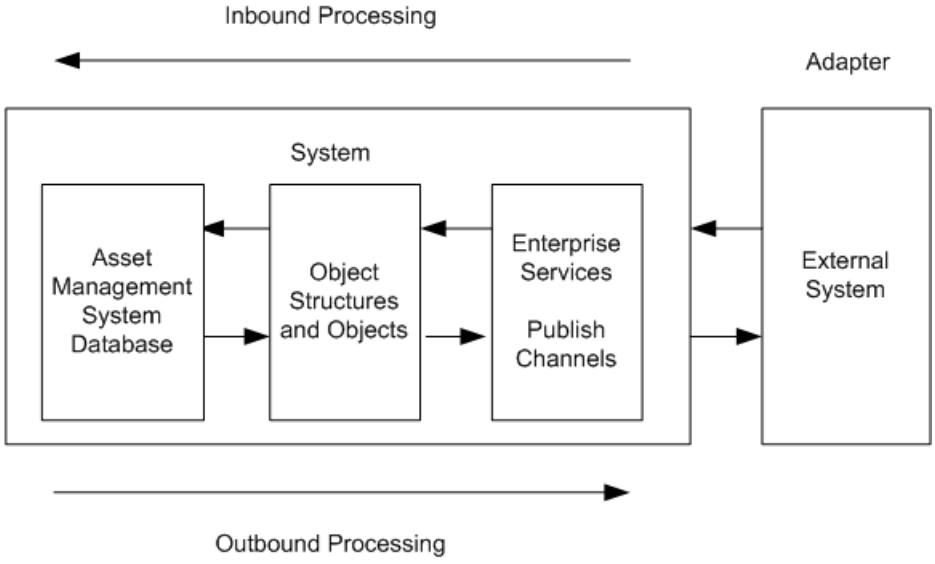
The installation of Maximo Enterprise Adapter for Oracle Applications customizes the integration framework for Oracle E-Business Suite.

Anyone involved in the implementation or day-to-day administration and use of the Maximo Enterprise Adapter for Oracle Applications should familiarize themselves with the integration framework architecture.

# Overview

The integration framework facilitates data exchange between Maximo Asset Management and external applications or systems in a real-time mode. Java™ procedures perform the integration processing, and you can customize the processing by using Java user exit procedures.

The following diagram shows how data is exchanged by using enterprise services and publish channels, each of which acts as a communication channel between Maximo Asset Management and external system. Data that the asset management software receives from external systems is inbound data. Data that the asset management software sends to external systems is outbound data.

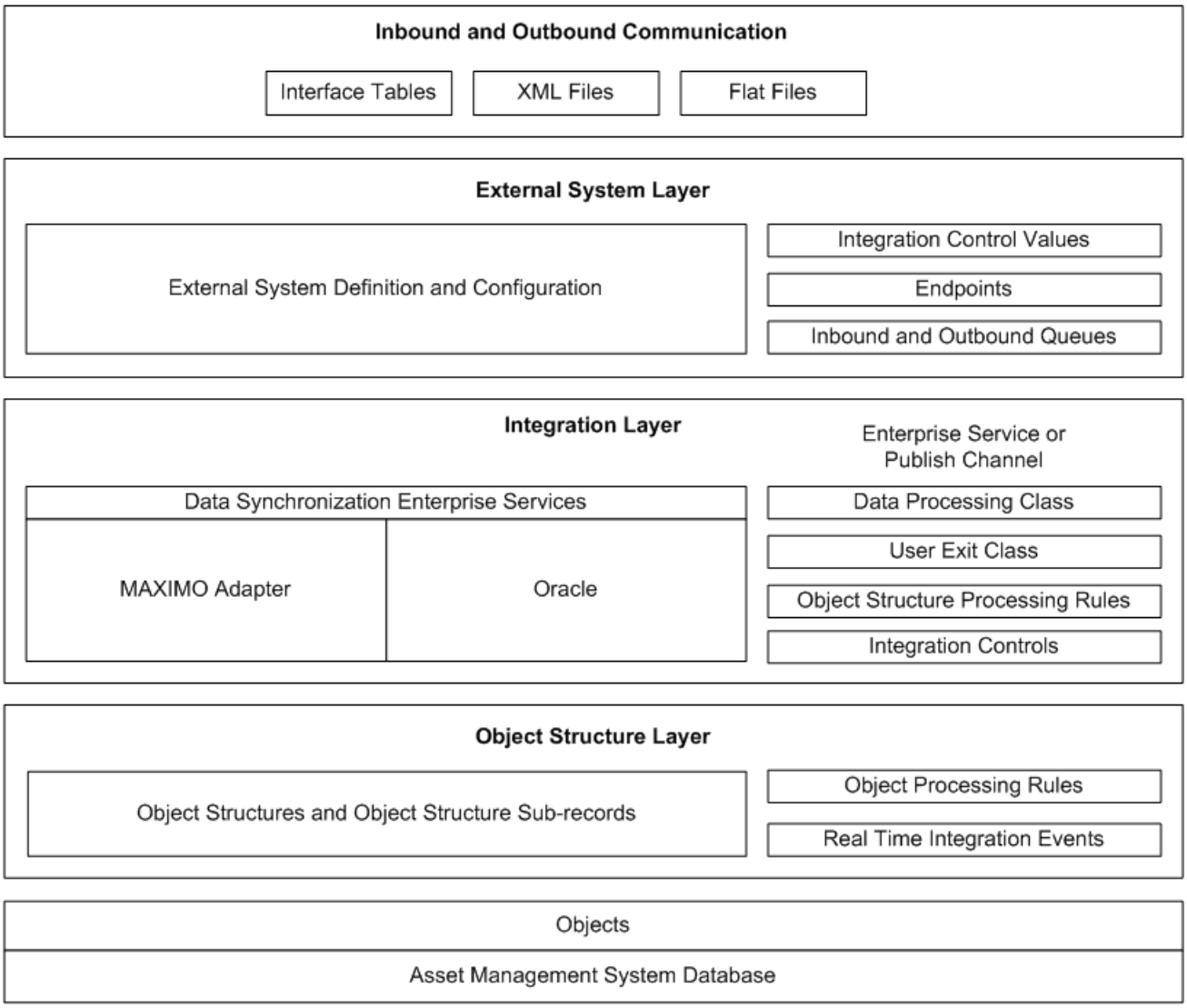


The following table describes the three integration processing layers.

| Layer | Description |
| --- | --- |
| Object structure | Create and manage object structures. Each object structure is built from one or more objects that provide the content that the enterprise services or publish channels need. |
| Integration | Create and manage enterprise services and publish channels, including business rules processing and integration controls. Enterprise services process integration framework messages in an inbound direction. Publish channels process integration framework messages in an outbound direction. Each enterprise service and publish channel can have a processing class, a user exit class, and processing rules. |
| External system | Create and manage external systems and their publish channels and enterprise services, which includes:<br>• Defining external systems that exchange data with the system<br>• Identifying the specific enterprise services and publish channels that are applicable to each external system for either inbound or outbound processing<br>• Setting up integration control values where applicable for each enterprise service or publish channel<br>• Identifying queue parameters for the system and the communication method that is used for sending data to the system |

In addition to the three layers, the integration framework includes specific entities for outbound and inbound communication: interface tables, XML files, and flat files.

The following diagram shows the three integration framework architecture processing layers.



# The object structure layer

The object structure layer interacts with the business objects and facilitates their creation and maintenance.

## Object structures and object structure sub-records

An object structure consists of one or more sub-records that correspond to business objects. During inbound processing, objects are created and the object fields are populated from the corresponding sub-record fields in the object structure before applying standard system application processing. During outbound processing, the system populates the sub-record fields from the corresponding fields in the original object. Except for certain generic integration fields, system objects are not updated in outbound messages.

When defining processing rules, use the following guidelines to determine if you should apply the rule to an object structure sub-record or to an object:

- In the outbound direction, you can apply processing rules to the object structure sub-records only.
- In the inbound direction, you can apply processing rules to object structure sub-records or objects.
    - If an inbound rule changes the key field value of an object, apply it to the object structure sub-record.
    - If an inbound rule does not evaluate or manipulate an object or object set, apply it to the object structure sub-record.
    - If an inbound rule evaluates or manipulates a user-defined field, apply it to the object structure sub-record.
    - If an inbound rule evaluates or manipulates an object or object field, apply it to the object.

Apply all rules for enterprise services to either objects or to object structure sub-records. Avoid creating rules for both objects and object structure sub-records. If you define processing rules for both objects, processing time for inbound messages increase.

### Processing rules

A processing rule defines an action that is performed on a field in a sub-record or object, or on the sub-record or object itself. You define an inbound processing rule on the Enterprise Service tab in the Enterprise Services application. You define an outbound processing rule on the Publish Channel tab in the Publish Channels application.

After you select the applicable service or channel, the application displays the sub-records that comprise the corresponding object structure. You must select the sub-record on which the processing rule applies.

# Real-time integration events

During the creation of an outbound publish channel, an event listener is registered on the primary object of the object structure. When the listener is enabled, it monitors the system for activity on the corresponding object. Whenever any instance of that object is created, updated, or deleted, outbound integration processing is initiated for all publish channels.

# The integration layer

The integration layer consists of adapters, enterprise services, and publish channels. Use adapters to group enterprise services and publish channels to meet your transaction needs. With enterprise services and publish channels, you can receive data from and send data to multiple external systems and applications.

## Adapters

All enterprise services and publish channels are defined within an adapter, which is a set of related programs, mappings, and controls.

The default adapter is the MAXIMO adapter. If necessary, you can add new enterprise services and publish channels to the existing adapter and create adapters. First determine whether you must create an adapter or you can use the predefined MAXIMO adapter.

Adapters have the following characteristics:

- Services and channels can be grouped within an adapter.
- Any adapter can work with interface tables and Web services.

You maintain adapters in the Adapter Domain window which you access in the Domains application. You can create multiple adapters, if necessary.

## Enterprise services

The enterprise service is a pipeline for querying system data and importing data into the system from an external system. You can configure enterprise services to process data synchronously (without a queue) or asynchronously (with a queue). Enterprise services can use multiple protocols such as Web service and HTTP.

The enterprise service has data processing layers transform data and apply business processing rules to data before it reaches the system objects. When the inbound message reaches the object structure layer, the XML message must be in the format of the object structure schema. The message can then be processed successfully.

The enterprise service can use the following processing layers:

- Processing rules – The integration framework provides a rule engine where you can filter and transform the XML message.
- User exit – Represents a Java class that you can use to filter, transform data, and implement business logic. You can use this class as part of an installation-customization.
- Data processing class – Represents a Java class that you can use to filter, transform data, and implement business logic. Adapters for Oracle and SAP® provide processing classes to support integration to these products.
- XSL map – Represents an XSLT style sheet that you can use to transform data and perform mapping of the XML message to another format.

# Publish channels

A publish channel is the pipeline for sending data asynchronously from the asset management software to an external system. Events that initiate publish channel processing are object events (insert, update, and delete), application-initiated calls, and data export.

The content of a publish channel XML message is based on the associated object structure. When you trigger publish channel processing, the integration framework builds the XML message based on the object structure. The system then moves the message through multiple processing layers before placing the message into a queue and releasing the initiator of the transaction. The publish channel can use the following processing layers:

- Processing rules – The integration framework provides a rules engine where you can filter and transform the XML message. You can implement rules in the Publish Channel application.
- User exit – Represents a Java class that you can use to filter data, transform data, and implement business logic. You can use this class as part of an installation-customization.
- Data processing class – Represents a Java class that you can use to filter, transform data, and implement business logic. Adapters for Oracle and SAP provide processing classes to support integration to these products.
- XSL map – Represents an XSLT style sheet that you can use to transform data and perform mapping of the XML message to another format.

After the system places the message into the queue, a polling thread (the system cron task) picks up the message and sends it to an external system through a configured endpoint. The endpoint identifies the protocol that the system uses to send data, such as HTTP or Web service. The endpoint also identifies the property values that are specific to that endpoint, such as URL, user name, and password.

## Operation type

Every enterprise service and publish channel has an operation type that indicates the purpose of the transaction.

In an enterprise service, you can select an operation type from a list of system options. The operations that are available for your selection depend upon the value you have specified in Query Only check box for the associated object structure.

Maximo Enterprise Adapter for Oracle Applications uses only the SYNC operation.

### Data formats

Maximo Asset Management and external systems can exchange data transactions by using XML messages, interface tables, and flat files. Maximo Asset Management accepts XML and interface table transactions for the real-time exchange of data and flat files for the bulk import and export of non-system data.

**Integration XML**

Integration XML is the XML representation that the objects recognize. Maximo Asset Management writes all outbound XML messages in this format and requires that all inbound non-system XML messages be converted to this format. If an external system uses another XML format, you must provide Java code or XSL style sheets to convert the data.

**Interface tables**

Interface tables are relational database tables that you can use in place of XML messages to transfer data between Maximo Asset Management and external systems. Each table contains the same data fields as the corresponding integration XML, in a flat, non-hierarchical format.

Maximo Enterprise Adapter for Oracle Applications uses interface tables.

**Flat files**

A flat file is a non-hierarchical, non-relational representation of the data columns in enterprise services, publish channels, or interface tables. You can use flat files to load master data into Maximo Asset Management, and to perform the import and export of non-system data to and from the asset management software.

You configure the object structure that is associated with the enterprise service or publish channel so that Maximo Asset Management can assign interface table names. You do so by selecting the Support Flat Structure check box on the Object Structure tab of the Object Structures application.

## Integration controls

With integration controls, you can configure the behavior of any enterprise service or publish channel according to the requirements of individual organizations and sites. Both processing rules and Java classes can access integration controls for evaluation purposes.

Integration controls are defined at the system level. You can assign controls to multiple enterprise services and publish channels. The control values can be configured at the external system level. Two external systems that process the same enterprise service can share the same processing logic, class files, and processing rules, yet process the data differently due to different control settings.

With the integration framework, you can create four types of integration controls:

| Control type | Description |
|---|---|
| Boolean | Specifies a value of 0 (false) or 1 (true) |
| Cross-reference | Replaces one value with another |
| List | Specifies a list of values |
| Value | Specifies a single value |

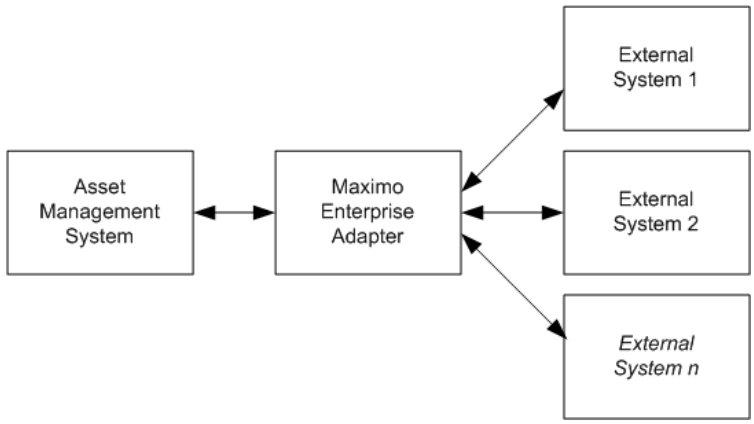
# The external system layer

Any business application that sends data to Maximo Asset Management or receives data from the asset management software is considered to be an external system. The external system layer enables the flow of data between Maximo Asset Management and external applications by defining the location and characteristics of external systems, and identifying the adapter, enterprise services, or publish channels that each external system uses.

## External system definition and configuration

An external system interacts with Maximo Asset Management, either as an endpoint (location) to which Maximo Asset Management sends outbound data, or as a source from which the asset management software receives inbound data. An external system can process inbound enterprise services, outbound publish channels, or a combination of both. Maximo Asset Management can be integrated with any number of external systems.

The following diagram shows how external systems interact with Maximo Asset Management. Each external system uses a single adapter to exchange messages with the asset management software in the format that is identified by the adapter. Each external system also uses a single endpoint to process outbound messages.



The name that is given to the external system is the name by which Maximo Asset Management recognizes the external system. Inbound XML messages and interface table transactions must provide this name in the SenderID field. Maximo Asset Management writes this name in the RecipientID field of outbound XML and interface tables.

In outbound transactions, the SenderID is the value of MAXVARS.MXSYSID.

## Integration control values

When you create an external system, the application copies the integration controls that are defined for the corresponding publish channels and enterprise services.

You can change the default value of the controls or add new values. If the definition of a control allows organization or site overrides, specify values for specific organizations or sites.

## Endpoints

An endpoint is a location to which an outbound queue sends data. Endpoints are independent of adapters and external systems, although two systems that use different adapters generally have different endpoints. An endpoint typically has an application component that processes the data sent from Maximo Asset Management.

External systems can be associated with endpoints in multiple ways; exact usage depends on your implementation. Possible scenarios include, but are not limited to, the following scenarios:

- A single endpoint per external system

  This is the common point-to-point scenario where each individual external system has a different endpoint.

- A single endpoint supporting multiple external systems

  Multiple external system use the same set of interface tables, or multiple external systems share a single queue. In this case, configure multiple systems to use the same endpoint, and the endpoint contains multiple instances of any outbound message (one per external system).

- A single endpoint per enterprise service or publish channel

  You can define the endpoint each enterprise service or publish channel uses. If you do not define an endpoint at the enterprise service or publish channel level, Maximo Asset Management moves data to the endpoint location that you defined at the external system level.

Each endpoint is associated with a handler, which is a processing class that defines how, and in what format the data is sent from the outbound queue to an endpoint.

The following table lists the endpoints that are provided by Maximo Asset Management.

| Endpoint | Description |
|---|---|
| MXFLATFILE | Writes flat files to a prespecified directory location. |
| MXIFACETABLE | Writes outbound transactions to local interface tables. |
| MXXMLFILE | Writes XML files to a prespecified directory location. |
| MXCMDLINE | Implements the CMDL handler; takes a command and endpoint as input and uses the SSH protocol to securely invoke the command on the target system and return the results. |

# Inbound and outbound queues

A queue is a Java Message Service (JMS) queue that the system uses as a staging area during the exchange of messages between Maximo Asset Management and external systems. JMS queue capability is available within the Oracle WebLogic® Server and IBM WebSphere® Application Server environments.

Maximo Asset Management uses one queue for outbound processing and two for inbound processing. The inbound queues differ in the sequence in which they process transactions, as follows:

- The sequential inbound queue processes transactions in strict first-in-first-out order, and stops processing when it finds an error in a transaction. Yout must use this queue to process integration transactions that are dependent upon the successful processing of previous integration transactions.
- The continuous inbound queue does not process in first-in-first-out order and continues processing transactions after it encounters an error in a transaction. You must use this queue to load data that is not dependent upon the successful processing of other transactions.

Within an external system, you can use the sequential queue, the continuous queue, or you can use one queue for all inbound enterprise services.

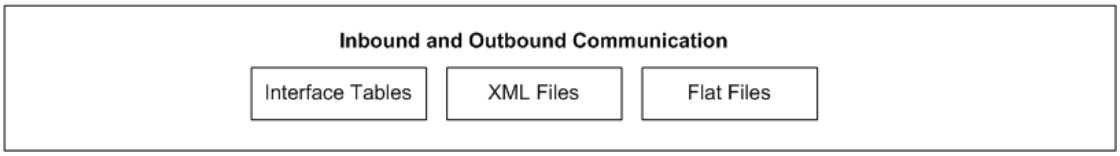
The system installation procedure installs the following JMS queues.

| Queue name | Description |
|---|---|
| cqin | Continuous inbound |
| sqin | Sequential inbound |
| sqout | Sequential outbound |
| cqinerr | Continuous inbound error queue |

For more information about JMS queues, see the JMS queue configuration information in the *IBM Maximo Asset Management Integration Guide*.

# Inbound and outbound communication

The integration framework can process inbound transactions by using flat files, XML files, and interface tables.




The integration framework sends outbound data from the outbound queue to an endpoint by using a handler. The handler is a processing class that defines how, and in what format, the data is to be delivered.

Most handlers use a set of properties, such as a specific URL, a user name and password, or a specific directory location. The values of these properties depends on the endpoint that is associated with the handler.

The asset management system provides the following predefined handlers. You can create additional ones, if necessary.

**Table 1: Handlers**

| Handler | Description |
|---|---|
| EJB | Delivers outbound data to an Enterprise JavaBean (EJB) executing in the local application server or a remote application server |
| FLATFILE | Delivers outbound data into a flat file whose location is configurable |
| HTTP | Delivers outbound data as an XML document to a URL over the HTTP or HTTPS protocols |
| IFACETABLE | Delivers outbound data into interface tables in a relational database |
| JMS | Delivers outbound data into a queuing system that has been enabled through Java Message Service (JMS) |
| WEBSERVICE | Delivers outbound data to a Web services component using SOAP over HTTP |
| XMLFILE | Delivers outbound data in XML format to a file on the local server or a shared network folder |
| CMDLINE | Implements the CMDL handler; takes a command and endpoint as input and uses the SSH protocol to securely invoke the command on the target system and return the results. |

The Oracle adapter uses only interface tables and the IFACETABLE handler.

For more information about handlers, see the endpoints and handlers information in the *IBM Maximo Asset Management Integration Guide*.

# Oracle integration processing

# 2

The installation of Maximo Enterprise Adapter for Oracle Applications customizes the integration for Oracle E-Business Suite. The adapter also includes additional components such as the outbound and inbound processing flows.

Anyone involved in the implementation or day-to-day administration or use of the Maximo Enterprise Adapter for Oracle Applications should familiarize themselves with Oracle integration processing.

# Adapter components

Maximo Enterprise Adapter for Oracle Applications installs the following Oracle-specific components:

- Enterprise services
- Publish channels
- Interface tables
- PL/SQL exit procedures
- PL/SQL triggers
- Concurrent jobs
- Integration controls
- Cron task

## Oracle enterprise services

The following enterprise services for inbound transactions are provided with Maximo Enterprise Adapter for Oracle Applications:

- Chart of accounts
- Company
- Vendor
- General ledger component
- Inventory balance
- Inventory
- Inventory issue
- Invoice
- Item
- Labor and craft
- Material issue
- Projects
- Purchase order
- Purchase contracts
- Purchase requisition
- Receipt
- Receipt of rotating items

## Oracle publish channels

The following publish channels for outbound transactions are provided with Maximo Enterprise Adapter for Oracle Applications:

- Purchase contract
- Invoice
- Inventory
- Item
- Material reservation
- Project transactions
- Purchase order
- Purchase requisition
- Receipt

# Integration processing

There are two options for configuring the inbound transfer of data from Oracle E-Business Suite to Maximo Asset Management. You can use either PL/SQL triggers or concurrent jobs. You cannot use a combination of PL/SQL triggers and concurrent jobs.

At installation time, you must specify whether you want to use triggers or concurrent jobs to send transactions to Maximo Asset Management. If you choose concurrent jobs, and you later decide to use triggers, you must re-set the environment variables and re-run the installation script for Oracle Applications.

## PL/SQL triggers

PL/SQL triggers run when an outbound transaction updates the interface tables or when an inbound transaction is generated in Oracle E-Business Suite.

## Concurrent jobs

Concurrent jobs are PL/SQL batch programs that you run or schedule to run periodically on Oracle E-Business Suite. You can use concurrent jobs instead of triggers to send transactions from Oracle E-Business Suite to Maximo Asset Management.

In Oracle E-Business Suite, when you generate an inbound transaction and a concurrent request runs, a concurrent job extracts the inserted or updated data from the last time that the concurrent job has run. Then, the concurrent job sends the data to Maximo Asset Management by using the integration framework. You can schedule your concurrent jobs to run as frequently as your business needs require.

You can run one concurrent request or a group of concurrent requests, referred to as a concurrent request set.

**Concurrent jobs configuration**

You can configure concurrent jobs to support partial purchasing or full purchasing in Oracle E-Business Suite. For example, if you create purchase orders in Oracle E-Business Suite but receive the orders in Maximo Asset Management and then invoice in Oracle E-Business Suite, you can configure concurrent jobs for purchase orders and invoices. You do not need to configure concurrent jobs for Receiving.

## Database triggers versus concurrent jobs

Triggers transmit data immediately to Maximo Asset Management. With concurrent jobs, you control when transactions are sent to Maximo Asset Management. Concurrent jobs transmit data from the last execution date and time to the current system date and time.

When you are planning your installation, you must decide whether to implement triggers or concurrent jobs. To transfer your data from Oracle E-Business Suite to Maximo Asset Management in real time, you implement triggers. To transfer your data from Oracle E-Business Suite to Maximo Asset Management in batches at

specified times, you implement concurrent jobs. If you change the data transfer process from one mode to the other, you must confirm that all current transactions have been processed.

When you are deciding between database triggers or concurrent jobs, consider the frequency and volume of transactions between Oracle E-Business Suite and Maximo Asset Management. When you are testing your implementation, ensure that the performance is suitable for the transactions that the integration supports.

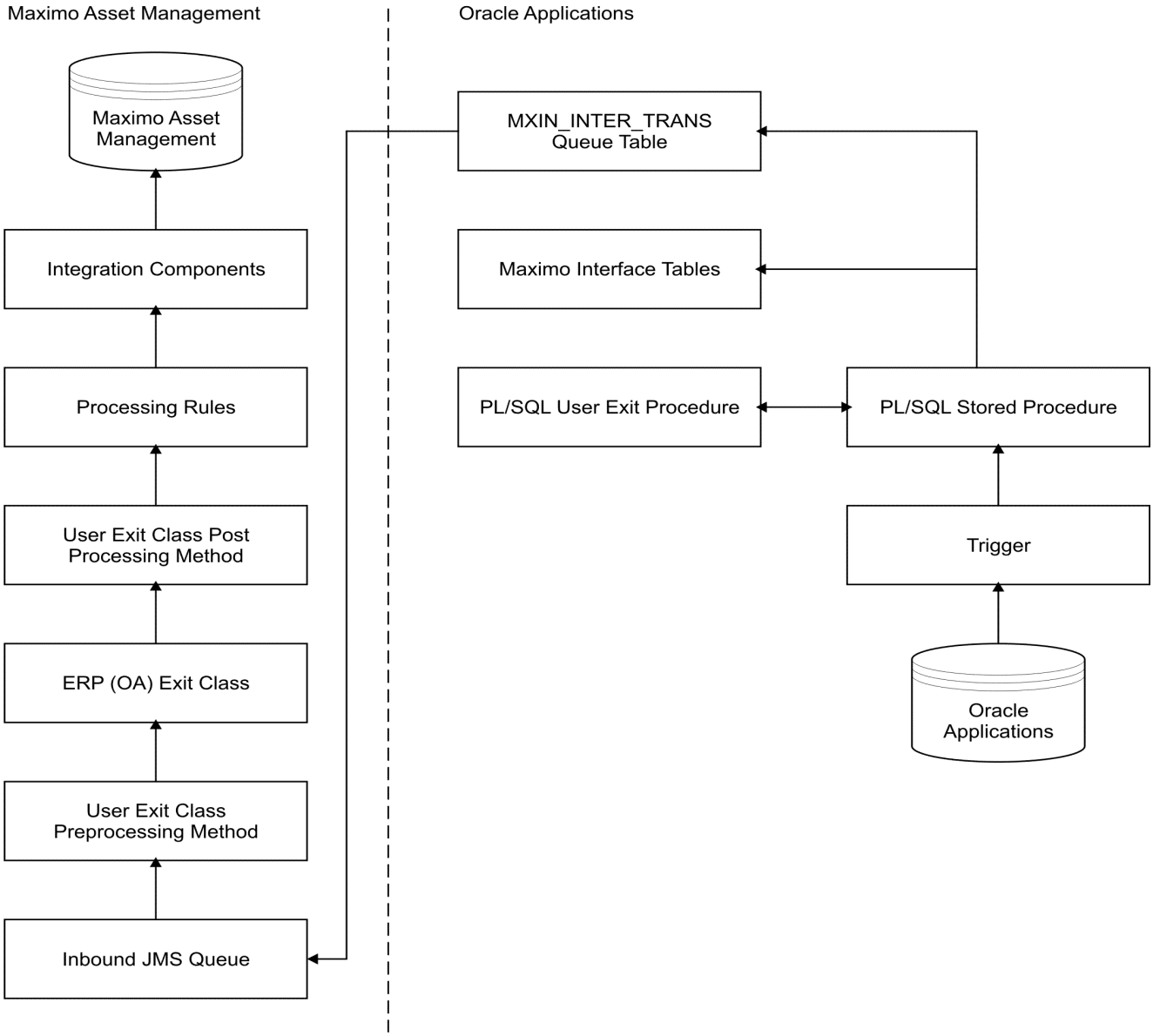
When you are implementing triggers or concurrent jobs, work with your Oracle Applications Database Administrator. For more information, see the Oracle Applications System Administrators guide.

# Inbound integration processing

## Data transfer from Oracle E-Business Suite to Maximo Asset Management by using database triggers

Data can be transfered from Oracle E-Business Suite to Maximo Asset Management by using PL/SQL database triggers. Transfering data from Oracle E-Business Suite to Maximo Asset Management by using PL/SQL database triggers is also known as the trigger architecture.

The diagram shows the inbound flow of data from Oracle E-Business Suite to Maximo Asset Management by using the PL/SQL triggers.



**PL/SQL triggers**

In the inbound direction, a trigger runs when you insert or update data on an Oracle E-Business Suite database table. The PL/SQL stored procedure checks integration controls, status codes, record types, to ensure that the data is to be sent to Maximo Asset Management. Additional processing might take place.

**ERP exit procedures**

ERP exit procedures can be either Java-language exit procedures or PL/SQL stored procedures. The ERP exit procedures are available for customization.

- The Java-language exit procedures validate and manipulate data as it moves between Maximo Asset Management and the interface tables.
- PL/SQL stored procedures and user exits manage data as it moves between the interface tables and Oracle E-Business Suite. The stored procedures contain predefined processing logic and should not be modified in any way.

# Example of inbound processing using triggers

**Transfer of purchase orders**

The transfer sequence of purchase orders is the same for every inbound transaction. The initiating action, integration control names, exit procedure names, and interface table names differ by enterprise service.

1. Create and approve a purchase order in Oracle E-Business Suite.
2. If the integration framework accepts and processes PO transactions, a PL/SQL trigger or concurrent job, and a procedure combination check the value of the POIN Oracle integration control. If the value is 0 (false), the integration framework process stops.
3. The same PL/SQL procedure moves the record from the base Oracle E-Business Suite table to Maximo Asset Management interface table (MXPO_IFACE). The procedure then updates the MXIN_INTER_TRANS queue table with the information about the action that must be performed on the interface table records.
4. A polling program checks the MXIN_INTER_TRANS table for new records. When the polling program finds the new record, the integration framework identifies the inbound JMS queue that is assigned to the enterprise service and external system and writes the message to that queue.
5. The integration framework retrieves the record from the inbound queue.
6. If a multiplication control exists for the enterprise service, the integration framework multiplies the object structure for every site or organization that is specified in the control.
7. If a Java user exit procedure exists, the integration framework runs the preprocessing method. You must use the preprocessing method to customize the external record (ER), which contains the data an Oracle E-Business Suite format.
8. The integration framework calls the Java ERP exit procedure (POInExt), which completes Oracle E-Business Suite processing and converts data from an enterprise service to an object structure format. Both the external record and internal record (IR) exist. The IR contains the transaction data in an integration framework format.

9. If a Java user exit procedure exists, the integration framework runs the postprocessing method. Use the postprocessing method to customize the mapping between the ER and the IR.

10. The integration framework runs the processing class for the enterprise service (MXPO_FROA12).

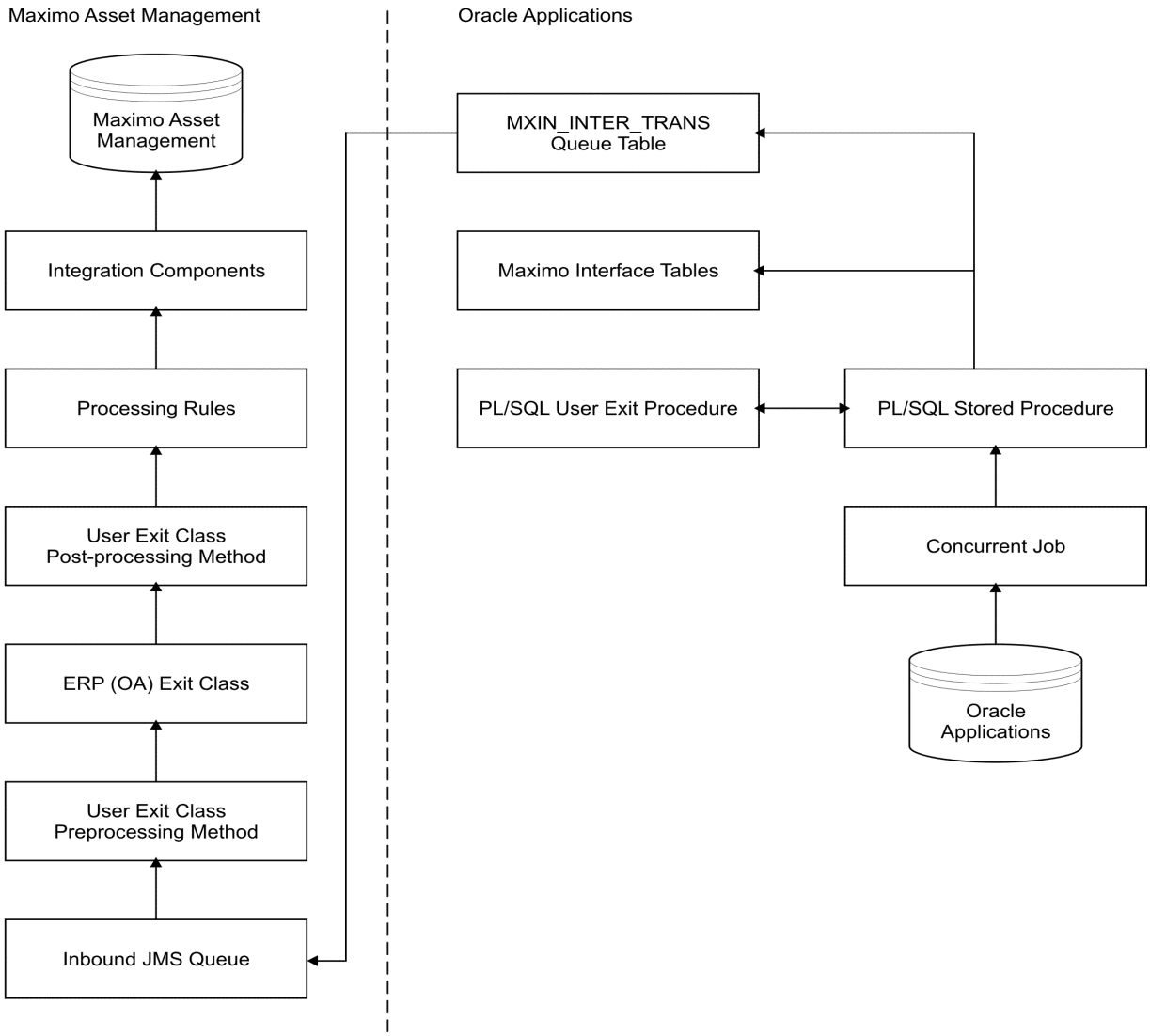
11. If any processing rules exist, the integration framework applies them to the object structure in the order that is specified by the processing rule sequence number.

12. The integration framework builds the objects from the object structure (MXPO) and passes the objects to the integration framework for standard application processing.

## Data transfer from Oracle E-Business Suite to Maximo Asset Management by using concurrent jobs

You can use concurrent job requests to send transactions from Oracle E-Business Suite to Maximo Asset Management. Concurrent jobs are batch programs that run PL/SQL stored procedures on Oracle E-Business Suite. You can run concurrent jobs manually or schedule concurrent jobs to run periodically.

The diagram shows the inbound flow of data from Oracle E-Business Suite to Maximo Asset Management by using concurrent jobs.




In Oracle E-Business Suite, when you generate an inbound transaction and a concurrent request runs, a concurrent job extracts the inserted or updated data from the last time that the concurrent job has run. Then, the data is sent to Maximo Asset Management by using the integration framework.

You can run a concurrent request or a concurrent request set, depending on the interface for the transaction and on your business requirements. When you install the Maximo Enterprise Adapter with concurrent job architecture, concurrent requests and request sets are provided to support standard integration scenarios. You can create additional concurrent requests or request sets to suit your specific integration requirements. The concurrent requests can be executed in the System Administrator application in Oracle E-Business Suite. You can configure Oracle Applications to make concurrent requests available for execution in any module. For a list of concurrent requests, see Concurrent requests provided with the concurrent job architecture in the *Maximo Enterprise Adapter for Oracle Applications Configuration Guide*.

**OAREPROCESS cron task**

As concurrent jobs run to a schedule, there may be a delay before transactions are transferred from Oracle E-Business Suite to Maximo Asset Management. This can

lead to a delay before a main record and its related records are transferred from Oracle E-Business Suite to Maximo Asset Management.

You use the OAREPROCESS cron task to synchronize the timing of concurrent job requests. After you install Maximo Enterprise Adapter for Oracle Applications, you must activate the cron task in Maximo Asset Management. You use the System Configuration > Platform Configuration application to enable the OAREPROCESS cron task.

**Example of OAREPROCESS cron task**

The current job for purchase orders is scheduled to run at midnight and the receiving concurrent job is scheduled to run at 10pm. A purchase order is created in Oracle E-Business Suite at 4pm. The purchase is received in Oracle E-Business Suite at 6pm. At 10pm, the Receiving concurrent job will try to transfer the receipt to Maximo Asset Management. The receiving transaction will fail as the purchase order does not exist in Maximo Asset Management. The OAREPROCESS cron task will then keep trying to send the receipt. When the purchase order concurrent job runs at midnight, the purchase order is created in Maximo Asset Management. When the purchase order exists in Maximo Asset Management, the receipt can be transferred by the crontask into Maximo Asset Management.

To configure the cron task, see the Cron task set-up topic in the Maximo Asset Management Information Center.

# Example of inbound processing that uses concurrent jobs

**Transfer of purchase orders**

The transfer sequence of purchase orders is the same for every inbound transaction. The initiating action, integration control names, exit procedure names, and interface table names differ by enterprise service.

Inbound processing using concurrent jobs is intended to transfer batches of records from Oracle E-Business Suite to Maximo Asset Management. However, in this example, the transfer of one record is considered to illustrate the processing steps.

1. You create and approve a purchase order in Oracle E-Business Suite.
2. If the integration framework accepts and processes PO transactions, the concurrent job runs according to the schedule that you have defined. The procedure that is associated with the concurrent job checks the value of the POIN Oracle integration control. If the value is 0 (false), the integration framework process stops.
3. The same associated stored procedure moves the record from the base Oracle E-Business Suite table to Maximo Asset Management interface table (MXPO_IFACE). The procedure then updates the MXIN_INTER_TRANS queue table with the information about the action that must be performed on the interface table records.
4. A polling program checks the MXIN_INTER_TRANS table for new records. When the polling program finds the new record, the integration framework identifies the inbound JMS queue that is assigned to the enterprise service and external system and writes the message to that queue.
5. The integration framework retrieves the record from the inbound queue.

6. If a multiplication control exists for the enterprise service, the integration framework multiplies the object structure for every site or organization that is specified in the control.

7. If a Java user exit procedure exists, the integration framework runs the preprocessing method. You must use the preprocessing method to customize the external record, which contains the data an Oracle E-Business Suite format.

8. The integration framework calls the Java ERP exit procedure (POInExt), which completes Oracle E-Business Suite processing and converts data from an enterprise service to an object structure format. Both the external record and internal record exist. The internal record contains the transaction data in an integration framework format.

9. If a Java user exit procedure exists, the integration framework runs the postprocessing method. Use the postprocessing method to customize the mapping between the external record and the internal record.

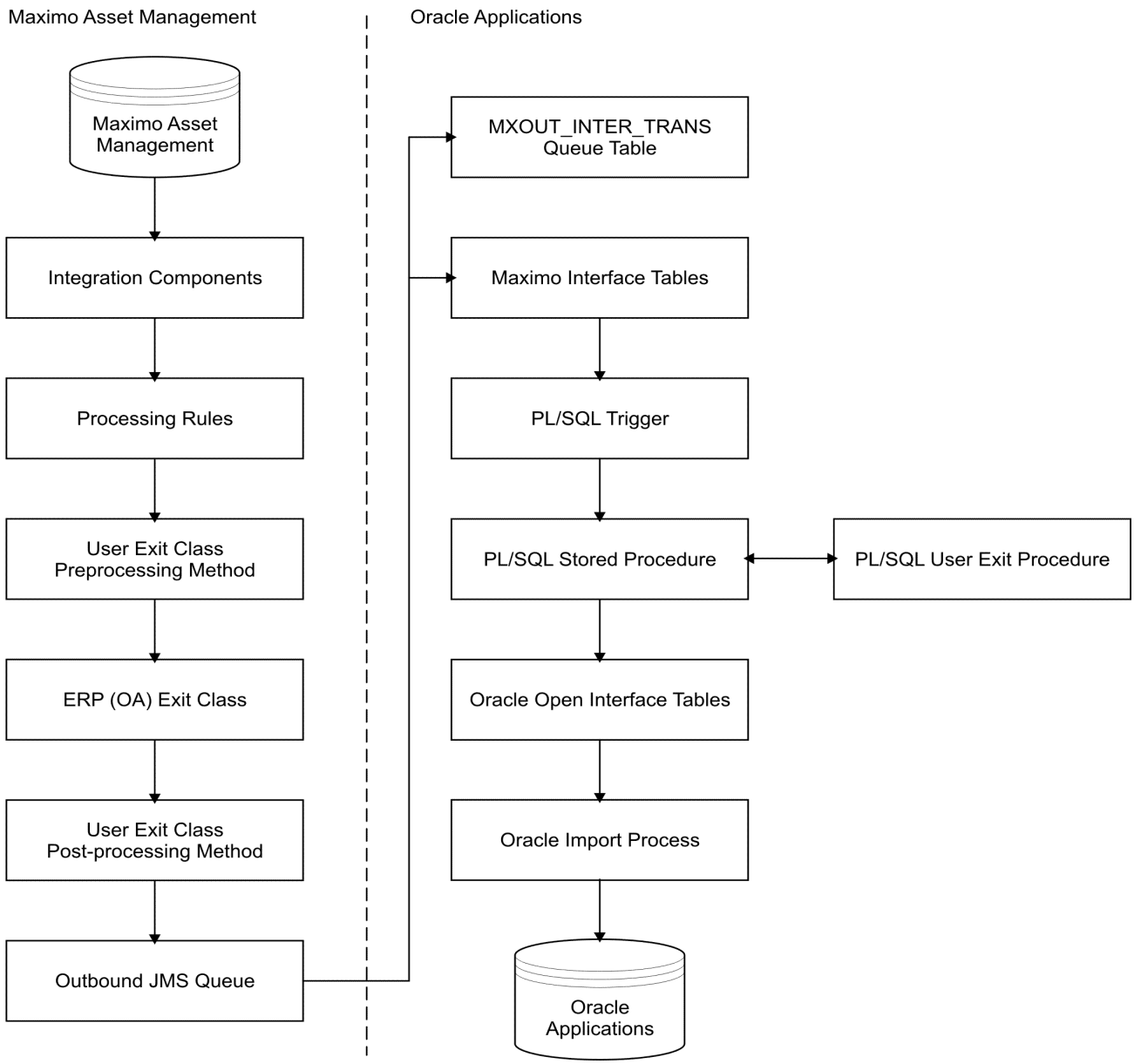
10. The integration framework runs the processing class for the enterprise service (MXPO_FROA12).

11. If any processing rules exist, the integration framework applies them to the object structure in the order that is specified by the processing rule sequence number.

12. The integration framework builds the objects from the object structure (MXPO) and passes the objects to the integration framework for standard application processing.

# Outbound integration processing

The following diagram shows the outbound flow of data from the integration framework to Oracle E-Business Suite.




**Oracle open interface tables**

The Oracle open interface tables are a staging area where outbound integration transactions reside until you initiate an Oracle E-Business Suite import process. These interface tables are standard, and are provided by the Oracle E-Business Suite.

**PL/SQL trigger**

When an outbound transaction is passed to an interface table, a trigger is run on the insert or update to the interface table. The trigger extracts the data from the interface table and passes it to the PL/SQL stored procedure for additional processing.

**ERP exit procedures**

There are two ERP exit procedures, they are Java-language exit procedures and PL/SQL stored procedures. The user exit procedures are available for customization.

- The Java-language exit procedures validate and manipulate data as it moves between Maximo Asset Management and the interface tables.
- PL/SQL stored procedures and user exits manage data as it moves between the interface tables and Oracle E-Business Suite. The stored procedures contain predefined processing logic and should not be modified in any way.

**Oracle Applications import process**

An Oracle E-Business Suite import process retrieves data from an Oracle open interface table and creates or updates records in Oracle E-Business Suite database. There is one import process for each outbound enterprise service.

# Example of outbound processing using triggers

**Transfer of purchase requisitions**

The transfer events of purchase requisitions are the same for every outbound transaction. The initiating action, integration control names, exit procedure names, and interface table names differ by publish channel.

1. Create and approve a purchase requisition.
2. The integration framework creates the object structure (MXPR) from the applicable objects.
3. The integration framework runs the processing class for the publish channel (MXPR_TOOA12).
4. If custom processing rules exist, the integration framework applies them to the object structure in the order specified by the processing rule sequence number.
5. If a Java user exit procedure exists, the integration framework runs the preprocessing method. You must use the preprocessing method to customize the internal record (IR), which contains the data in an integration framework format.
6. The integration framework calls the Java ERP exit procedure (PROutExt), which performs predefined the Oracle E-Business Suite processing, and converts data from an object structure to a publish channel format. At this point, both the internal record and external record (ER) exist. The ER contains the transaction data in an Oracle E-Business Suite format.
7. If a Java user exit procedure exists, the integration framework runs the postprocessing method. Use the postprocessing method to customize the mapping between the IR and the ER.
8. The integration framework writes the transaction to the outbound JMS queue.
9. The cron task that polls the outbound queue passes the message to a message router, which identifies and invokes the interface table handler.
10. The processing class that is associated with the handler writes records to the interface table (MXPR_IFACE) and the MXOUT_INTER_TRANS queue table. The latter specifies the action that Oracle E-Business Suite must perform on the interface table records.

11. A PL/SQL trigger and procedure combination on the MXPR_IFACE interface table moves the record from that table to the Oracle E-Business Suite PO_REQUISITIONS_INTERFACE open interface table.

12. The PR Requisition Import request set, which Oracle E-Business Suite supplies, imports the PR directly into the Oracle E-Business Suite database with the necessary validations.

# Interface tables

# 3

The integration framework uses two sets of interface tables to exchange data between Maximo Asset Management and Oracle E-Business Suite. Interface tables are relational database tables that contain a flat (non-hierarchical) representation of integration framework data.

Interface tables are an option for integration with systems that use database tables to exchange data. Interface tables are generated based on the definition of the object structure that is associated with an enterprise service or publish channel. Invocation channels, object structure services, standard services, and query operations do not support interface tables.

Data synchronization messages can be exchanged by using interface tables. Interface tables do not support the Query and Invoke operations.

A single external system can transfer outbound data by using either XML messages, interface table flat messages, or both. You can assign an endpoint value to an external system at the publish channel level.

# Location of interface tables

The endpoint definition for an external system or a publish channel points to the database where its interface tables are stored. The database can be a local database or a remote database.

The predefined interface table endpoint (MXIFACETABLE) points to the local database, and you can add endpoints for remote databases.

# Names of interface tables

The integration framework registers interface table names to an enterprise service or a publish channel. You must configure the object structure that is associated with the enterprise service or publish channel to support flat files. The Support Flat Structure check box must be selected on the object structure.

Default names for the interface tables are provided by the adapter, but you cannot change or customize these interface table names.

# Interface queue tables

The interface queue tables identify the sequence in which a receiving system should process the records in the respective interface tables. Two queue tables exist, one for inbound transactions and the other for outbound transactions.

| Interface queue table | Processing direction |
|---|---|
| MXOUT_INTER_TRANS | Outbound |
| MXIN_INTER_TRANS | Inbound |

Some transactions depend on the successful processing of a previous transaction. For example, a PO record must be processed before the PO receipt record. The receiving system must process the records in the same sequence in which the sending system created the records.

All inbound and outbound transactions must have a record that is inserted into the corresponding inbound or outbound queue table. This record contains an IFACENAME column which identifies the publish channel or enterprise service. The TRANSID is a unique sequential identifier that the interface table uses to identify the record or records that are associated with the transaction. You can identify the contents of a transaction by looking up all the records with a given TRANSID value in the corresponding interface table.

The sequence of TRANSID identifies the sequence in which records are processed by the integration framework. For example, when the PO and receipt are entered into Maximo Asset Management, the TRANSID values for the PO record must be lower than the TRANSID values for the PO receipt records that reference that PO.

The primary difference between the MXIN_INTER_TRANS and MXOUT_INTER_TRANS queue tables is the direction of the interface table

records that they track. The external system must write to the MXIN_INTER_TRANS queue table, and the integration framework must read from it. The integration framework writes to the MXOUT_INTER_TRANS queue table, and the external system reads from it.

The external system can use the MXOUT_INTER_TRANS table or retrieve outbound records from interface tables. The interface queue tables are generated the first time that you create interface tables for an endpoint. Each endpoint has its own pair of interface queue tables and own a counter for maintaining the outbound TRANSID value.

# Creation of interface tables

When an enterprise service and a publish channel use the same interface table, the Create Interface Tables window displays a list of interface tables based on the uniqueness of the interface table name and its corresponding endpoint.

You can create interface tables for enterprise services and publish channels when the associated object structures are marked as flat supported. The Support Flat Structure check box must be selected on the object structure. The alias conflict must also be resolved before an interface table is created.

You can create interface tables for data synchronization on enterprise services and on publish channels. Interface tables do not support Query and Invoke operations.

You can create interface tables for a specific endpoint. You must identify where the tables are created. The database location that is referenced by the endpoint can be a local database or a remote database.

When you create interface tables on a local database, the columns are registered in the Maximo Asset Management data dictionary. Local interface tables that use a database table and a database column show all updates (except insertions and deletions) to a base column attribute (such as data type) when you run the database configuration operation.

When columns are added to or deleted from the base table, you must regenerate the corresponding enterprise service and the publish channel interface tables to apply the column changes. No changes are applied in the remote databases. You must regenerate remote interface tables to apply the column changes.

# Regeneration of interface tables

When columns are added or deleted from the Maximo Asset Management database tables, you must regenerate all local and remote interface tables that are associated with those object structures.

You can regenerate interface tables by using the Create Interface Tables option on the Select Action menu in the External Systems application. If you select the Rename Existing check box, the application backs up existing data in the corresponding interface table to the INTERFACETABLENAME_BAK table.

If necessary, restore the data to the new table. If you do not back up the table, the table is dropped and the data is lost when you regenerate the table.

You cannot regenerate an interface table when the MXIN_INTER_TRANS queue table contains a record that points to that interface table. When a row exists in that queue table, the corresponding inbound transaction is ready to process, or the inbound transaction is in error.

The interface table creation process does not check for records in the MXOUT_INTER_TRANS queue table.

# Deletion of interface tables and records

When one or more related inbound transaction records are successfully processed in an interface table, the corresponding record is deleted from the MXIN_INTER_TRANS queue table. This means that the transaction was delivered successfully to the inbound Java Message Service (JMS) queue.

Records are deleted from the MXIN_INTER_TRANS queue table, never from the individual interface tables. The system administrator determines when and how to delete records from the interface tables.

For outbound transactions, the external system must manage the deletion and archiving of data in the queue table and interface tables. An administrator must manage the archiving of data in the interface tables.

You cannot delete interface tables by using the user interface or by deleting the corresponding object structure. If necessary, a system administrator can drop the table.

# Format of interface tables

The format of an interface table is the same as the format of the corresponding object structure. The interface table includes the persistent and nonpersistent columns that are included in the object structure. The interface table excludes the columns that are excluded from the object structure.

The interface tables include additional columns that identify the sequence in which the sending system writes, and the receiving system processes the records in the various interface tables.

## Key columns

If the interface table represents a hierarchical object structure with parent-child object relationships, the table does not include any part of the child object key columns that are included in the parent object key columns.

For example, PERSONID is a key column in the PERSON, PHONE, EMAIL, and SMS records. The PERSONID column appears only at the parent (PERSON) level in the MXPERSON_IFACE interface table.

# Duplicate columns and aliases

The XML representation of a hierarchical object structure contains duplicate column names, but interface table and flat file representation do not. If an object structure has duplicate non-key column names in both a parent object and a child object, a duplicate column name error occurs when the interface table or a flat file record is generated.

To resolve the duplicate column name problem, change the system alias for duplicate column names. Every system database column can have an alias alternate name. When an alias exists, Maximo Asset Management uses the alias when interface tables and flat files are generated. You must change the alias to eliminate the duplicate column name error.

## Column name lengths and aliases

Most columns do not have an alias, but some columns have aliases that support the predefined enterprise services or publish channels.

A database column can have only one alias. Ensure that any alias that you assign to a column is valid for every object structure that uses that column. If multiple object structures use the object, a change to an alias affects every interface table that is associated with those object structures.

The columns in the predefined object structures have asset management system-assigned aliases. Check for duplicates when you create a hierarchical object structure or when you add an object to a predefined object structure.

The Add/Modify Alias window shows the fields and aliases for the objects in a selected object structure, and identifies any duplicate alias names with a check in the duplicate column. If a duplicate alias exists, overwrite its value in the ALIASNAME column. If a duplicate does not exist, you cannot change the alias.

# Restricted columns

The HASLD field, which is an internal system column, is excluded from all object structures. Do not include this column in any object structure that is associated with an interface table. The LANGCODE field is also excluded from the predefined object structures.

# Integration processing columns

The following table shows the columns that are used in the interface table sequencing, retrieval, and processing. Some columns are in either the interface queue tables or the interface tables; some are in both places.

| Column name | Interface queue table | Interface tables |
|---|---|---|
| IFACENAME | Yes | No |
| TRANSID | Yes | Yes |
| TRANSSEQ | No | Yes |
| EXTSYSNAME | Yes | No |
| ACTION | Yes | No |
| IMPORTMESSAGE | Yes (used inbound only) | No |
| TRANSLANGUAGE | Yes | No |
| MESSAGEID | Yes (used outbound only) | No |
| IFACETBNAME | Yes (used outbound only) | No |

## IFACENAME column

The IFACENAME column in the interface queue table contains the name of the enterprise service or the publish channel that is used in a transaction. The IFACENAME column is populated in the outbound transactions. For inbound transactions, the external system must populate this column with the enterprise service name that corresponds to the row that is inserted into an interface table.

## TRANSID column

The TRANSID column in an interface queue table is a sequential number that uniquely identifies an integration transaction. The TRANSID and the interface table name, identifies a unique transaction. The interface queue table can contain one record with a TRANSID value. The corresponding interface table can have one or more records with the TRANSID, depending on the number of records that are written to that interface table as part of that enterprise service or the publish channel.

For example, assume that you create a purchase order with one line item. This transaction uses the predefined MXPOInterface, and it increments the TRANSID value that is associated with the MXOUT_INTER_TRANS queue table to 1065. The transaction produces the following records:

- One entry in the MXOUT_INTER_TRANS queue table, with the IFACENAME value equal to the MXPOInterface value, and the TRANSID value equal to 1065.
- One entry in the MXPO_IFACE interface table, with the TRANSID value equal to 1065.

If the same purchase order has three line items, the transaction produces the following records:

- One entry in the MXOUT_INTER_TRANS queue table, with the IFACENAME value equal to the MXPOInterface value, and the TRANSID value equal to 1065.
- Three entries in the MXPO_IFACE interface table, each with the TRANSID value equal to 1065.

  In this case, each entry with the TRANSID value equal to 1065 has a unique secondary sequence number.

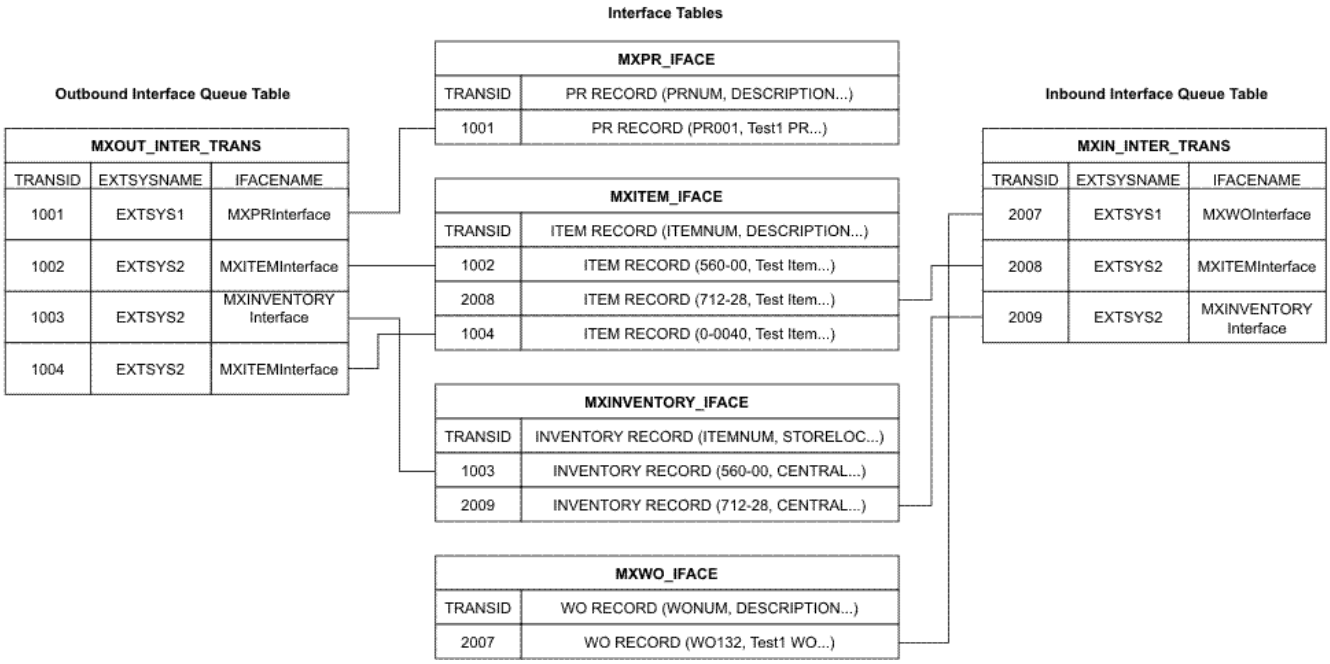
If a transaction writes to multiple interface tables, the interface queue table contains a separate record with a unique TRANSID value for each interface table.

Each interface queue table maintains its own TRANSID counter. The outbound TRANSID value is initialized when the interface queue table records are generated. You must create and maintain the TRANSID counters that populate the inbound queue tables and the interface table records.

If the external systems do not correctly manage the inbound TRANSID counters, sequential processing is not guaranteed. Ensure that the TRANSID values that the external system generates does not duplicate the TRANSID value that is generated. Errors occur if duplicate TRANSID values exist and when you process the same object structure in both an inbound and an outbound direction by using a single interface table.

Each endpoint has its own set of interface queue tables and its own outbound TRANSID counter.

The following diagram shows an example of the relationship between the interface queue tables and the interface table records. The interface tables contain both inbound and outbound transactions.



The data in the MXOUT_INTER_TRANS queue table directs the external system to process the interface table records in the following sequence.

| Sequence | Interface table | Record identifier (TRANSID) |
|---|---|---|
| 1 | MXPR_IFACE | 1001 |
| 2 | MXITEM_IFACE | 1002 |
| 3 | MXINVENTORY_IFACE | 1003 |
| 4 | MXITEM_IFACE | 1004 |

The data in the MXIN_INTER_TRANS queue table directs the integration framework to process the interface table records in the following sequence.

| Sequence | Interface table | Record identifier (TRANSID) |
|---|---|---|
| 1 | MXPO_IFACE | 2007 |
| 2 | MXITEM_IFACE | 2008 |
| 3 | MXINVENTORY_IFACE | 2009 |

## TRANSSEQ column

When multiple records in an interface table share the same TRANSID value, the TRANSSEQ column provides a secondary sequence number that indicates the sequence in which those records should be processed.

For example, if a purchase order has three line items, that transaction might produce the following records:

- One entry in the MXOUT_INTER_TRANS queue table, with the IFACENAME value equal to the MXPOInterface value, and the TRANSID value equal to 1065.
- Three entries in the MXPO_INTERFACE table:
  - One entry (PO line 1) with the TRANSID value equal to 1065 and the TRANSSEQ value equal to 1.
  - One entry (PO line 2) with the TRANSID value equal to 1065 and the TRANSSEQ value equal to 2.
  - One entry (PO line 3) with the TRANSID value equal to 1065 and the TRANSSEQ value equal to 3.

The TRANSSEQ column is only in the interface tables.

## EXTSYSNAME column

An interface table can contain both inbound and outbound transactions. The following table shows that the EXTSYSNAME column in the interface queue tables can contain inbound or outbound data.

| Value of EXTSYSNAME | Processing direction |
|---|---|
| The external system that is the destination of the transaction. | Outbound |
| The value of a valid and enabled external system that is defined in the integration framework. | Inbound |

## ACTION column

The following table shows the ACTION column values in an interface queue table and the corresponding action that the integration framework or external system applies to the interface table.

| ACTION column value | Action |
|---|---|
| Add | Inserts the data that is provided in the message. |
| Delete | Deletes the parent data, and any related child data, from the database. |
| Change | Updates parent and child data contents of the message, but does not delete existing child data that is not explicitly specified in the message. |
| Replace | Replaces the existing records with the contents of the message and deletes existing child data that is not referenced in the message. |
| AddChange | If the primary record does not exist, processes as an add action; otherwise, processes as a change action. |
| Null | If the primary record does not exist, processes as an add action; otherwise, process as a replace action. |

Maximo Asset Management populates the ACTION column in the outbound messages. If the external system does not populate the column in the inbound messages, the integration framework tries to retrieve and replace the corresponding database record. If the record does not exist, the integration framework tries to add it to the database.

## IMPORTMESSAGE column

The IMPORTMESSAGE column holds any error message that was produced when the interface table row was moved to the inbound queue.

## TRANSLANGUAGE column

The TRANSLANGUAGE column identifies the language of the transaction. For an outbound transaction, this value indicates the language of the user who initiated the transaction. For an inbound transaction, this value indicates the language of the transaction. Any attributes that support a multilingual

environment are expected to be in the language that the TRANSLANGUAGE value defines.

### MESSAGEID column

The MESSAGEID column is a unique identifier that Maximo Asset Management assigns to every outbound transaction.

### IFACETBNAME column

The IFACETBNAME column is the name of the interface table that corresponds to the IFACENAME column. This column applies to outbound transactions only.

## Long description columns in Oracle databases

Long description columns are stored in a CLOB (character large object) column in an Oracle database. However, interface tables contain two versions of each CLOB column, one with data type CLOB, and one with data type ALN with a character length of 4000. In the following example, the name of the CLOB column is the column alias. The name of the alphanumeric column is the column alias with the suffix *2*.

| Data type | Description column name |
|---|---|
| CLOB | PO_DESCRIPTION_LD |
| ALN | PO_DESCRIPTION_LD2 |

Maximo Asset Management populates both columns in outbound transactions. For inbound transactions, the integration framework uses the value in the ALN column if it is not null; otherwise, the value in the CLOB column is used.

# Interface table polling

A predefined cron task, IFACETABLECONSUMER, polls the MXIN_INTER_TRANS queue table and uses the IFACENAME, EXTSYSNAME, and TRANSID values in the queue table to place the corresponding interface table records into the appropriate inbound JMS queue. Then, individual records in the JMS queue are processed.

The interface table polling process checks that the record external system and enterprise service names are valid and currently enabled. If they are not, the record is marked in error and remains in the interface table.

If you disable interface table polling, new records remain in the interface tables. The messages that were sent to the inbound JMS queue are processed.

You must set up a mechanism to retrieve outbound transactions from the interface tables. You can use a polling program, as Maximo Asset Management does for inbound transactions, triggers, or any other mechanism.

The cron task has the following configurable parameters. All parameters are optional.

| Parameter | Description |
|---|---|
| EXITCLASS | Java exit class that enables the manipulation of data before it is written to an inbound queue. |
| ENDPPOINT | Endpoint that is associated with the interface table. The default value is the predefined endpoint value that points to the local database. |
| ENTERPRISESERVICE | Enterprise service to be polled. The default (null value) is all enterprise services. If you specify a value for this parameter, you also must specify a value for the EXTSYSNAME parameter. The values limit the polling thread to a specific enterprise service instead of the default behavior, which polls for all enterprise services. |
| TARGETENABLED | Optional Boolean flag that controls whether the cron task runs on a specific application server. The default value is 0 (false). |
| EXTSYSNAME | External system to be polled. |
| QUEUETABLE | Enterprise service queue table. The default value is MXIN_INTER_TRANS. |

# Improve polling performance

You can perform advanced configuration of the interface table polling process to improve its performance reading data from interface tables.

Taking steps to improve interface table polling performances is necessary only if you send inbound transactions through the continuous JMS queue and do not require transactions to be maintained in first-in-first-out sequence.

Unless otherwise indicated, the configuration activities in this section are performed in the Cron Task Setup application in the System Configuration module.

## Cron tasks

The interface table polling process uses a single default cron task called IFACETABLECONSUMER. This cron task reads all transactions from all interface tables for all external systems that write to the tables.

For improved single-server and multi-server performance, you can configure multi-threaded interface table polling by defining multiple instances of this task with different property values. Multi-threaded polling is particularly useful when running in a clustered configuration, as different threads can run on different servers, thereby balancing the load.

For example, to designate an instance of the cron task to run on a specific application server, perform the following actions. Assume that the name of the cron task instances is instance1:

In the Cron Task Setup application, set the TARGETENABLED property to 1.

In the Maximo Asset Management application `server1` setup, set the –DIFACETBCONSUMER.*instance1*=1.

When implementing multiple cron tasks, you also must implement mutually exclusive selectors to avoid processing a transaction more than once.

For more information about cron tasks, refer to the *IBM Maximo Asset Management System Administrator Guide* and the online help for the Cron Task Setup application.

## Selectors

You use a selector to add a WHERE clause to a cron task. If you define multiple instances of the cron task, you must define selectors, so that each instance reads mutually exclusive interface table rows. For example, if Maximo Asset Management exchanges data with two external systems, the first thread might poll one system and the second thread might poll the second system.

You define selectors by assigning values to the EXTSYSNAME (external system) and ENTERPRISESERVICE (enterprise service) parameters in the Cron Task Setup application. You can add an IN clause to a selector by entering a pipe-delimited set of values.

To direct a cron task to select only purchase order records for system EXTSYS1 from the queue tables:

- Set ENTERPRISESERVICE=MXPOInterface
- Set EXTSYSNAME=EXTSYS1

To list multiple interface names in the ENTERPRISESERVICE property:

- Set ENTERPRISESERVICE=MXPOInterface|MXPRInterface|…

Requirements for configuring multiple instances of the cron task:

- The selectors must be mutually exclusive, so messages are not processed multiple times.
- The selectors must retrieve all the enterprise service transactions that you use, so that no messages remain unprocessed.

# Configuring external systems

To use interface tables, you must create the tables and configure the IFACETABLECONSUMER cron task.

### Overall processing

To configure the external systems to perform general interface queue table and interface table processing:

1. Define procedures to restore the backups of the interface tables.
2. Create interface tables in the user interface.
3. Manage the archiving of interface tables.

**Inbound processing**

To configure the external system to perform inbound interface queue table and inbound interface table processing:

1. Create and initialize the outbound TRANSID counter.
2. Create records for each interface table that an inbound transaction writes to:
    a. Create an interface table record and populate it with the following information:
        - The transaction data
        - The incremented TRANSID value
        - If multiple records exist for the same interface table, the incremented TRANSSEQ value
    b. Create an MXIN_INTER_TRANS queue record with the following information:
        - The same TRANSID value that is contained the interface table record
        - The name of the enterprise service that corresponds to the interface table, in the IFACENAME column
        - Optional: The ACTION value
        - The identifier of the external system, in the EXTSYSNAME column
    c. Perform a single commit, to commit all records for a transaction at one time.

**Inbound null columns**

If a column in an interface table contains a null value, the applicable objects process the column in the following ways:

- By default, the field is not updated by the transaction.
- If you add the empty tag when you are using a user exit, the object adds a null value to the field in the Maximo Asset Management database.

**Outbound processing**

To configure the external system to perform outbound interface queue table and inbound interface table processing:

1. Set up a process to retrieve interface table transactions by using the MXOUT_INTER_TRANS queue table. You can use a polling program, a trigger, or any other mechanism.
2. For the polling program to process transactions sequentially:

**a** Read the records in the MXOUT_INTER_TRANS queue table in the TRANSID sequence.

**b** Enable each record in the MXOUT_INTER_TRANS queue table:

- Access the interface table that you just identified, and retrieve the first record in which the TRANSID value matches the TRANSID value in the current MXOUT_INTER_TRANS queue record. If the interface table contains multiple records with the same TRANSID value, retrieve and process them in TRANSSEQ sequence.
- Process data according to the value in the ACTION column of the interface queue table.

**c** Commit all records for a single database transaction.

**d** Delete the current record from the MXOUT_INTER_TRANS queue table.

**3.** Implement error management, based on your external system requirements.

# Organizational structure

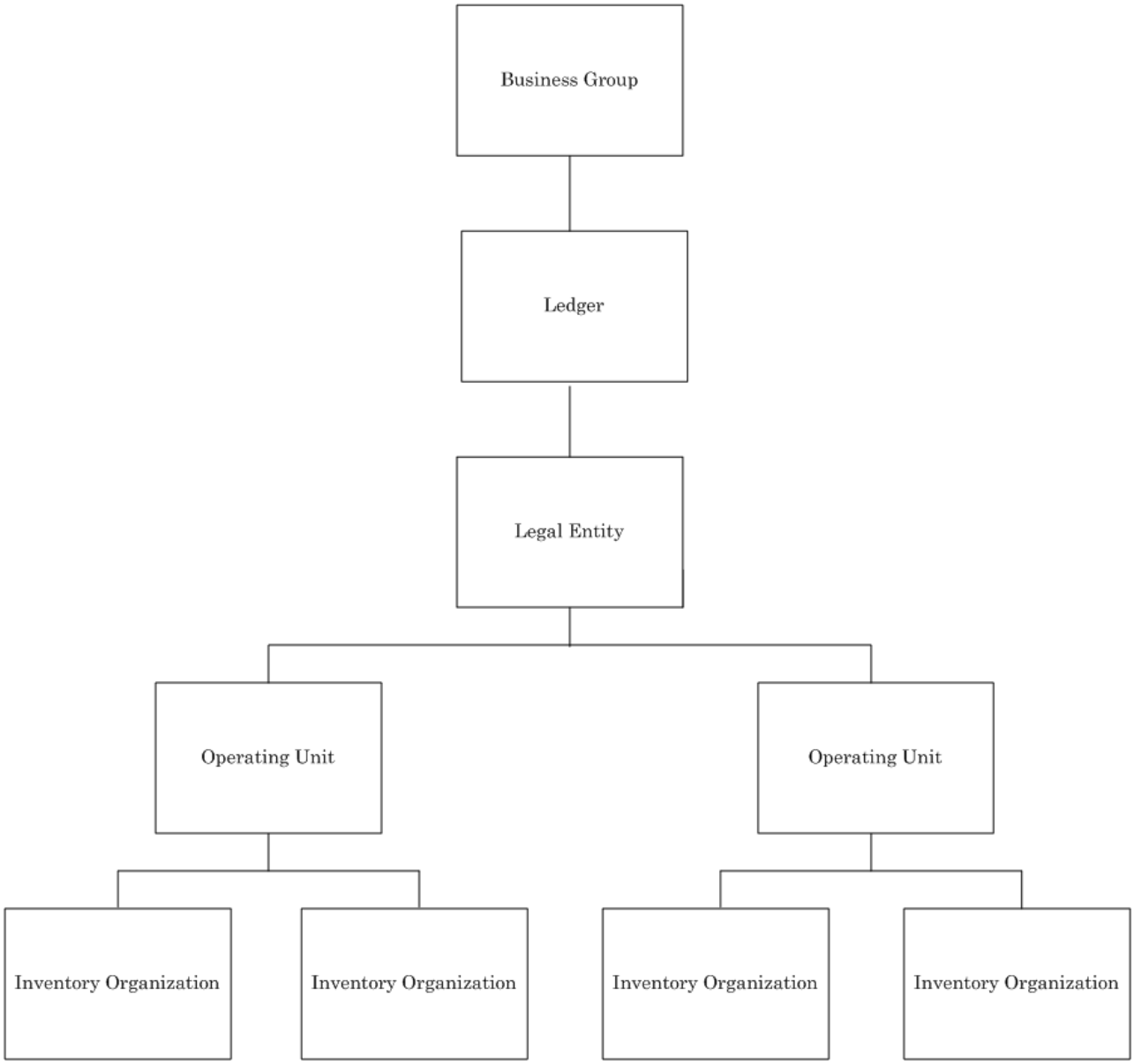
# 4

Before you can configure the integration between Maximo Asset Management and the Oracle E-Business Suite, you must understand the relationships between organization levels in Maximo and Oracle E-Business Suite.

You can define multiple organizations and the relationships among them in a single installation in Oracle E-Business Suite.

The following table and diagram show the organization levels and their relationships with one another in Oracle.

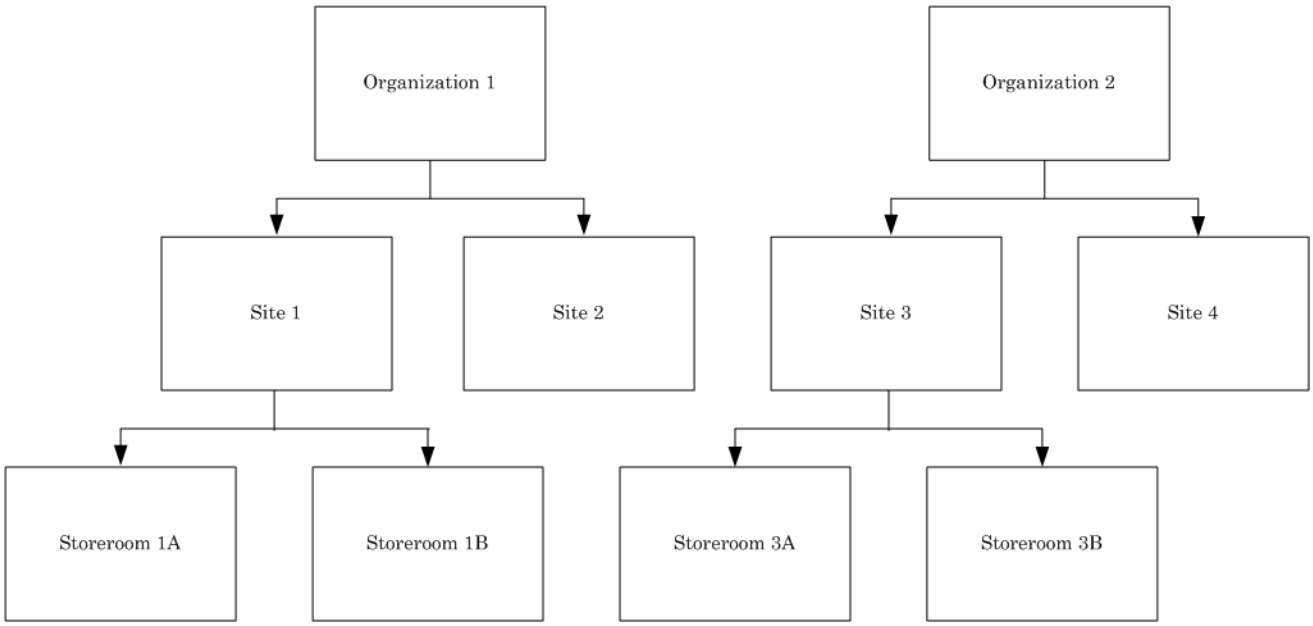
| Organization level | Description |
|---|---|
| Business group | A consolidated enterprise, a major division, or an operation company.<br><br>Multiple sets of books can share the same business group when they share the same business group attributes. |
| Ledger | A financial reporting entity.<br><br>All transactions against a particular ledger use the same accounting calendar, chart of accounts, and functional currency. |
| Legal entity | A legal company with its own tax identifier and fiscal and tax reports |
| Operating unit | An organization which can be a sales office, division, or department.<br><br>All accounts payable, accounts receivable, order entry, purchasing, receipt, and shipping transactions are generated. |
| Inventory organization | An organization which is a distribution center, manufacturing plant, sales office, or warehouse.<br><br>All distribution, inventory transactions, or manufacturing are tracked. |


Business Group
Ledger
Legal Entity
Operating Unit
Operating Unit
Inventory Organization
Inventory Organization
Inventory Organization
Inventory Organization

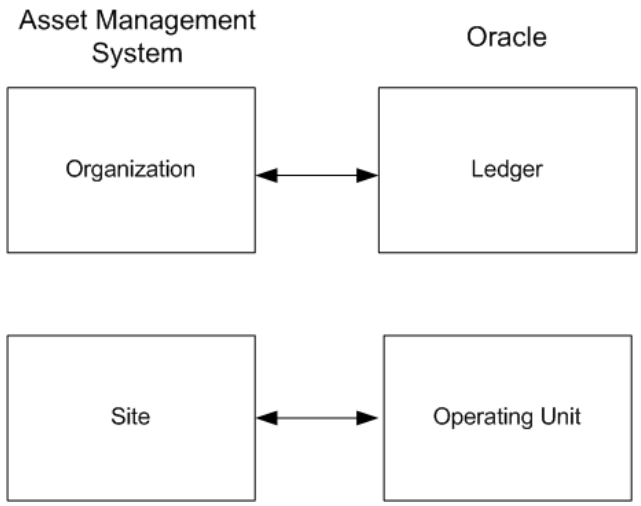
An organization is a legal entity to which one or more sites belong. You define the chart of accounts and base currency at the organization level, and each site uses the values from the organization to which it belongs. You also define items, labor, purchase agreements, and vendors at the organization level.

You can define data such as equipment, inventory, purchase orders, and work orders at the site level. Each site can then have its own set of equipment and inventory that other sites cannot share or view.

The following diagram illustrates the organizational and site levels, and their relationships to data in the system.




An organization corresponds to a ledger in the Oracle E-Business Suite, a site to an Oracle E-Business Suite operating unit, and a storeroom to an Oracle E-Business Suite inventory organization.



# Integration controls

# 5

When you use integration controls you can configure the behavior of the integration framework processing according to the requirements of individual organizations and sites.

Integration controls contain the values that the integration framework uses to perform the following functions:

- Translate the Maximo Asset Management values to the Oracle values
- Translate the Oracle values to the Maximo Asset Management values
- Provide the default values for null columns
- Validate the data values

# Integration controls

The integration controls are defined within Maximo Enterprise Adapter for Oracle Applications. They are used by Java and PL/SQL processing exits. If you create custom processing rules, the rules can reference the integration controls. Most integration controls can be configured at the site, organization, or system level.

An integration control can be configured to override values at any of the following levels:

| Control value | Description |
|---|---|
| System-level | Applies to all organizations and sites. If the control is not configured for organization-level or site-level values, the system default is used. If the control is configured for organization-level or site-level values but no value exists for a particular organization or site, the system-level value is used. |
| Organization-level | Applies to all sites within an organization. If a control is configured for organization-level values but no value exists for a particular organization, the system-level value is used. |
| Site-level | A site-level value applies to a specific site within an organization. If a control is configured for site-level values but no value exists for a particular site, the organization-level value is used. If no value exists, the system-level value is used. |

## Control types

You can work with four types of integration controls to meet your Oracle processing needs.

### Boolean control

A boolean type control specifies a value of 0 (false) or 1 (true).

### List control

A list type control contains a list of values. You can enter multiple values for the control and optionally assign a domain value to the control. Assigning a domain value ensures the validation of values entered for that control, at any level. If a domain is not assigned, the values entered are not validated.

### Value control

A value type control contains a single value. You can enter a single value for the control and assign a domain value to the control.

### Cross-reference control

A cross-reference control replaces one value with another. In a publish channel, an application value is converted to an external system value and, in an enterprise service, an external system value is converted to an application value. You can assign a domain value to a cross-reference control. If a domain is specified, any value specified for the control is validated against that domain. If a domain is not assigned, there is no validation of the values entered.

A processing error can result when a one-to-many mapping exists between the system values and external values. A one-to-many mapping is valid when the cross-reference control is defined as a multiplication control.

**Sites**

Sites correspond to external system business units, but the two systems use different values for these entities. A cross-reference control can perform the translation between the two values. For example, a cross-reference control in an enterprise service can translate the business unit value EX001 to the site value MX001. In an outbound direction, the same control can translate the site value MX001 to the business unit value EX001.

**Multiplication control**

A multiplication control is a cross-reference control that copies an inbound message for multiple organizations or sites. It has one Oracle E-Business Suite value and multiple system values. A multiplication control can update the company in every organization in the database.

## Masking data

You can use three integration controls to transfer user, item, and storeroom values: GENUSR, GENITEM, and GENSTORE. The transfer can take place between systems without having to define the values in both Maximo Asset Management and Oracle E-Business Suite.

### Masking users

When you use the GENUSR integration control, you can bypass the task of defining Maximo Asset Management users in Oracle E-Business Suite before sending outbound transactions to Oracle E-Business Suite.

You can use this control to set up a dummy user. The integration framework uses this value, rather than the one on the transaction, when transferring data to Oracle E-Business Suite. If the value of GENUSR is null, user masking is not in effect. You set the GENUSR control at the organization level and you can define one dummy user per organization.

### Masking items

When you use the GENITEM integration control, you can transfer item data from Maximo Asset Management to Oracle E-Business Suite without having to define the items in Oracle E-Business Suite.

You can use this control to set up one dummy item in Oracle E-Business Suite. The integration framework uses this value, rather than the value in the transaction, when transferring item data (for example, a purchase requisition line) to Oracle E-Business Suite. If the value of the GENITEM control is null, item masking is not in effect. You can configure the GENITEM control at the site level and you can assign one dummy item per site.

### Masking storerooms

When you use the GENSTORE integration control, you can transfer purchasing and inventory data without having to synchronize storerooms between Maximo Asset Management and Oracle E-Business Suite.

You can use this control to define a dummy storeroom. The integration framework uses this value, rather than the value in the transaction, when transferring storeroom data to Oracle E-Business Suite. If the value of GENSTORE is null, storeroom masking is not in effect.

You can use this control when you want to manage inventory exclusively, but need to send requisitions and purchase orders to Oracle E-Business Suite. You can configure the GENSTORE control at the site level and you can assign one dummy storeroom per site.

## Integration control creation and configuration

Unless you create a service or channel or modify the processing of an existing service or channel, you do not need to create integration controls. Modifying control values at the external system level is generally sufficient to customize predefined service and channel processing.

You can create an integration control in the Enterprise Services application and Publish Channels application. The online help in both applications guide you through the process of creating a control.

When you create an integration control, keep the following points in mind:

- Ensure that the control name is unique across all adapters.
- You must add the control to existing external systems that use it. You can add a control in the External Systems application.

# Integration control details

The tables on the following pages show the integration controls that are installed by the OA12 adapter. The columns within the table are:

| Column name | Description |
|---|---|
| Control name | Name of the integration control |
| Description | A brief description of the control |
| Type | Control type:<br><br>B = Boolean<br>L = List<br>V = Value<br>X = Cross-reference |
| Domain | If applicable, the name of the Maximo Asset Management domain that contains the valid value(s) for the control.<br><br>N/A = Domains are not applicable to the control<br>N = No domain |
| Organization and site | Indicates if the system-level control value can be overridden at the organization or site level<br><br>N = Neither<br>O = Organization<br>S = Site |
| Value | The predefined value(s) of the control<br><br>(D) = default value |

| Control name | Description | Type | Domain | Org or site | Value |
|---|---|---|---|---|---|
| APLINETYPE | Default invoice line type in Oracle E-Business Suite | V | N | S | ITEM (D) |
| APSEND | Status(es) at which invoices are sent from Maximo Asset Management to Oracle E-Business Suite | L | IVSTATUS | S | APPR |
| CATEGORYXREF | Cross-reference an item category and the Oracle E-Business Suite stock-enabled flag | X | CATEGORY | S | STK = Y<br>NS = N |
| COAXREF | Cross-reference an organization ID and the Oracle E-Business Suite chart of accounts ID | X | N | N | Null (D) |
| CONVHRS | Hours to convert annual salary to hourly rate | V | N | N | 2080 |
| CRAFTXREF | Multiply the Oracle E-Business Suite craft code to organizations | X | N | N | Null (D) |
| DTC_EXP | Destination type code for direct issue PO lines | V | N | S | EXPENSE (D) |
| DTC_INV | Destination type code for replenishment PO lines | V | N | S | INVENTORY (D) |
| GENITEM | Dummy identifier for masking item number | V | N | S | Null (D) |
| GENSTORE | Dummy identifier for masking storeroom on outbound transaction | V | N | S | Null (D) |
| GENUSR | Dummy identifier for masking user ID in outbound transactions | V | N | O | Null (D) |
| GLCURNCY | Default GL currency code in the Oracle E-Business Suite | V | N | O | USD (D) |
| GLCURNCYTYPE | Type of currency exchange rates | V | N | O | User (D) or Corporate |
| GLSOURCE | Source business object of a general ledger journal | L | GLSOURCE | S | Null |
| INVCSTATUSXREF | Translate internal and external invoice status, if using synonyms | X | N | S | APPR=APPR<br>PAID=PAID |
| INVCTYPEXREF | Cross-reference Maximo Asset management and Oracle E-Business Suite invoice document types | X | INVTYPE | S | INVOICE = STANDARD<br>CREDIT = CREDIT<br>DEBIT = DEBIT |

| Control name | Description | Type | Domain | Org or site | Value |
|---|---|---|---|---|---|
| ITMSUBXREF | Cross-reference Oracle E-Business Suite inventory organization and default subinventory code | X | N | S | Null (D) |
| JECATXREF | Cross-reference a transaction type and Oracle E-Business Suite journal category<br><br>Valid transaction types:<br><br>AVGCSTADJ, CAPCSTADJ, CREATEASSET, CURBALADJ, CURVAR, INSERTITEM, INVCEVAR, KITCOSTVAR, MATRECTRANS-INVOICE, MATRECTRANS-KITBREAK, MATRECTRANS-KITMAKE, MATRECTRANS-MISCLRECEIPT, MATRECTRANS-POCOST, MATRECTRANS-RECEIPT, MATRECTRANS-RETURN, MATRECTRANS-TRANSFER, MATUSETRANS-INVOICE, MATUSETRANS-ISSUE, MATUSETRANS-KITBREAK, MATUSETRANS-KITMAKE, MATUSETRANS-MISCLRECEIPT, MATUSETRANS-POCOST, MATUSETRANS-RETURN, MATUSETRANS-TRANSFER, NON-WORK, OT-REF, PCOUNTADJ, RECBALADJ, SERVRECTRANS, SICK, STDCSTADJ, STDRECADJ, TAX1, TAX2, TAX3, TAX4, TAX5, TOOLTRANS, TOTAL, TRAV, VAC, WMATL, WORK | X | N | O | Null (D) |
| JEPROJSEND | The project transaction types to be sent to Oracle E-Business Suite | L | GLSOURCE | O | LABTRANS<br>MATRECTRANS<br>MATUSETRANS<br>SERVRECTRANS<br>TOOLTRANS |
| LABXREF | Multiply an Oracle E-Business Suite labor code to organizations | X | N | N | Null (D) |
| LANGXREF | Cross-reference a language code and the Oracle E-Business Suite process set ID | X | N | N | Null (D) |

| Control name | Description | Type | Domain | Org or site | Value |
|---|---|---|---|---|---|
| LINETYPEXREF | Cross-reference a line type (synonyms) and the Oracle E-Business Suite line type | X | LINETYPE | S | EXTERNAL = Goods<br>ITEM = Goods<br>MATERIAL = Goods<br>SERVICE = Services<br>SPORDER = Goods<br>STDSERVICE = Services<br>TOOL = Goods |
| LOTTYPEXREF | Cross-reference the Maximo Asset Management and the Oracle E-Business Suite lot control indicators | X | LOTTYPE | O | NOLOT = 1<br>LOT = 2 |
| OACATXREF | Cross-reference the line type and the Oracle E-Business Suite item or service category identifier.<br><br>**Note**: Cross-reference a value ITEM to an external value Null so the category defaults based on item number and destination organization | X | LINETYPE | S | Null (D) |
| OAITMLOTPREFIX | Lot prefix for lot controlled item | V | N | S | Null (D) |
| OAITMLOTSTARTNUM | Lot start number for lot controlled item | V | N | S | Null (D) |
| OAPCDEFORDERUNIT | Order unit default on purchase contract line when no unit of measure is specified on a purchase contract line in Oracle | S | N | S | Null (D) |
| ORGXREF | Cross-reference an organization and the Oracle E-Business Suite ledger ID | X | N | N | Null (D) |
| PCSEND | Statuses at which contracts are sent from Maximo Asset Management to Oracle E-Business Suite | L | CONTRACTSTATUS | O | APPR |
| PCSTATUSXREF | Cross-reference the internal and external contract status, if using synonyms | X | CONTRACTSTATUS | O | APPR = APPR<br>CAN = CAN<br>CLOSE = CLOSE |
| PCTYPEXREF | Cross-reference Maximo Asset Management and Oracle E-Business Suite contract types | X | CONTRACTTYPE | O | BLANKET = BLANKET<br>BLANKET = CONTRACT BLANKET<br>= PLANNED |

| Control name | Description | Type | Domain | Org or site | Value |
|---|---|---|---|---|---|
| POSEND | Statuses at which purchase orders are sent from Maximo Asset Management to Oracle E-Business Suite | L | POSTATUS | S | Any PO status<br>APPR (D) |
| POSTATUSXREF | Cross-reference the internal and external PO status, if using synonyms | X | POSTATUS | S | APPR = APPR |
| POTYPEXREF | Cross-reference Maximo Asset Management and the Oracle E-Business Suite purchase order types | X | POTYPE | S | STD = STANDARD<br>REL = RELEASE |
| PRSEND | Statuses at which purchase requisitions are sent from Maximo Asset Management to Oracle E-Business Suite | L | PRSTATUS | S | Any PR status<br>APPR (D) |
| PRSTATUSXREF | Cross-reference the internal and external PR status, if using synonyms | X | PRSTATUS | S | APPR = APPR<br>CAN = CAN<br>CLOSE = COMP<br>WAPPR = WAPPR |
| SITEXREF | Cross-reference the site ID and the Oracle E-Business Suite operating unit | X | N | N | Null (D) |
| SUBLOCDEL | Delimiter between subinventory code and locator ID in the bin number | V | N | S | - (D) hyphen |
| USEPRNUM | Specifies when integration framework uses the PR number as the Oracle E-Business Suite PR number (value 1) or uses the auto generated number used by the Oracle import routine (0) | B | N/A | S | 1 or 0 (D) |

# Oracle integration controls

The MAXORACTLS table contains additional controls that Oracle E-Business Suite uses to determine the data to be sent to Maximo Asset Management. These controls are created during the installation process and are used by PL/SQL exit procedures. The controls fall into the following two categories.

| Control category | Description |
|---|---|
| Inbound | These controls specify whether any records of a particular transaction type are to be sent to Maximo Asset Management. |
| Processing | These controls manage such information as log records, error records, triggers, concurrent jobs in Oracle E-Business Suite. |

## Oracle inbound controls

The Oracle inbound controls are defined on an enterprise service. They specify whether Maximo Asset Management accepts any transactions for a particular enterprise service, for example, invoices. If Maximo does not accept the transaction, the adapter does not send any transactions for that enterprise service to the interface tables, thereby decreasing network traffic and improving overall performance.

| Control value | Action |
|---|---|
| 0 | The adapter does not send any transactions for the enterprise service from Oracle E-Business Suite to the interface table. |
| 1 | The adapter sends transactions for the enterprise service from Oracle E-Business Suite to the interface table. The integration framework determines which transactions to process on the basis of integration control settings and processing logic. |

If the value of the Oracle POIN control is 1, the adapter sends all purchase order transactions from the Oracle E-Business Suite interface table to the Maximo Asset Management interface table.

If the value of the Oracle POIN control is 0, the adapter sends no purchase order transactions from Oracle E-Business Suite to the Maximo interface table.

The following table lists the Oracle inbound controls.

| Control name | Default value | Description |
|---|---|---|
| APIN | 1 | Controls transfer of inbound invoices |
| COAIN | 1 | Controls transfer of inbound charts of accounts |
| COMIN | 1 | Controls transfer of inbound companies (vendors) |
| ISUIN | 1 | Controls transfer of inbound material issues |
| INVBALIN | 1 | Controls transfer of inbound item balances |
| ITEMIN | 1 | Controls transfer of inbound items |
| LCIN | 1 | Controls transfer of inbound labor codes |
| PCIN | 1 | Controls transfer of inbound purchase contracts |
| POIN | 1 | Controls transfer of inbound purchase orders |
| RCVIN | 1 | Controls transfer of inbound receipts |

The following table lists the Oracle inbound controls for concurrent jobs.

| Control name | Control value format | Description |
|---|---|---|
| CR_POAPPR_DATE | fmDD-MON-RRRR HH24:MI:SS | Records the last date concurrent requests ran for PO approvals. |
| CR_POLINE_DATE | fmDD-MON-RRRR HH24:MI:SS | Records the last date concurrent requests ran for PO lines. |
| CR_PRIMP_DATE | fmDD-MON-RRRR HH24:MI:SS | Records the last date concurrent requests ran for PO requisitions. |
| CR_RECEIPT_DATE | fmDD-MON-RRRR HH24:MI:SS | Records the last date concurrent requests ran for receipts. |
| CR_RCVSERIAL_DATE | fmDD-MON-RRRR HH24:MI:SS | Records the last date concurrent requests ran for receiving rotating items. |

| Control name | Control value format | Description |
|---|---|---|
| CR_INVC_DATE | fmDD-MON-RRRR HH24:MI:SS | Records the last date concurrent requests ran for invoices. |
| CR_INVCAPPR_DATE | fmDD-MON-RRRR HH24:MI:SS | Records the last date concurrent requests ran for invoice approvals. |
| CR_POREL_DATE | fmDD-MON-RRRR HH24:MI:SS | Records the last date concurrent requests ran for PO releases. |
| CR_POIMP_DATE | fmDD-MON-RRRR HH24:MI:SS | Records the last date rejected concurrent requests ran for PO interfaces. |
| CR_POACTION_DATE | fmDD-MON-RRRR HH24:MI:SS | Records the last date concurrent requests ran for PO action histories. |
| CR_GLCC_DATE | fmDD-MON-RRRR HH24:MI:SS | Records the last date concurrent requests ran for GL code combinations. |
| CR_GLCOMP_DATE | fmDD-MON-RRRR HH24:MI:SS | Records the last date concurrent requests ran for GL components. |
| CR_GLCOMPDESC_DATE | fmDD-MON-RRRR HH24:MI:SS | Records the last date concurrent requests ran for GL component descriptions. |
| CR_MTLITM_DATE | fmDD-MON-RRRR HH24:MI:SS | Records the last date concurrent requests ran for inventory items. |
| CR_MTLITMTL_DATE | fmDD-MON-RRRR HH24:MI:SS | Records the last date concurrent requests ran for inventory item descriptions. |

| Control name | Control value format | Description |
|---|---|---|
| CR_VNDHDR_DATE | fmDD-MON-RRRR HH24:MI:SS | Records the last date concurrent requests ran for suppliers. |
| CR_VNDSITE_DATE | fmDD-MON-RRRR HH24:MI:SS | Records the last date concurrent requests ran for supplier sites. |
| CR_VNDCONT_DATE | fmDD-MON-RRRR HH24:MI:SS | Records the last date concurrent requests ran for supplier contacts. |
| CR_VNDCONTUPD_DATE | fmDD-MON-RRRR HH24:MI:SS | Records the last date concurrent requests ran for supplier contact processes. |
| CR_PAEMP_DATE | fmDD-MON-RRRR HH24:MI:SS | Records the last date concurrent requests ran for PA employees master data. |
| CR_PACOMP_DATE | fmDD-MON-RRRR HH24:MI:SS | Records the last date concurrent requests ran for master data related to PA employees compensate rates. |

# Oracle processing controls

The Oracle processing controls manage such information as log records, error records, triggers, and concurrent jobs in Oracle E-Business Suite. The following table shows the Oracle processing controls.

| Control name | Default value | Description |
|---|---|---|
| APITRANS | 1 | If value is 0, does not transfer data from Oracle E-Business Suite to the Maximo Asset Management interface tables<br><br>If value is 1, transfers data from Oracle E-Business Suite to the Maximo interface tables |

| Control name | Default value | Description |
|---|---|---|
| APITRG | 0 | If value is 0, enables the triggers on the Oracle E-Business Suite base tables<br><br>If value is 1, disables the triggers and enables the concurrent jobs on the Oracle E-Business Suite base tables |
| EXCEPTIONTOEXT | 0 | If value is 0, inserts and updates records in the Oracle E-Business Suite interface tables and logs errors<br><br>If value is 1, sends errors back to the Oracle E-Business Suite, instead of stopping execution and logging errors |
| EXCEPTIONTOMAX | 0 | If value is 0, inserts the records in the Maximo Asset Management interface tables and logs the processing errors in the interface table status columns<br><br>If value is 1, sends processing errors back to the application server |
| EXTSYS | OA12 | External system name, as defined in Maximo Asset Management |
| FCEXTRACTDATE | Null | Last FC Extract Date (fmDD-MON-RRRR HH24:MI:SS) |
| GLSOBID2 | 0 | Reporting Ledger ID used to retrieve EXCHANGERATE2 in Oracle E-Business Suite |
| LOGLEVEL | ERROR | Sends transaction information to the MAXORALOG tables for troubleshooting purposes.<br><br>If value is ERROR, provides information about conditions that caused processing to stop.<br><br>If value is WARNING, provides information about conditions that did not cause processing to stop.<br><br>If value is INFO, provides informational messages, such as the entry and exiting of procedures.<br><br>If value is DEBUG, provides detailed debugging information. |
| MEAORAVER | 7.6.0.0 | The installed version of Maximo Enterprise Adapter for Oracle Applications |

| Control name | Default value | Description |
|---|---|---|
| PROJPAY | 0 | The source of the Oracle pay rate<br><br>If value is 0, pay rate is from the Oracle HR table<br><br>If value is 1, pay rate is from employee cost rate in Oracle Projects |

# Oracle integration control configuration

You set the value of the Oracle integration controls with a SQL statement.

The following table shows the structure of the MAXORACTLS table where the controls reside.

| Field | Description | Length | Value | Data type |
|---|---|---|---|---|
| CTLNAME | Control name | 25 | Not null | Varchar2 |
| CTLVALUE | Control value | 50 | Null | Varchar2 |
| CTLDESC | Control description | 200 | Null | Varchar2 |

To update an Oracle integration control, use the following format:

```
update MAXORACTLS
set    CTLVALUE = 'control value'
where  CTLNAME  = 'control name';
commit;
```

The following example sets the LOGLEVEL control to a DEBUG value.

```
update MAXORACTLS
set    CTLVALUE = 'DEBUG'
where  CTLNAME  = 'LOGLEVEL';
commit;
```

You must wait a minute for changes to the MAXORACTLS table to take effect.

# Customization with user exits

# 6

Every enterprise service and publish channel has corresponding Java and PL/SQL user exit procedures that you can use to override or to customize the predefined integration framework processing.

For inbound transactions, the integration framework calls the PL/SQL user exits before writing data to the interface tables. The integration framework also calls the Java user exits before writing data to the Maximo database.

For outbound transactions, the integration framework calls the Java user exits before writing data to the interface tables. The integration framework also calls the PL/SQL user exits after retrieving data from the interface tables and before writing it to the Oracle open interface tables.

You also can customize the integration framework processing by using integration controls and processing rules.

# Java user exit procedures

The integration framework applies Java user exit procedures immediately before and after it runs the predefined processing for the corresponding enterprise service or publish channel. You can use preprocessing and postprocessing methods within a user exit to indicate when the code is to be applied.

Although you can give any name to a user exit procedure, consider using the following naming conventions.

| Processing direction | File naming convention |
|---|---|
| Inbound | Max`Integrationpoint`User.class |
| Outbound | Mout`Integrationpoint`User.class |

The Oracle adapter does not install any predefined user exit procedures.

If you use the application EAR file, you must build and deploy it after updating Java user exit procedures. For more information about building the application EAR file, see the *IBM Maximo Asset Management System Administrator Guide*.

**ATTENTION** The *IBM Maximo Asset Management Integration Guide* states that you can update Java exit procedures. This statement is not true of the Maximo Enterprise Adapter for Oracle Applications. These exit procedures contain predefined code that formats data to the specific requirements of Maximo Asset Management and Oracle E-Business Suite.

The Java ERP exit procedures reside in the following directory:

`rootdirectory`\applications\maximo\businessobjects\classes\psdi\iface\oa11i

# PL/SQL user exit procedures

The Oracle adapter installs skeleton PL/SQL user exit procedures for every enterprise service and publish channel. The PL/SQL user exit procedures reside in the following directory:

`rootdirectory`\ORACLEAPIS\apisuser

# Outbound PL/SQL user exit procedures

In outbound processing, the PL/SQL user exit provides the opportunity to map additional data or modify default mapping to the Oracle open interface table, or to skip the record. The outbound PL/SQL user exit is called just before data is inserted in the Oracle open interface table. The user exit processing typically has three parameters:

- The Maximo Asset Management interface table record
- The Oracle open interface table records
- The skip_transaction flag

If the skip_transaction flag is set to true, the integration framework immediately exits the process and does not insert any more records. However, in the case of item records, a skip_transaction can skip the insert and update (or delete and insert) the current record, not immediately exit the process, and go on to insert and update other records.

# Inbound PL/SQL user exit procedures

In inbound processing, the PL/SQL user exit provides the opportunity to map additional data or modify default mapping to the Maximo Asset Management interface table, or to skip the record. The inbound PL/SQL exit is called just before data is inserted in the Maximo Asset Management interface table. The user exit processing typically has three parameters:

- The Oracle base table records
- The Maximo Asset Management interface table record
- The skip flags

If the skip_transaction flag is set to true, the integration framework immediately exits the process and does not insert any more records. If the user exit procedure has a skip_line flag that is set to true, the integration framework skips the insert of the current line record and continues processing without skipping other lines.

**ATTENTION** Do not modify any PL/SQL files besides user exit procedures. You can identify the types of PL/SQL files by their file extension.

| PL/SQL file | File extension |
|---|---|
| Installation scripts | .sql |
| PL/SQL packages | .pkg |
| PL/SQL package definition | .psp |
| Stored procedures | .sp |
| User exit procedures | .usp |
| Triggers | .trg |

# Multiple language support

# 7

The Maximo Asset Management database can contain some elements, such as descriptions and long descriptions, in multiple languages. Publish channels and enterprise services can include these translated columns.

A single system database can contain data in multiple languages. You can specify the base language when you install the application. If your system uses a language that differs from the base language, you can enable the integration framework to send and receive non-base language data to and from external systems.

# Default processing for translatable columns

When you log in, you can choose a language code other than the Maximo Asset Management base language. In any application, you can then enter language-specific values for columns that are designated as translatable.

By default, outbound transactions contain the applicable column values in the language that is associated with the login session. The language values display whether the transaction is initiated by an application or the Data Export feature.

For example, if the base language of your system is English, you can log in as a French user and update an item record with a French description. The outbound message contains the item description in French, even when the description also exists in English, or a third language.

# Language tables

If a database table contains translatable columns, the database contains a corresponding table called L_tablename, for example, ITEM and L_ITEM. The L_tablename table stores the non-base language values for every translated column except the long description. Long descriptions in all languages are in the LONGDESCRIPTION table.

To include translated values in the output XML, include the L_tablename and LONGDESCRIPTION objects in the applicable object structures.

Provide the base language values as a service input to object structures that have the L_tablename as part of their object definition. Your service input must be in the core object, and all other languages must be in the additional language enabled object.

For example, when English is the base language, the ITEM table contains the English description of an item and the L_ITEM table contains the French and German descriptions of that item. The LONGDESCRIPTION table contains the English, French, and German long descriptions.

When you add the L_tablename object to an object structure, assign the same value to the transLanguage and baseLanguage attributes. Otherwise, the base language values are not available and are processed for the multiple language-enabled fields.

# The translanguage attribute

The attribute identifies the language in which the values for applicable multilingual fields are specified.

If this attribute is missing or does not contain a value, all data is assumed to be in the base language. If the transLanguage value cannot be interpreted, or if the value does not identify a valid language, an error is returned to the service requester.

# Chart of accounts

The Oracle E-Business Suite FND_FLEX_VALUES table holds general ledger account segments and, if applicable, the FND_FLEX_VALUES_TL table holds the description of the segments in languages other than the Oracle E-Business Suite base language. The inbound integration framework generates one transaction record for each combination of segment and language; for example, if Oracle E-Business Suite holds one base language and two additional installed languages for a segment, the integration framework generates three records per segment. The Java exit procedure then evaluates the records and continues processing records that meet one of the following conditions:

- The record is in the Maximo Asset Management base language
- The description field is language-enabled

If a record does not meet either condition, the integration framework skips the record. If necessary, you can write a user exit class or create processing rules to skip additional records.

The general ledger account description is a concatenation of the account segment descriptions. The multilingual processing also applies to the inbound chart of accounts transactions.

# Vendors

In Maximo Asset Management, the vendor description is configured for multilingual support. If the Oracle E-Business Suite is configured for multilingual support, the vendor address can be translated into multiple languages. The integration framework passes the following Oracle fields that are associated with multilingual vendor addresses to the PL/SQL user exit procedure:

- AP_SUPPLIER_SITES_ALL.LANGUAGE
- AP_SUPPLIER_SITES_ALL.VENDOR_SITE_CODE_ALT
- AP_SUPPLIER_SITES_ALL.ADDRESS_LINES_ALT

The integration framework does not pass these values to Maximo Asset Management. If necessary, you can customize the corresponding PL/SQL user exit to pass these fields to the MXVENDOR_IFACE table.

# Items

You can enter item descriptions in multiple languages, so the Oracle open interface table might receive item records in several languages. You must perform the item import process separately for each language.

You must first cross-reference language codes and language numbers in the LANGXREF integration control.

# Importing the item

For each import, you must perform the following actions:

- Log in to Oracle E-Business Suite in the language to be imported
- Enter the language number in the Process Set parameter

The inbound integration framework generates one transaction record for each item and language. For example, if the Oracle E-Business Suite holds one base language and two additional installed languages for an item description, the integration framework generates three records per item.

The Java exit procedure evaluates the records and continues processing records that meet one of the following conditions:

- The record is in the Maximo Asset Management base language
- The description field is language-enabled

If a record does not meet either condition, the integration framework skips the record. If necessary, you can write a user exit class or create processing rules to skip additional records.

# System maintenance and error management

# 8

You must perform periodic system maintenance to improve server efficiency. Some of the tasks that you can perform are to purge interface tables and to update Maximo Asset Management configurations.

Queue error management is initiated when an error condition is identified. The error identification occurs when a message is processing in an outbound or inbound queue. You can view, correct, cancel, and reprocess the erroneous messages.

# Purge the interface tables

The adapter inserts all data records into the Maximo Asset Management interface tables, to provide an audit trail of activity. These records remain in the interface tables until you purge them.

Whether you have to purge a particular interface table depends on the enterprise service or publish channel that you use, but consider periodically purging the tables in the following list. You can use any SQL tool to query or update these tables.

- MAXORALOG
- MXCOA_IFACE
- MXCRAFT_IFACE
- MXGLCOMP_IFACE
- MXGLTXN_IFACE
- MXINVBAL_IFACE
- MXINVENTORY_IFACE
- MXINVISSUE_IFACE
- MXINVOICE_IFACE
- MXINVRES_IFACE
- MXITEM_IFACE
- MXLABOR_IFACE
- MXOUT_INTER_TRANS
- MXPC_IFACE
- MXPO_IFACE
- MXPR_IFACE
- MXRCVROTITM_IFACE
- MXRECEIPT_IFACE
- MXVENDOR_IFACE

# Updating the Maximo Asset Management database configuration

If you update the Maximo Asset Management database configuration after installing the integration, you must recreate the Oracle adapter interface tables to ensure that the interface tables have the same configuration.

To update the Maximo Asset Management database configuration:

1. Complete all transactions in the applicable Maximo Asset Management interface table.
2. Re-create the Maximo Asset Management interface tables in the Oracle E-Business Suite database.
3. Run the install.sql.

To ensure that Maximo Asset Management backs up the data in the interface tables, select the Rename Existing check box in the Create Interface Tables dialog box. You access this dialog box in the Select Action menu in the External Systems application.

**ATTENTION** Before re-creating a Maximo Asset Management interface table, archive the data in the table. When the integration framework re-creates an interface table, it drops the existing one.

# Error management

There are general questions to consider when an integration framework processing problem occurs. You can debug processing errors by using the interface table status fields and the Maximo Asset Management interface log.

You also can manage and view the integration framework transaction messages that have been flagged with an error in the Message Reprocessing application. You can view the error Extensible Markup Language (XML) file without needing to gain access the integration framework server error files.

For additional information about the Message Reprocessing application, see the error management information in the *IBM Maximo Asset Management Integration Guide*.

## Troubleshooting procedures

If data from the source application does not appear in the destination application, take the following steps to resolve the problem:

1. If you reconfigured the Maximo Asset Management database, ensure that you re-created the interface tables and then reinstalled the Oracle E-Business Suite objects.
2. Check the error status code of the Maximo Asset Management interface tables, and the interface log for errors.
3. Correct all errors.
4. If the problem occurs with inbound data, check that the value of the corresponding Oracle inbound control value is equal to 1.
5. Check that all interface table columns are correctly mapped between the Maximo Asset Management and Oracle E-Business Suite. The column type and length must be compatible.
6. Query the database to check that the user objects (triggers and procedures) are still valid.

# Interface table status

Every outbound interface table contains the following columns, which contain information about the status of the transaction:

| Column | Description | |
|---|---|---|
| OA_IFACESTATUS | The status of the transaction. | |
| | DONE | The transaction was successful. |
| | WARNING | The data was transferred but a problem was encountered. Check OA_IFACEMESSAGE for error messages. |
| | ERROR | Due to a system or data error, the data transfer did not occur. Check OA_IFACEMESSAGE for error messages. |
| OA_IFACETIMESTAMP | The date and time the transaction completed or failed. | |
| OA_IFACEMESSAGE | A general message | |

# Interface log

The Maximo Asset Management interface log file, MAXORALOG, is used to record logging and debugging information. The Oracle processing control LOGLEVEL controls the level of detail recorded in MAXORALOG.

| LOGLEVEL value | Logged data |
|---|---|
| ERROR (default) | Information about conditions that caused processing to stop |
| WARNING | Information about conditions that did not cause processing to stop |
| INFO | Informational messages, such as the entry and exiting of procedures |
| DEBUG | Detailed debugging information, including data values |

Consider running periodic reports against the Maximo Asset Management interface log table to check for errors. Check the OA_IFACESTATUS column to identify transactions that did not complete successfully. You can use any SQL tool to run these reports.

To record ERROR, WARNING, or INFO level information, change the value of the LOGLEVEL control.

To record DEBUG level information, you can change the value of the LOGLEVEL control to DEBUG, or you can run a debugging script that sets the LOGLEVEL value to DEBUG.

## Running the debug script

To run the debugging script, complete the following steps:

1. Connect to SQL*Plus® as the MAXORA user.
2. Run the following command:

```
exec MAXAPISET.FLOWON
```

3. Reprocess the transaction in error.
4. Use a SQL tool to run the following command:

```
start C:\MAXIMO\ORACLEAPIS\INSTALL\SETOFVAR
```

5. To run the script showflow.sql in the INSTALL directory and produce an output file of data from the MAXORALOG table, run the following command:

```
start &INSTALLDIR.SHOWFLOW
```

6. View the output file, showflow.out, that the showflow script writes to the log directory.

   This output file contains detailed information about the error transaction.

## Disabling the debug flow

After debugging, disable the debugging flow. To disable the debug flow:

1. Connect to SQL * Plus as the MAXORA user.
2. Run the following command:

```
exec MAXAPISET.FLOWOFF
```

This action resets the LOGLEVEL setting to ERROR.

# Reprocessing interface table errors

When an error occurs in an outbound transaction, the adapter stops processing the transaction and updates the interface table status columns with information about the error. There is no user interface for correcting errors in the Maximo Asset Management interface tables, so you must update the record directly with a SQL tool.

The triggers, which reside on the outbound interface tables, are of trigger type before insert or update. Therefore, directly updating the interface table transactions ensures that the transaction is processed.

Triggers do not exist for the inbound enterprise services. These processes are asynchronous and the integration framework retrieves the records.

The following steps describe how to update an outbound transaction with an error in the interface table, then send the transaction to the Oracle open interface tables again.

To reprocess the transaction, complete the following steps:

1. Back up the interface table that contains the transaction in error.
2. Identify the record and note the values in the TRANSID and TRANSSEQ fields.
3. Use a SQL update statement to correct the incorrect data in the record in the interface table.
4. Use the following SQL update statement to set the OA_IFACESTATUS, OA_IFACETIMESTAMP, and OA_IFACEMESSAGE columns in the record to Null and pass the record from Maximo Asset Management to the Oracle open interface table:

```
update INTERFACETABLE
set    OA_IFACESTATUS    = NULL,
       OA_IFACETIMESTAMP = NULL,
       OA_IFACEMESSAGE   = NULL
where  TRANSID            = 'transid value' and
       TRANSSEQ           = 'transseq value';
commit;
```

The trigger does not reprocess a transaction until you set OA_IFACESTATUS to null.

The particular error you encounter may require additional database updates.

# Financial integration

# 9

Maximo Asset Management and Oracle E-Business Suite can exchange the following types of financial data:

| Financial data type | Processing direction |
|---|---|
| General ledger component | Inbound |
| Chart of accounts | Inbound |
| General ledger journal | Outbound |

There are factors that you must be aware of when you integrate Maximo Asset Management and Oracle E-Business Suite financial activities. Some factors include general ledger and account effectiveness, transaction transfer, integration control use, and multilingual capabilities.

# General Ledger component integration

The general ledger (GL) component transfers new and updated general ledger segments from the Oracle E-Business Suite to Maximo Asset Management. The chart of accounts integration is a two-step process: you first transfer individual general ledger segments to Maximo Asset Management, then you transfer the general ledger accounts.

In Oracle E-Business Suite, general ledger segments are defined at the system level. General ledger components are defined at the organization level, so the integration framework copies the inbound general ledger components to every organization that is defined in the COAXREF integration control.

## Effectivity of General Ledger segments

The values in the Enabled, From, To and Posted fields in the Oracle E-Business Suite Segment Values window indicate whether an Oracle general ledger segment is active or inactive. The Enabled, From, and To fields are on the Values, Effective tab and the Posted field is on the Values, Hierarchy, Qualifiers tab.

If the Posted and Enabled check boxes are selected and the system dates fall between the From and To dates, the segment is active.

The SYNCGLCOMP synchronization script sends active and inactive segments to Maximo Asset Management. If a segment is active, the Active check box in the Maximo Asset Management GL Component Maintenance dialog box is selected.

The integration framework uses the following logic to evaluate the From and To dates in the Oracle E-Business Suite and set the value of the Active check box in Maximo Asset Management:

| **Oracle Application E-Business Suite** | | **Maximo Asset Management** |
|---|---|---|
| **From date** | **To date** | **Active value** |
| Null | Null | Checked |
| Null | > or = system date | Checked |
| Not applicable | < system date | Not checked |
| > system date | Not applicable | Not checked |
| < or = system date | Null | Checked |
| From date <= system date <= To date | | Checked |

The general ledger component integration does not update the Maximo Asset Management components when the status of an Oracle segment changes to active or inactive based on the system date. To ensure that Maximo Asset Management uses only active components, run the SYNCGLCOMP synchronization script frequently.

To keep your data synchronized, maintain your General ledger segments in Oracle E-Business Suite and not in Maximo Asset Management.

For more information about the synchronization script, see the *IBM Maximo Enterprise Adapter for Oracle Applications Configuration Guide.*

## General Ledger transfer

The integration framework transfers general ledger segments from the Oracle E-Business Suite to Maximo Asset Management when value of the Oracle integration control COAIN equals 1.

When loading general ledger components to Maximo Asset Management, add a filter to PL/SQL user exit procedure ofglcomp.usp to select only those components that belong to the charts of accounts that are used in Maximo Asset Management. Errors occur when Maximo Asset Management receives duplicate components that belong to the Oracle charts of accounts that Maximo Asset Management does not need.

For example, to send only the components for charts of accounts 101 and 50195 to Maximo Asset Management, add the following logic to ofglcomp.usp:

```
IF MXGLCOMPREC.ORGID NOT IN (101, 50195) THEN
   SKIP_TRANSACTION := True;
End If;
```

## Integration controls

The general ledger component uses the following Maximo Asset Management and Oracle integration controls.

| Control | Description | System |
|---|---|---|
| COAIN | Send the Oracle E-Business Suite chart of accounts data to Maximo Asset Management | Oracle E-Business Suite |
| COAXREF | Cross-reference the Maximo Asset Management organization ID and the Oracle E-Business Suite chart of accounts ID | Maximo Asset Management |

## Multilingual support

The integration supports multilingual descriptions of general ledger components. For more information, see the multiple language support information chapter.

# Chart of Accounts integration

The chart of accounts integration transfers new and updated general ledger accounts from the Oracle E-Business Suite to Maximo Asset Management.

## General Ledger account configuration

Maximo Asset Management can have multiple general ledger (GL) account structures. The system-level GL account structure must be configured to the maximum length used by organizations in Oracle E-Business Suite.

You must configure the GL accounts in Maximo Asset Management to correspond to the account structures in Oracle E-Business Suite. For more information about configuring GL components in Maximo Asset Management, see the *Configuring GL accounts* topic in the Maximo Asset Management Information Center.

You can write a user exit procedure to transform the format of the Oracle E-Business Suite account numbers to conform to the Maximo Asset Management format.

Maximo Asset Management and the integration framework do not support dependent account components.

**Example**

The following example shows how to do this when an Oracle E-Business Suite segment contains a null value. The Oracle E-Business Suite allows null segments in any part of an account number, but Maximo Asset Management does not allow a null segment to precede a populated segment.

For example, the accounts in the Oracle E-Business Suite have three segments:

- Segment1 = null
- Segment2 = size(2)
- Segment3 = size(3)

For example, the user exit procedure must shift those segments so the Maximo Asset Management account consists of only two segments:

- Segment1 = size(2)
- Segment2 = size(3)

**Transaction transfer to the Oracle E-Business Suite**

When using the account number in an outbound transaction, you must convert the two-segment account in Maximo Asset Management to a three-segment account by shifting the two existing segments to the right and creating the first segment of the account in the Oracle E-Business Suite as a null.

To convert the two-segment account to a three-segment account, you must add the following statements, in the sequence shown, to the MXE_USR_GLOUTX_SP procedure in the ofgloutx.usp file which is located in the `apisuser` directory.

```
OFGLSegs(3) := OFGLSegs(2);
OFGLSegs(2) := OFGLSegs(1);
OFGLSegs(1) := Null;
nSegs := 3;
```

The final statement (nSegs:= 3) changes the number of segments from two to three.

**Transaction transfer to Maximo Asset Management**

When using the account number in an inbound transaction, you must convert the three-segment Oracle E-Business Suite account to two segments by shifting the two non-null segments to the left, and making the third segment of the Maximo Asset Management account a null.

To shift the non-null segments to the left and make third segment null, you must add the following statements, in the sequence shown, to the MXE_USR_GLINX_SP procedure in the ofglinx.usp file which is located in the `apisuser` directory.

```
OFGLSegs(1) := OFGLSegs(2);
OFGLSegs(2) := OFGLSegs(3);
OFGLSegs(3) := Null;
nSegs := 2;
```

The final statement changes the number of segments from three to two.

# Effectivity of accounts

When the adapter transfers an account from Oracle E-Business Suite to Maximo Asset Management, the adapter calculates if the account is active using the dates set in Oracle E-Business Suite.

In the Oracle E-Business Suite General Ledger Accounts window, the values in the Enabled, From Effective Date, To Effective Date, and Allow Posting fields indicate whether a general ledger account is active or inactive. The account is active if the Enabled and Allow Posting check boxes are selected and the system date is between the From Effective date and the To Effective date.

In Maximo Asset Management, you can set a time frame for which an account is active by entering a start date and optionally, an end date. If the end date is not set then the account will never expire. There is a flag set in Maximo Asset Management to indicate if the general ledger account is active.

The SYNCCOA synchronization script sends active and inactive accounts from Oracle E-Business Suite to Maximo Asset Management. If an Oracle account is active, its effective dates are set in the Maximo Asset Management Chart of Accounts application. The Active check box in Maximo Asset Management is set based on the values of the From Effective date and the To Effective date.

The integration framework uses the following logic to calculate the active dates of the account.

If the Enabled and Allow Posting check boxes are selected, the active dates of the account are set in Maximo Asset Management as follows:

| Oracle E-Business Suite | Maximo Asset Management |
|---|---|
| From Effective Date | Active Date |
| To Effective Date | Expiration Date |

If the Enabled and Allow Posting check boxes are not selected the active dates of the account are set in Maximo Asset Management as follows: .

| Oracle E-Business Suite | Maximo Asset Management |
|---|---|
| From Effective Date | Active Date |
| If To Effective Date not populated | Active Date set to current date |
| If From Effective Date >than the current date | Expiration Date is set to From Effective date |
| If From Effective Date <than the current date | Expiration Date is set to current date |

Maximo Asset Management has a column called OLDEXPIREDATE that stores the previous value of the expiration date. The Oracle E-Business Suite To effective date value is mapped to this column.

The chart of accounts integration does not update the Maximo Asset Management accounts when the status of an Oracle account changes to active or inactive based on the system date. To ensure that Maximo Asset Management uses only active components, run the SYNCCOA synchronization script frequently.

The setting of the Maximo Asset Management system property, mxe.int.updatecoafromglcomp, controls if GL Accounts are reactivated when GL Components are reactivated. When set to 1 (true), the accounts are reactivated, When set to 0, the accounts are not reactivated.

For more information about the synchronization script, see the *IBM Maximo Enterprise Adapter for Oracle Applications Configuration Guide*.

## General Ledger transfer

The integration framework transfers general ledger accounts from the Oracle E-Business Suite to Maximo Asset Management when the value of the Oracle integration control COAIN equals 1.

# Integration controls

The chart of accounts uses the following Maximo Asset Management and Oracle integration controls.

| Control | Description | System |
|---|---|---|
| COAIN | Send the Oracle E-Business Suite chart of accounts data to Maximo Asset Management | Oracle E-Business Suite |
| COAXREF | Cross-reference the Maximo Asset Management organization ID and the Oracle E-Business Suite chart of accounts ID | Maximo Asset Management |

Since the Oracle E-Business Suite charts of accounts are defined at the system level and the Maximo Asset Management charts of accounts are defined at the organization level, the integration framework uses the COAXREF integration control to map the Oracle chart of accounts ID to the Maximo Asset Management organization for the account.

# Journal integration

The general ledger journal integration transfers the following journal transactions from Maximo Asset Management to the Oracle E-Business Suite General Ledger module:

| Transaction | Transaction type |
|---|---|
| Inventory adjustment | INVTRANS |
| Invoice transaction | INVOICETRANS |
| Labor transaction | LABTRANS |
| Issue and transfer | MATUSETRANS |
| Material receipt | MATRECTRANS |
| Service receipt | SERVRECTRANS |
| Tool transaction | TOOLTRANS |

## Journal transfers

The integration framework transfers journals from Maximo Asset Management to Oracle E-Business Suite when the following conditions are met:

- The transaction type is not included in the GLSOURCE integration control.
- The general ledger debit and credit accounts are not null.
- The general ledger debit and credit accounts are fully specified.
- The general ledger debit and credit accounts are different.
- The transaction cost is not zero.

# Integration controls

The general ledger journal uses the following Maximo Asset Management integration controls.

| Control | Description |
|---|---|
| GLCURNCY | Default currency code in Oracle E-Business Suite |
| GLSOURCE | Source business object of the Maximo Asset Management general ledger journals that are skipped by the integration framework |
| JECATXREF | Cross-reference the Maximo Asset Management transaction type and the Oracle E-Business Suite journal category |
| JEPROJSEND | The Maximo Asset Management project transaction types that are sent to the Oracle E-Business Suite |
| ORGXREF | Cross-reference the Maximo Asset Management organization and the Oracle E-Business Suite Ledger ID |
| SITEXREF | Cross-reference the Maximo Asset Management site ID and the Oracle E-Business Suite operating unit |

# Labor and craft integration

# 10

Maximo Asset Management and the Oracle E-Business Suite can exchange the following types of labor data:

| Labor data type | Processing direction |
|---|---|
| Craft | Inbound |
| Labor | Inbound |

There are factors that you must be aware of when integrating Maximo Asset Management and Oracle E-Business Suite labor and craft activity. Some factors include the Oracle E-Business Suite, labor, and craft structure, data mapping and effectivity, and integration control use.

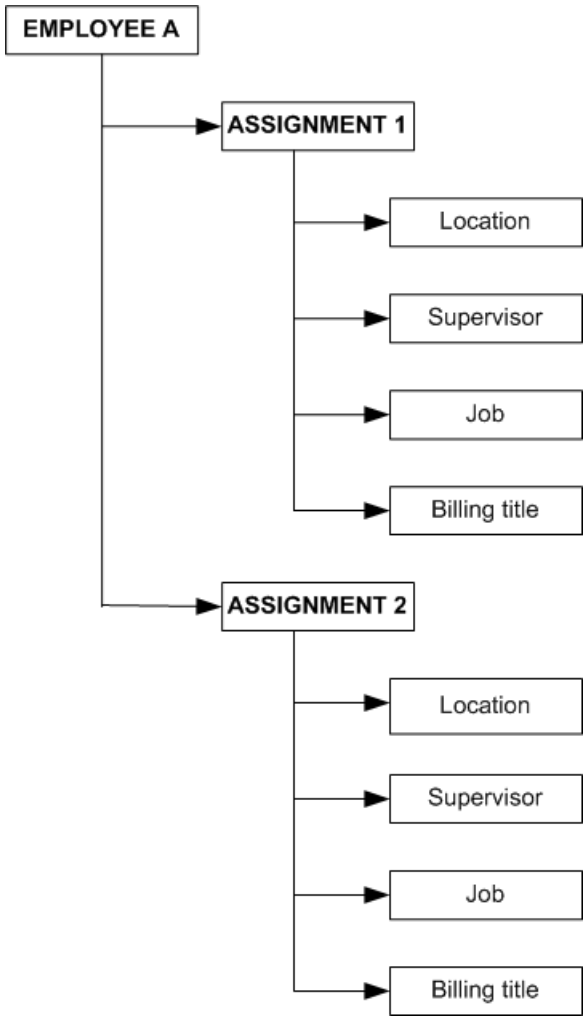
# Labor and craft integration

The labor and craft enterprise services transfer new and updated employee and pay rate data from the Oracle E-Business Suite to Maximo Asset Management.

# Structure of Oracle E-Business Suite assignments

In the Oracle Human Resources (HR), each employee has an assignment, which includes key data such as the location, supervisor, job title, and billing title of the employee. If an employee has multiple assignments, one is designated the primary assignment.

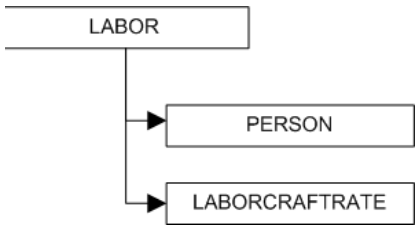
The following diagram shows the employee assignment structure in the Oracle E-Business Suite. The employee and assignment are defined at the business group level, and pay rates are defined at the employee-assignment level.

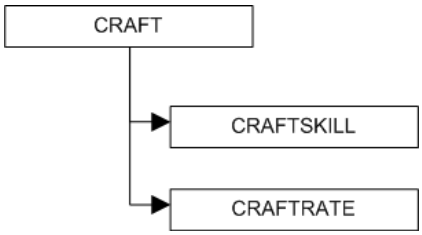


# Structure of the Maximo Asset Management labor and craft data

The Maximo Asset Management labor and craft data exist in separate tables, which are defined at the organization level.

The following diagrams show the labor and the craft structures in Maximo Asset Management:



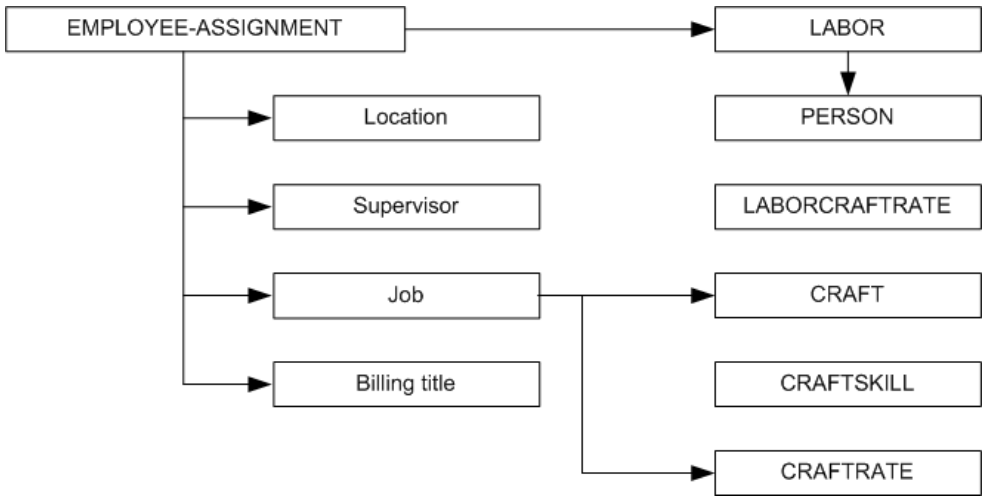




# Employee and job data mapping

The Oracle employee-assignment data maps to the LABOR record in Maximo Asset Management, and the integration framework creates a corresponding PERSON record. The Oracle employee-assignment job data maps to the CRAFT record in Maximo Asset Management. The Oracle employee data is replicated across all Maximo Asset Management organizations.

The integration framework transfers the employee primary assignment only. If the job definition for an employee primary assignment changes, you must use the Maximo Asset Management Labor application to delete the existing employee, job, and rate combination and define a new one.

The following diagram shows a mapping of the Oracle employee assignment to Maximo Asset Management:




The Oracle E-Business Suite users can associate an assignment to an employee for which a rate, but no job (craft), exists. The integration framework requires a job to be associated with an assignment. If no job exists, an integration framework error occurs. Users can write a user exit procedure to add a dummy craft code to inbound transactions that do not include a craft.

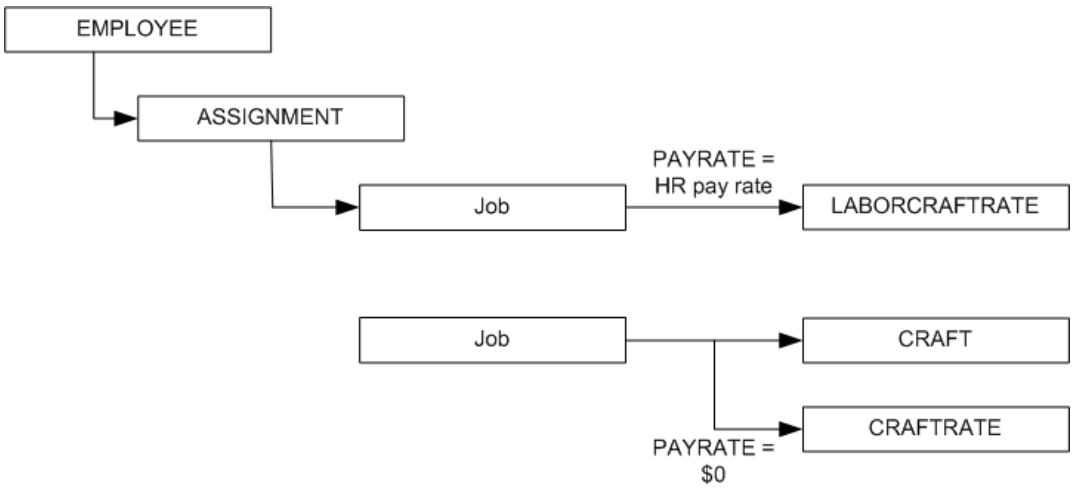
# Pay rate mapping

Pay rates that are set at the assignment level in the Oracle E-Business Suite and are used to map to the pay rate in Maximo Asset Management. The Maximo Asset Management pay rates exist for a labor, craft, and skill combination. Cost rates for labor are defined in the LABORCRAFTRATE table and cost rates for crafts are defined in the CRAFTRATE table.

The integration framework processing populates the pay rate in the following way:

1. Retrieves the job and other required fields to obtain data for the CRAFT and CRAFTRATE object structures.
2. Sets the rate on the CRAFTRATE record to zero.
3. Retrieves the job that is associated with the assignment, to populate the craft on the LABORCRAFTRATE record.
4. Sets the rate on the LABORCRAFTRATE record to the HR pay rate.

The following diagram shows the mapping of the Oracle pay rates to Maximo Asset Management:




# Effectivity of labor and rates

The values in the From Effective Date and To Effective Date fields in the Oracle E-Business Suite People window indicate whether an Oracle employee is active or inactive. If the system date falls between the From and To dates, the account is active.

The SYNCLABOR synchronization script sends active and inactive employees to Maximo Asset Management. If an employee is active, the Status field in the Maximo Asset Management Labor window displays ACTIVE.

The integration framework uses the following logic to evaluate the From and To dates in the Oracle E-Business Suite and set the value of the Status field in Maximo Asset Management:

| Oracle E-Business Suite: From date | Oracle E-Business Suite: To date | Maximo Asset Management: Status value |
|---|---|---|
| Null | Null | Active |
| Null | > or = system date | Active |
| Not applicable | < system date | Inactive |
| > system date | Not applicable | Inactive |
| < or = system date | Null | Active |
| From date <= system date <= To date | From date <= system date <= To date | Active |

The labor integration does not update the Maximo Asset Management employee data when the status of an Oracle employee changes to active or inactive on the basis of the system date. To ensure that Maximo Asset Management uses only active employees, you must run the SYNCLABOR synchronization script frequently.

# Filter

The integration framework transfers labor from the Oracle E-Business Suite to Maximo Asset Management when the following conditions are met:

- The value of the Oracle integration control LCIN is 1.
- The assignment is the employee primary assignment.

# Integration controls

The labor and craft integration uses the following Maximo Asset Management and Oracle integration controls.

| Control | Description | System |
|---|---|---|
| CONVHRS | Hours to convert annual salary to hourly rate | Maximo Asset Management |
| CRAFTXREF | Multiply the Oracle E-Business Suite craft code to the Maximo Asset Management organizations | Maximo Asset Management |
| LABXREF | Multiply the Oracle E-Business Suite labor code to the Maximo Asset Management organization | Maximo Asset Management |
| LCIN | Send the Oracle E-Business Suite labor data to Maximo Asset Management | Oracle E-Business Suite |
| ORGXREF | Cross-reference the Maximo Asset Management organization and the Oracle E-Business Suite ledger ID | Maximo Asset Management |

# Purchase integration 11

Maximo Asset Management and Oracle E-Business Suite can exchange the following types of purchasing data:

| Purchasing data type | Processing direction |
|---|---|
| Vendor | Inbound |
| Purchase requisition | Outbound |
| Contract | Outbound, Inbound |
| Purchase order | Outbound, Inbound |
| Receipt | Outbound, Inbound |
| Invoice | Outbound, Inbound |

There are factors that you must be aware of when integrating Maximo Asset Management and the Oracle E-Business Suite purchase activity. Some factors include the Oracle E-Business Suite purchase configurations, invoice, purchase requisition, contract, receipt, and invoice integration.

# Purchase configurations

There are six common purchasing and invoicing configurations. A shaded square indicates the system in which the document is created.

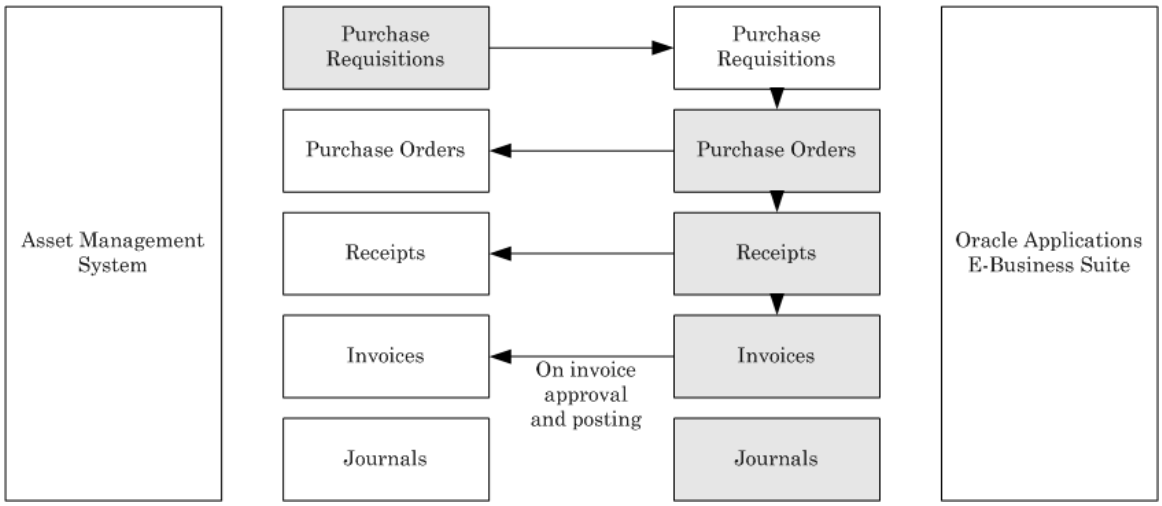
### Configuration 1

In this configuration, only purchase requisitions are entered in Maximo Asset Management. The process of invoice matching takes place in Oracle E-Business Suite, where all the necessary purchase order, receipt, and invoice information is available.

For this configuration to work correctly, you must have implemented, at a minimum, the following the Oracle E-Business Suite modules:

- General Ledger
- Payables
- Purchasing

The following diagram shows the details of the first purchase configuration:



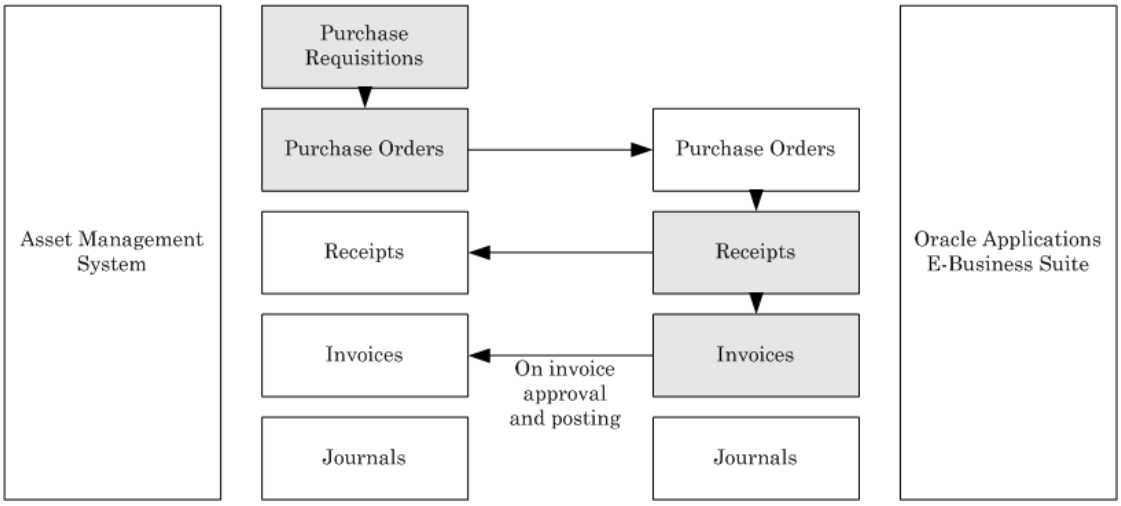

### Configuration 2

In this configuration, most of the purchasing process occurs in Maximo Asset Management. The purchase order has the same one-to-one mapping of the header lines, and distributions as in Maximo Asset Management. Receipts and invoice matching are done in Oracle. E-Business Suite

For this configuration to work correctly, you must have implemented, at a minimum, the following the Oracle E-Business Suite modules:

- General Ledger
- Payables
- Purchasing

The following diagram shows the details of the second purchase configuration:



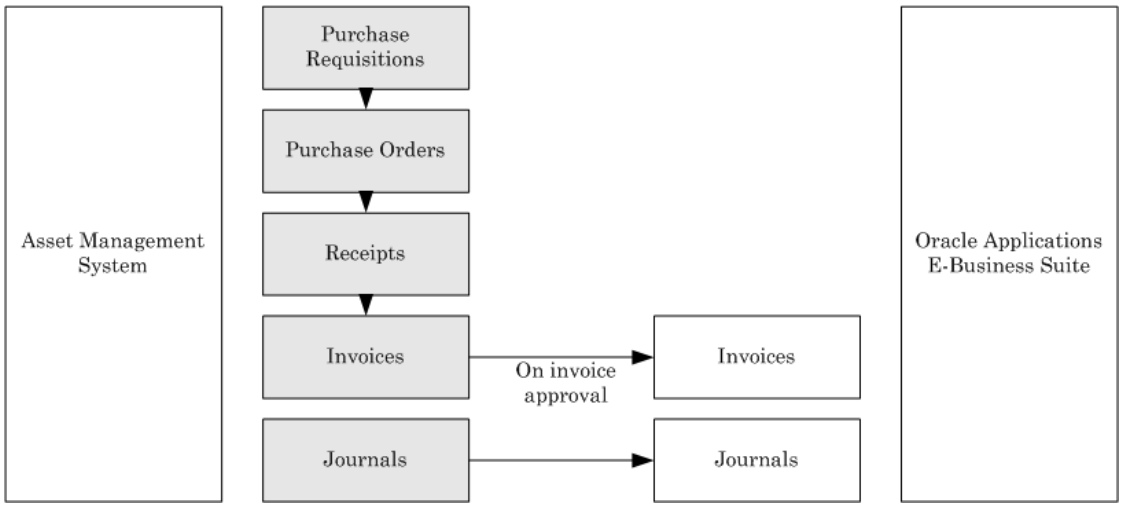

**Configuration 3**

In this configuration, the entire purchasing process occurs in Maximo Asset Management.

For this configuration to work correctly, you must have implemented, at a minimum, the following Oracle E-Business Suite modules:

- General Ledger
- Payables

The following diagram shows the details of the third purchase configuration:




**Configuration 4**

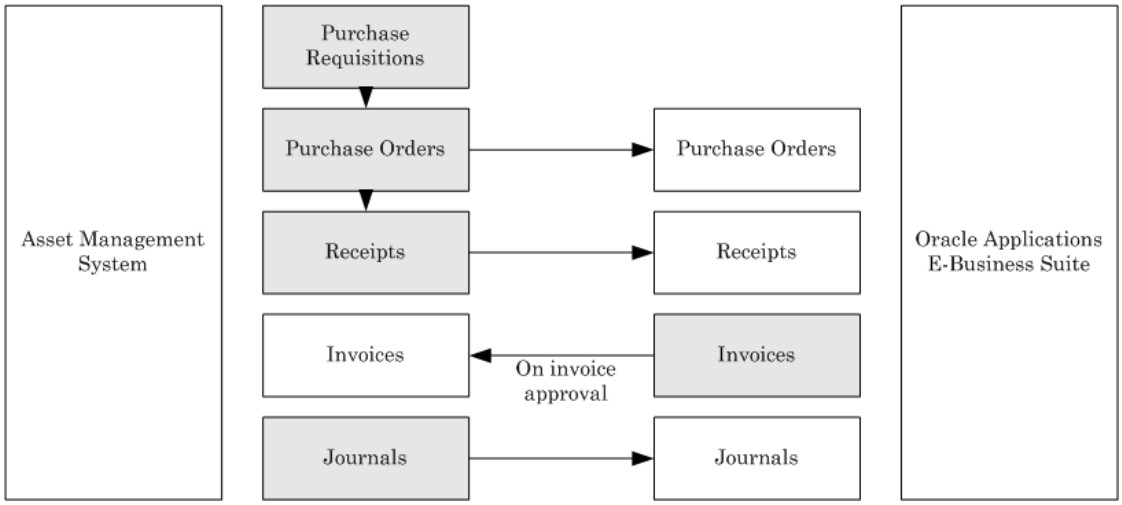
In this configuration, the purchase requisition, purchase order, and receipts are entered in Maximo Asset Management. The purchase order has the same one-to-one mapping of the header lines, and distributions as in Maximo Asset

Management. When transferring the receipts to Oracle E-Business Suite, the integration framework transfers only the receipts line.

For this configuration to work correctly, you must have implemented, at a minimum, the following the Oracle E-Business Suite modules:

- General Ledger
- Payables
- Purchasing

The following diagram shows the details of the fourth purchase configuration:



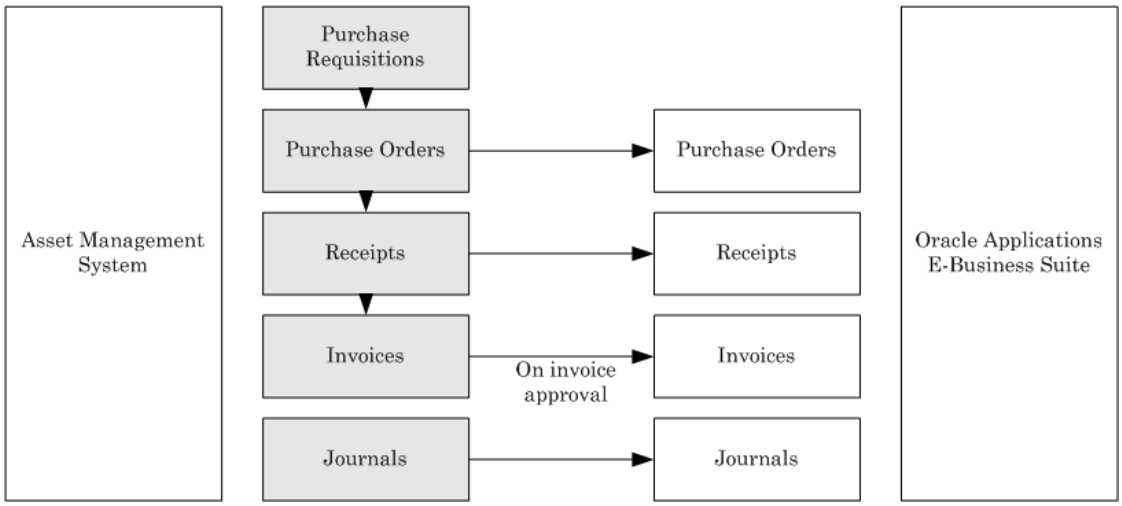

**Configuration 5**

In this configuration, the entire purchasing process occurs in Maximo Asset Management. The purchase order has the same one-to-one mapping of the header lines, and distributions as in Maximo Asset Management. When transferring the receipts to Oracle E-Business Suite, the integration framework transfers only the receipts line.

For this configuration to work correctly, you must have implemented, at a minimum, the following the Oracle E-Business Suite modules:

- General Ledger
- Payables
- Purchasing

The following diagram shows the details of the fifth purchase configuration:


Asset Management System
Purchase Requisitions
Purchase Orders
Receipts
Invoices
Journals
On invoice approval
Purchase Orders
Receipts
Invoices
Journals
Oracle Applications E-Business Suite

# Purchase and invoice considerations

The purchase and invoice documents refer to purchase requisitions, purchase contracts, and purchase orders as a group.

## Purchase sites and pay sites

An Oracle E-Business Suite vendor can be designated a purchasing site, a pay site, or both. A purchase site is used for the purchase of goods and services, while a pay site is used for payment of an invoice.

The integration framework uses the purchase site and pay site settings as follows:

- Passes purchasing and pay site information to Maximo Asset Management on the vendor record
- Validates that vendors on purchase requisitions and purchase orders are purchasing sites
- Validates that vendors on invoices are pay sites

You can use Application Designer of Maximo Asset Management to show the purchasing and pay site value in the Company application. You can change the vendor on an invoice to a different vendor than the one that appears on the purchase order. However, the new vendor must be designated a pay site.

## Status controls

An integration control that is associated with each purchase and invoice publish channel specifies the statuses at which the integration framework sends the document to Oracle E-Business Suite.

The publish channels and their corresponding controls are:

| Publish channel | SEND control |
|---|---|
| Purchase contract | PCSEND |
| Purchase order | POSEND |
| Purchase requisition | PRSEND |
| Invoice | APSEND |

By default, the value of each control is APPR. The integration framework sends the corresponding transaction to Oracle E-Business Suite when the document status is APPR, or approved. If you modify the values in a control, include all synonyms for the statuses that are associated with the control.

# Purchase line types

Oracle E-Business Suite provides two default purchase line types, Goods and Services, and you can define additional types as needed. Maximo Asset Management provides the line types that are shown in the following table. You cannot change the default types or create new ones.

When Oracle Services Procurement is implemented, Oracle E-Business Suite provides two additional line types: fixed price and rate value base.

For outbound purchasing transactions, the integration framework uses the LINETYPEOUTXREF integration control to cross-reference Maximo Asset Management line types and the standard Oracle E-Business Suite line types.

| Maximo Asset Management line type | Oracle E-Business Suite line type |
|---|---|
| ITEM | Goods |
| MATERIAL | Goods |
| SERVICE | Services |
| STDSERVICE | Services |
| SPORDER | Goods |
| EXTERNAL | Goods |
| SERVICE | Fixed price |
| SERVICE | Rate |

Since inbound transactions may contain user-defined line types that the integration framework does not recognize, the inbound processing derives Maximo Asset Management line type from the value basis of the line and the presence or absence of an item number. Oracle line types are defined a value basis, as follows:

| Oracle value basis | Description |
|---|---|
| Amount | Order, receive, and invoice based on the value of goods or services provided. |
| Quantity | Order, receive and invoice based on the quantity of services provided. The integration framework treats outside processing as quantity-based. |
| Fixed price | If Oracle Services Procurement is implemented, you must enter and receive services by amount only. You cannot change the unit of measure and the unit price on purchasing document lines. |
| Rate | If Oracle Services Procurement is implemented, you must enter services by rate (price) and amount. You must also receive items by amount. |

The following table shows how the integration framework derives the line types. For inbound transactions, the integration framework assigns line types ITEM and SERVICE only.

| Value basis | Item number | Line type |
|---|---|---|
| Amount | N/A | SERVICE |
| Quantity | No | SERVICE |
| Quantity | Yes | ITEM |
| Fixed price | N/A | SERVICE |
| Rate | N/A | SERVICE |

The derivation of line type is not necessary when an inbound purchase order references a PR and PR line, since the integration framework can retrieve the line type from the PR line. Likewise, if an inbound invoice references a PO line, the integration framework can retrieve the line type of the PO line.

## Purchase order line type derivation

The following example illustrates how the integration framework derives the purchase order line type.

1. A PO with no association to a Maximo Asset Management requisition is created in Oracle E-Business Suite.
2. A PO line with an amount-based line type is entered in Oracle E-Business Suite.

   This example uses the following values:

| Column | Value |
|---|---|
| Line type | SERVICE |
| Quantity | 1000 |
| Unit of measure | Dollars |
| Unit price | 1 |

3. The Oracle E-Business Suite PO is approved.
4. The integration framework passes the PO to Maximo Asset Management.

   In Maximo Asset Management, the PO line item displays the following data for the amount-based line type:

| Column | Value |
|---|---|
| Quantity | Null |
| Unit of measure | Dollars |
| Unit cost | Null |
| Line cost | 1000 |

5. The PO line item is fully received in Maximo Asset Management.
6. The integration transfers the receipt to Oracle E-Business Suite.
7. The Oracle import process imports the receipts into Oracle E-Business Suite.
8. An invoice is created in Oracle E-Business Suite with a cost of $995.
9. Accounting for the invoice is created, and the invoice is validated.
10. The integration framework passes the invoice to Maximo Asset Management.

    Variances are not created for invoice lines with a null quantity. If the invoice was for $1100, an error would occur in Oracle E-Business Suite, since the invoice quantity cannot be greater than the received quantity.

## Null item numbers

On outbound purchasing documents, the integration framework sets the item number on the following line types to null:

- MATERIAL
- SERVICE
- STDSERVICE
- TOOL

## Destination type codes

The Oracle E-Business Suite purchase requisition and purchase order lines contain a destination type code, which indicates the final destination of the items.

Valid destination type codes in Oracle E-Business Suite

| Destination type code | Description |
|---|---|
| Expense | The goods are delivered to the requester at the expense location. |
| Inventory | The goods are received into inventory upon delivery. |
| Shop floor | The goods are delivered to an outside processing operation defined by work in process. |

Although Shop Floor is a valid destination in Oracle E-Business Suite, the integration framework uses only the Expense and Inventory destination codes.

The integration framework stores the destination type code values in the following integration controls.

| Control | Description |
|---|---|
| DTC_EXP | Used for direct issues and assigned to the expense the Oracle E-Business Suite value. A direct issue item is an item that you have ordered that is issued immediately upon receipt to an asset, location, or work order, rather than stocked in a storeroom. |
| DTC_INV | Used for replenishment and assigned to the inventory the Oracle E-Business Suite value. A replenishment item is an item that are stocked in a storeroom. |

The following factors determine whether the integration framework retrieves the destination type code from the DTC_EXP or the DTC_INV control:

| | |
|---|---|
| Is the item reorderable? | If the REORDER flag is enabled, then the item is reorderable. |
| Is the item masked? | If the GENITEM integration control contains a dummy item rather than a null value, the item is masked. |
| Is the storeroom masked? | If the GENSTORE integration control contains a dummy storeroom rather than a null value, the storeroom is masked. |
| Is the purchase order line direct issue? | • Items for storeroom replenishment (the value of PO.ISSUE or PR.ISSUE is<br>• N)<br>• Items for direct issue to a work order, GL account, equipment, or location<br>• in Maximo Asset Management (the value of PO.ISSUE or PR.ISSUE is Y)<br>• Items for an internal requisition |

The integration framework determines the destination type code as follows:

The destination type code is DTC_INV, if the item meets following conditions:

- For Oracle owned items, the reorder flag for the item is set to true
- the item is not a direct issue
- the item is not masked or it is an Oracle-owned item
- the Storeroom is not masked.

In all other cases, the destination type code is DTC_EXP.

# Multicurrency control

When you use the Purchasing applications of Maximo Asset Management, you can track transactions in a second currency, for example, for Euro currency reporting. Oracle E-Business Suite uses Multiple Reporting Currencies (MRC) for the same purpose. However, Maximo Asset Management allows only one additional currency, while Oracle E-Business Suite allows multiple reporting currencies for a single base (primary) currency.

All transactions between the two systems use the conversion rates in Oracle E-Business Suite. The adapter uses these conversion rates to calculate the reporting currency for outbound transactions and the Maximo Asset Management second currency amount for inbound applications.

The GLSOBID2 Oracle integration control contains the identifier of the reporting ledger in Oracle E-Business Suite. The currency for that ledger must match the Maximo Asset Management second currency code.

## Encumbrance and liability management

Maximo Asset Management and Oracle E-Business Suite differ in their approaches to encumbrance and liability management. Oracle E-Business Suite allows multiple companies, departments, organizations, subsidiaries, and so on. In Oracle E-Business Suite you manage encumbrance and liability at the organization level. In Maximo Asset Management, you can manage encumbrance and liability in more detail by assigning AP suspense and RBNI accounts at the vendor level.

Maximo Asset Management receives the AP control account, but not the AP suspense or the RBNI account, for each Oracle E-Business Suite vendor site. If you configure the Maximo Asset Management vendor default accounts before synchronizing vendors, the AP suspense and RBNI accounts default from the vendor. You configure vendor defaults in the Company-Related Accounts window, which you access in the Select Action menu in the Chart of Accounts application.

If you do not configure the Maximo Asset Management vendor defaults and you use the third purchasing configuration, you can use Maximo Asset Management to enter the AP suspense account and the RBNI account for each Oracle E-Business Suite vendor site. Invoice processing in Maximo Asset Management requires all three accounts.

## Loaded costs

When an outbound invoice line is matched to a purchase order and the line cost differs from the loaded cost (for example, the user applied taxes or prorated services), the integration framework calculates the difference between the line cost and the loaded cost for the line, and creates an invoice distribution, with type MISCELLANEOUS, on the invoice in the Oracle E-Business Suite.

## Cost distributions

The integration framework does not support multiple cost distributions (distribution of the cost of an item across multiple general ledger accounts). While Oracle E-Business Suite supports multiple cost distributions, Maximo Asset Management does not.

## Multilingual support

The integration framework supports multilingual vendor addresses in inbound transactions.

# Vendor integration

The vendor integration transfers new and updated vendors, vendor sites, and vendor site contacts from Oracle E-Business Suite to Maximo Asset Management.

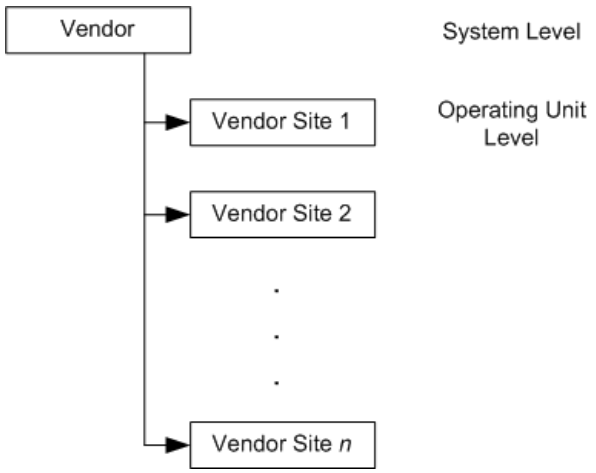
If you create a vendor in Oracle E-Business Suite, you must make all updates to that vendor in Oracle E-Business Suite. If you update the vendor in Maximo Asset Management, the change does not appear in Oracle E-Business Suite.

If you update a vendor site contact in Oracle E-Business Suite, that update is sent to Maximo Asset Management.

## Structure of vendor data

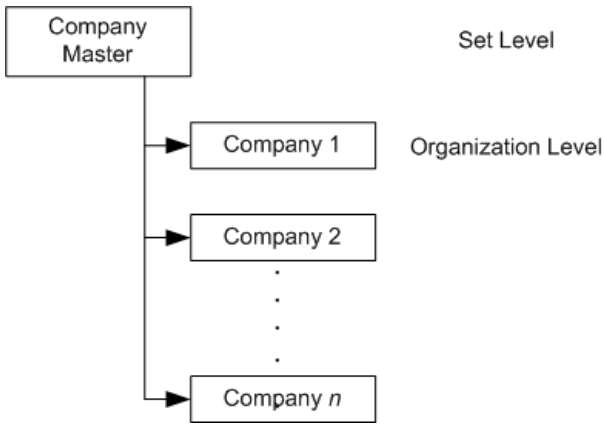
Oracle vendor data consists of a vendor header at the system level and one or more vendor sites are at the operating unit level, which is the equivalent of the Maximo Asset Management site level.

The following diagram shows the vendor structure in Oracle E-Business Suite:




Maximo Asset Management vendor data consists of a vendor master at the set level and one or more vendor records at the organization level. An organization has only one vendor set, but a vendor set can belong to more than one organization.

Maximo Asset Management refers to vendors as companies. The following diagram shows the company structure in Maximo Asset Management:

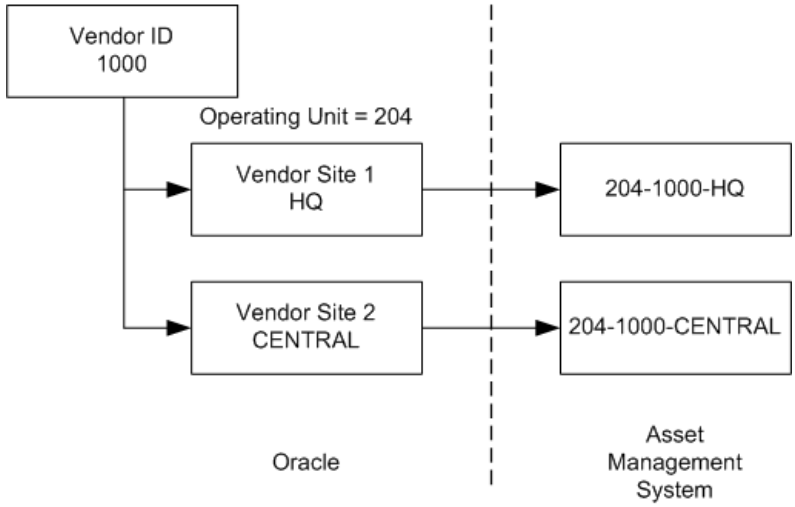



The integration framework passes the Oracle E-Business Suite vendor site records to Maximo Asset Management organization level. The integration framework determines Maximo Asset Management set identifier from the organization information about the inbound vendor record.

Before using the integration framework, you must configure the relationship between company master sets and Maximo Asset Management organizations in the Sets application.

The company identifier in Maximo Asset Management is a concatenation of the Oracle E-Business Suite operating unit, vendor ID (SEGMENT1), and vendor site code.

The following diagram shows the mapping of the Oracle vendor and site identifiers to Maximo Asset Management:




The integration framework does not create the company masters. To direct Maximo Asset Management to create a company master for each inbound vendor it creates, select the Automatically Add Companies to Company Master check box in the Sets application. If you do not, errors occur.

Although Maximo Asset Management allows a one-to-many relationship between company masters and company record, there is a one-to-one relationship for vendors added by using the vendor integration.

## Effectivity of vendors

The value in the Inactive Date field in the Oracle E-Business Suite Suppliers window determines whether an Oracle vendor is active or inactive. If a vendor header is inactive, its vendor sites are all inactive. If a vendor header is active, the Inactive On date determines the status of each vendor site. If the system date is earlier than the Inactive On date, the vendor is active.

The integration framework passes the following fields to the PL/SQL user exit. If necessary, you can write a user exit procedure to include these fields in the interface table.

- AP_SUPPLIERS.START_DATE_ACTIVE
- AP_SUPPLIERS.END_DATE_ACTIVE
- AP_SUPPLIER_SITES_ALL.INACTIVE_DATE

The SYNCVND synchronization script sends active and inactive vendors to Maximo Asset Management. If a vendor is active, the Disqualified Vendor check box in Maximo Asset Management Company Master and Companies windows is not selected.

The integration framework uses the following logic to evaluate the INACTIVE_DATE and END_DATE_ACTIVE fields in the Oracle E-Business Suite

and set the value of the Disqualified Vendor check box in Maximo Asset Management.

| Oracle E-Business Suite | | Maximo Asset Management |
|---|---|---|
| **End date active** | **Inactive date** | **Disqualified vendor value** |
| Null | Null | Not checked (vendor header) |
| Null | > system date | Not checked (vendor site) |
| Null | < or = system date | Checked (vendor site) |
| > system date | | Not checked (vendor header) |
| > system date | > system date | Not checked (vendor site) |
| > system date | < or = system date | Checked (vendor site) |
| < or = system date | any value | Checked (vendor header and vendor site) |

The vendor integration does not update Maximo Asset Management vendors when the status of an Oracle vendor changes to active or inactive based on the system date. To ensure that Maximo Asset Management uses only active vendors, run the SYNCVND synchronization script frequently.

For more information about the synchronization script, see the *IBM Maximo Enterprise Adapter for Oracle Applications Configuration Guide.*

## Filter

The integration framework transfers vendors from the Oracle E-Business Suite to Maximo Asset Management when the following conditions are met:

- The value of the Oracle integration control COMIN is 1.
- The vendor is designated a purchasing site, a pay site, or both in Oracle E-Business Suite.

## Effectivity of vendor site contacts

The Inactive Date on the Oracle E-Business Suite Suppliers Contact Directory field determines if the contact is active or not. If the inactive date is less than the current system date, the contact is inactive for all associated sites in the Oracle E-Business Suite.

During the execution of the vendor sync script, Maximo Enterprise Adapter for Oracle sends only active contacts that are associated with sites (Addresses). If the inactive date is updated in the Oracle E-Business Suite Suppliers Contact Directory window, the adapter deletes the contacts in Maximo Asset Management.

The following guidelines are followed for contact deletion:

| Oracle E-Business Suite | Maximo Asset Management |
|---|---|
| Inactive Date < system date | Contact deleted |
| Inactive Date > = system date | Contact is not deleted |

The vendor integration does not update vendor site contacts when the status of an Oracle vendor site contact changes to active or inactive on the basis of the system date. To ensure that Maximo Asset Management uses only active vendor site contacts, you must run the SYNCVND synchronization script frequently.

## Filter

The integration framework transfers vendor contacts from Oracle E-Business Suite to Maximo Asset Management when the value of the Oracle integration control COMIN is 1.

## Integration controls

The vendor integration uses the following Maximo Asset Management and Oracle integration controls.

| Control | Description | System |
|---|---|---|
| COMIN | Send the Oracle E-Business Suite vendor data to Maximo Asset Management | Oracle E-Business Suite |
| GENUSR | Dummy identifier for masking user ID in outbound transactions | Maximo Asset Management |
| ORGXREF | Cross-reference Maximo Asset Management organization and the Oracle E-Business Suite ledger ID | Maximo Asset Management |

# Purchase requisition integration

The purchase requisition integration transfers purchase requisitions from Maximo Asset Management to the Oracle E-Business Suite when the status of the purchase order matches a status in the PRSEND integration control.

If an error occurs when the Oracle E-Business Suite is importing a PR, the integration framework sends a status change transaction to Maximo Asset Management. That transaction changes the PR status in Maximo Asset Management to WAPPR (waiting for approval).

## Filter

The integration framework transfers purchase requisitions from Maximo Asset Management to the Oracle E-Business Suite when the status of the PR is equal to a status in the PRSEND control.

## Integration controls

The purchase requisition integration uses the following Maximo Asset Management integration controls.

| Control | Description | System |
|---|---|---|
| DTC_EXP | Destination type code for direct issue PO lines | Maximo Asset Management |
| DTC_INV | Destination type code for replenishment PO lines | Maximo Asset Management |
| GENITEM | Dummy identifier for masking item number | Maximo Asset Management |
| GENSTORE | Dummy identifier for masking storeroom in outbound transactions | Maximo Asset Management |
| GENUSR | Dummy identifier for masking user ID in outbound transactions | Maximo Asset Management |
| GLCURNCY | Default GL currency code in the Oracle E-Business Suite | Maximo Asset Management |
| GLCURNCYTYPE | Type of currency exchange rates | Maximo Asset Management |
| LINETYPEXREF | Cross-reference Maximo Asset Management line type (synonyms) and the Oracle E-Business Suite line type | Maximo Asset Management |
| OACATXREF | Cross-reference Maximo Asset Management line type and the Oracle E-Business Suite item or service category identifier | Maximo Asset Management |
| ORGXREF | Cross-reference Maximo Asset Management organization and the Oracle E-Business Suite ledger ID | Maximo Asset Management |

| Control | Description | System |
|---|---|---|
| PRSEND | Statuses at which purchase requisitions are sent from Maximo Asset Management to the Oracle E-Business Suite | Maximo Asset Management |
| PRSTATUSXREF | Cross-reference Maximo Asset Management internal and external PR status, if using synonyms | Maximo Asset Management |
| SITEXREF | Cross-reference Maximo Asset Management site ID and the Oracle E-Business Suite operating unit | Maximo Asset Management |
| USEPRNUM | Specifies if integration framework uses Maximo Asset Management PR number or autogenerated PR number | Maximo Asset Management |

# Purchase order integration

The purchase order interfaces support the transfer of new and updated purchase orders from Maximo Asset Management to Oracle E-Business Suite. The data transfer uses Oracle open interface tables and purchasing public APIs.

The purchase order interface uses standard concurrent processes and the Purchasing documents open interface for creating purchase orders and purchase contracts. After a purchase order and a purchase contract is approved in Maximo Asset Management, the adapter adds records to the following tables:

- PO_HEADERS_INTERFACE
- PO_LINES_INTERFACE
- PO_DISTRIBUTIONS_INTERFACE

The table shows the updated information sent by the purchase order interface to Oracle E-Business Suite from Maximo Asset Management.

| Changes in Maximo Asset Management that trigger the transfer of updated information to Oracle E-Business Suite | Method used to transfer updates from Maximo Asset Management to Oracle E-Business Suite |
|---|---|
| Updates to the Quantity, Price, Need By, and Promise Date fields in purchase orders | Oracle standard APIs |
| Updates to the Contract price field | Purchasing documents open interface |
| Cancellation of purchase orders and contracts | Oracle standard APIs |

The releases in Maximo Asset Management are sent as standard purchase orders to Oracle E-Business Suite.

The purchase order integration transfers purchase orders from Maximo Asset Management to Oracle E-Business Suite when the status of the purchase order matches a status in the POSEND integration control.

The purchase order integration also transfers approved purchase orders from the Oracle E-Business Suite to Maximo Asset Management. If an error occurs when the Oracle Application E-Business Suite is importing a PO, the integration framework sends a status change transaction to Maximo Asset Management. That transaction changes the PO status in Maximo Asset Management to WAPPR (waiting for approval).

The purchase order integration does not support the transfer of internal purchase orders from Maximo Asset Management to Oracle E-Business Suite.

# Structure of purchase order line numbers

On inbound transactions, the POLINE number is a concatenation of the number 1, the four-digit Oracle line number, the four-digit Oracle shipment number, and the four-digit Oracle distribution number (for example, 1000500020001). The POLINE cannot contain multiple cost distributions, and all lines must go to the same site.

# Transfer of purchase order revision information

The details on a purchase order can be changed during the course of its lifecycle. When a purchase order is changed, its revision number is updated.

The purchase integration supports purchase order revision for inbound and outbound transactions. You can send purchase order revision information from Oracle E-Business Suite to Maximo Asset Management and vice versa.

## Transfer of inbound purchase order revision

When you create or update a purchase order in Oracle E-Business Suite, and transfer it to Maximo Asset Management, a new revision of the purchase order is created by the integration framework.

When you create a purchase order in Oracle E-Business Suite, and then transfer it to Maximo Asset Management, the revision number is set to 0. However, if the purchase order in Oracle E-Business Suite has a revision number before it is transferred to Maximo Asset Management, then this revision number is assigned to the transferred purchase order in Maximo Asset Management.

When you update the transferred purchase order in Oracle E-Business Suite, Maximo Asset Management creates a revision of the purchase order. The new revision contains the purchase order updates that you made in Oracle E-Business Suite.

When you cancel a purchase order in Oracle E-Business Suite, it will also be canceled in Maximo Asset Management.

## Receipts processing

You can transfer related records, such as receipts or invoices, from Oracle E-Business Suite to Maximo Asset Management. The related records are applied to the active revision of the purchase order. Partial receipts are also applied to the active revision of the purchase order in Maximo Asset Management.

You can control whether receipts are processed in Maximo Asset Management by setting the Allow receipts flag. If the Allow receipts flag is set to Yes, the receipt will be processed in Maximo Asset Management. If the Allow receipts flag is set to No, the receipt will not be processed in Maximo Asset Management.

You set the Allow receipts flag to Yes or No, depending on whether you create purchase orders and manage receipts in Maximo Asset Managment or Oracle E-Business Suite. The following table shows the settings of the Allow receipts flag.

Settings for the Allow receipt flag

| Create purchase order | Manage receipts | Allow receipts flag setting |
|---|---|---|
| Maximo Asset Management | Oracle E-Business Suite | Yes |
| Maximo Asset Management | Maximo Asset Management | Yes or No |
| Oracle E-Business Suite | Oracle E-Business Suite | Yes |
| Oracle E-Business Suite | Maximo Asset Management | No |

## Transfer of outbound purchase order revision

The transfer of purchase order revision information from Maximo Asset Management to Oracle E-Business Suite has some specific scenarios because of how the purchase order revision feature works in both products.

The transfer of outbound purchase order revision works in the following way:

- In Maximo Asset Management, when you create a purchase order and run the concurrent job to send the transaction to Oracle E-Business Suite, a purchase order revision 0 is created in the Oracle ERP software.
- In Maximo Asset Management, if you revise a purchase order, the revised purchase order is not sent to Oracle E-Business Suite until you approve it and run the applicable concurrent job in Maximo. Then Maximo Asset Management sets purchase order revision 0 to revised. Oracle E-Business Suite creates purchase order revision 1 in approved status and archives purchase order revision 0.
- Oracle E-Business Suite does not skip revision numbers. You can create a purchase order in Maximo Asset Management and revise it to revision 2 before you transfer it to Oracle E-Business Suite. Then when you transfer it to Oracle E-Business Suite, Oracle E-Business Suite creates purchase order revision 1.
- Related records, for example, outbound receipts and invoices, are applied to the active revision of the purchase order.
- Outgoing partial receipts are applied to the active revision of the purchase order in Oracle E-Business Suite.

## Out of sync purchase order revision information

Maximo Asset Management and Oracle E-Business Suite handle purchase order revision information differently. During the purchase order life cycle, if you create and manage purchase orders in Maximo Asset Management, the related revision information might become out of sync between the two products.

The revision information might be different in Maximo Asset Management and Oracle E-Business in the following scenarios:

- In Maximo Asset Management, you can cancel a purchase order without updating the revision number. However, when the cancellation information is transferred to Oracle E-Business Suite, the purchase order is canceled and the revision number is updated.

- In Maximo Asset Management, you can delete lines from purchase order, without revising it. However, when the information is transferred to Oracle E-Business Suite, the lines in the purchase order are canceled and the revision information is updated.

In Maximo Asset Management, if you revise a purchase order without updating the Quantity, Price, Need by, and Promise Date fields, Maximo Asset Management does not send the updated information to Oracle E-Business Suite. As a result, Oracle E-Business Suite does not create a revision.

## Purchase order receiving tolerance

Purchase Order Receiving Tolerance provides a mechanism to control over-receipt situations in an enterprise.

The purchase order integration transfers receiving tolerance defined at the purchase order line level in Oracle E-Business Suite. These receiving tolerance are only transferred if the over-receipt action is defined as Reject in Oracle. This is because in Maximo Asset Management, the over-receipt action is always reject and will not allow the user to receive beyond what is defined within the receiving tolerance parameters.

Similarly, receiving tolerance on the purchase order line is transferred from Maximo Asset Management to Oracle E-Business Suite through the purchasing integration. The over-receipt action on the PO shipment line in Oracle is set as Reject.

The receiving tolerance should be defined in the system where the purchase order will be created and managed.

## Purchase order ownership

The integration framework defines an Oracle-owned purchase order as one that is created and managed in the Oracle E-Business Suite, and a Maximo Asset Management -owned purchase order as one that is created and managed in Maximo Asset Management.

The integration framework determines purchase order ownership by the value in the OWNERSYSID column of Maximo Asset Management PO.

| Purchase order owner | OWNERSYSID value |
|---|---|
| Maximo Asset Management | Null |
| Oracle E-Business Suite | OA |

## Filter

The integration framework transfers purchase orders from Maximo Asset Management to the Oracle E-Business Suite when the following conditions are met:

- The status of the PO is equal to a status in the POSEND integration control.
- The purchase order is not an internal purchase order.

The integration framework transfers purchase orders to Maximo Asset Management when the following conditions are met:

- The value of the Oracle integration control POIN is 1.
- The purchase order is approved.

## Integration controls

The purchase order integration uses the following Maximo Asset Management and Oracle integration controls.

| Control | Description | System |
|---|---|---|
| DTC_EXP | Destination type code for direct issue PO lines | Maximo Asset Management |
| DTC_INV | Destination type code for replenishment PO lines | Maximo Asset Management |
| GENITEM | Dummy identifier for masking item number | Maximo Asset Management |
| GENSTORE | Dummy identifier for masking storeroom in outbound transactions | Maximo Asset Management |
| GENUSR | Dummy identifier for masking user ID in outbound transactions | Maximo Asset Management |
| GLCURNCYTYPE | Type of currency exchange rates | Maximo Asset Management |
| LINETYPEXREF | Cross-reference Maximo Asset Management line type (synonyms) and the Oracle E-Business Suite line type | Maximo Asset Management |
| OACATXREF | Cross-reference Maximo Asset Management line type and the Oracle E-Business Suite item or service category identifier | Maximo Asset Management |
| ORGXREF | Cross-reference Maximo Asset Management organization and the Oracle E-Business Suite ledger ID | Maximo Asset Management |
| POIN | Send the Oracle E-Business Suite purchase order data to Maximo Asset Management | Oracle E-Business Suite |
| POSEND | Statuses at which purchase orders are sent from Maximo Asset Management to the Oracle E-Business Suite | Maximo Asset Management |
| POSTATUSXREF | Cross-reference Maximo Asset Management internal and external PO status, if using synonyms | Maximo Asset Management |

| Control | Description | System |
|---|---|---|
| POTYPEXREF | Cross-reference Maximo Asset Management and the Oracle E-Business Suite purchase order types | Maximo Asset Management |
| SITEXREF | Cross-reference Maximo Asset Management site ID and the Oracle E-Business Suite operating unit | Maximo Asset Management |

# Contract integration

The contract outbound interfaces support the transfer of new and updated contracts from the Maximo Asset Management to Oracle E-Business Suite. The data transfer takes place using Oracle open interface tables and purchasing public APIs.

The contract integration has the following capabilities:

- Uses standard concurrent processes and the purchasing documents open interface for creating purchase contracts.
- After a purchase contract is approved in Maximo Asset Management, the adapter adds records to the following tables:
  - PO_HEADERS_INTERFACE
  - PO_LINES_INTERFACE
  - PO_DISTRIBUTIONS_INTERFACE
- Sends updated information to Oracle E-Business Suite using purchasing documents open interface when the Contract Price field is updated.
- Sends updated information to Oracle E-Business Suite using Oracle standard APIs, when Maximo Asset Management contracts are canceled.

The contract integration transfers contracts from Maximo Asset Management to the Oracle E-Business Suite when the status of the contract matches a status in the PCSEND integration control. The approved contracts are also transferred from the Oracle E-Business Suite to Maximo Asset Management.

## Types of contracts

Maximo Asset Management Contracts application recognizes the following types of contracts:

- Blanket
- Labor
- Lease
- Pricing
- Purchase
- Rental
- Service
- Software license
- Master
- Warranty

The outbound integration sends only blanket contracts from Maximo Asset Management to the Oracle E-Business Suite. A Maximo Asset Management blanket contract is an agreement to purchase materials or services from a particular vendor at an agreed upon price and terms. To place an actual order, you must issue a blanket release against a blanket purchase agreement.

Unlike Maximo Asset Management, the Oracle E-Business Suite includes contracts as part of the Purchasing module. The purchasing module recognizes the following types of documents.

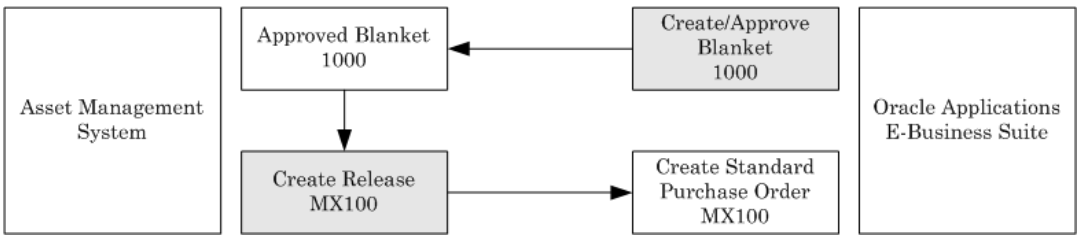
- Blanket purchase agreement
- Blanket releases
- Contract purchase agreements
- Global agreements
- Planned purchase orders
- Scheduled releases
- Standard

The inbound integration treats Oracle blanket purchase agreements and contract purchase agreements as contracts.

An Oracle blanket purchase agreement is equivalent to a Maximo Asset Management blanket contract. An Oracle contract purchase agreement is an agreement with a supplier, with specific terms and conditions, that does not specify the goods and services that are purchased. You later issue a standard purchase order that references the contract. The lines on the contract are read-only and you can input data only on the contract header and in the terms and conditions.

## Structure of contract numbers

The purchase order number for an inbound blanket release is a concatenation of the blanket agreement number, a hyphen, and the release number. In the following diagram, the purchase order number of the first release against blanket order 1000 is 1000-1.




## Status of contracts

In the Oracle E-Business Suite, blanket and contract purchase agreements use the same statuses as purchase orders. Valid status values are:

- Incomplete
- Approved
- Finally closed
- Canceled

Valid contract status values in Maximo Asset Management are:

- DRAFT
- WAPPR
- APPR

- PNDREV
- REVISD
- SUSPEND
- EXPIRD
- CANCEL
- CLOSE
- WSTART

The PCSEND integration control contains the statuses at which Maximo Asset Management sends contracts to the Oracle E-Business Suite. The default value of the control is APPR (approved).

## Filter

The integration framework transfers contracts from Maximo Asset Management to the Oracle E-Business Suite when the following conditions are met:

- The status of the contract is equal to a status in the PCSEND integration control.
- The contract is a blanket contract.

The integration framework transfers contracts from the Oracle E-Business Suite to Maximo Asset Management when the following conditions are met:

- The value of the Oracle integration control PCIN is 1.
- The contract is a blanket purchase agreement, contract purchase agreement, or planned purchase order**.**
- The status of the contract is approved.

# Integration controls

The contract integration uses the following Maximo Asset Management and Oracle integration controls.

| Control | Description | System |
|---|---|---|
| GENITEM | Dummy identifier for masking item number | Maximo Asset Management |
| GENUSR | Dummy identifier for masking user ID in outbound transactions | Maximo Asset Management |
| GLCURNCYTYPE | Type of currency exchange rates | Maximo Asset Management |
| LINETYPEXREF | Cross-reference Maximo Asset Management line type (synonyms) and the Oracle E-Business Suite line type | Maximo Asset Management |
| OACATXREF | Cross-reference Maximo Asset Management line type and the Oracle E-Business Suite item or service category identifier | Maximo Asset Management |
| OAPCDEFORDERUNIT | Order unit to default on PC, when no unit of measure specified on purchase order line | Maximo Asset Management |
| ORGXREF | Cross-reference Maximo Asset Management organization and the Oracle E-Business Suite ledger ID | Maximo Asset Management |
| PCIN | Send the Oracle E-Business Suite contract data to Maximo Asset Management | Oracle E-Business Suite |
| PCSEND | Statuses at which contracts are sent from Maximo Asset Management to the Oracle E-Business Suite | Maximo Asset Management |
| PCSTATUSXREF | Cross-reference Maximo Asset Management internal and external contract status, if using synonyms | Maximo Asset Management |
| PCTYPEXREF | Cross-reference Maximo Asset Management and the Oracle E-Business Suite contract types | Maximo Asset Management |
| SITEXREF | Cross-reference Maximo Asset Management site ID and the Oracle E-Business Suite operating unit | Maximo Asset Management |

The PCTYPEXREF control cross-references the Oracle and Maximo Asset Management PO types, as follows:

| Oracle document | Oracle PO type | Maximo Asset Management PO type |
|---|---|---|
| Blanket purchase agreement | BLANKET | BLANK |
| Contract purchase agreement | CONTRACT | BLANK |
| Planned purchase order | PLANNED | BLANK |

# Receipt integration

The receipt integration transfers completed receipts from Maximo Asset Management to the Oracle E-Business Suite and completed receipts from the Oracle E-Business Suite to Maximo Asset Management.

You must complete the receiving, inspection, and serialization in the same system. For example, if you perform receiving in the Oracle E-Business Suite, a serialized item that requires inspection must be inspected and serialized in the Oracle E-Business Suite.

## Receipt tolerance

The first time that you create an item or a service in the Item Master application, you can specify the percentage of items or services that can be received over the amount that is ordered in the initial purchase order. The percentage value is applied to the organization level and to the vendor level for an item or a service.

When you add lines to a purchase order, you can modify the percentage of items or services that can be received over the amount that is ordered in the initial agreement. You can also choose to specify either a quantity or an amount, instead of a percentage. The receipt tolerance must be defined in the system where the purchase order is created and managed.

The purchase order integration process transfers the receipt tolerance specified at the purchase order line item level in Oracle E-Business Suite. Receipt tolerances are only transferred if the receipt tolerance is defined as Reject in Oracle. In Maximo Asset Management, the receipt tolerance is rejected if the amount of services or materials received exceeds the specified receipt tolerance amount.

Similarly, the receipt tolerance on the purchase order line item is tranferred from Maximo Asset Management to Oracle E-Business Suite through the purchasing integration. The receipt tolerance specified on the PO shipment line in Oracle is set as Reject. For details on receipt tolerance in Maximo Asset Management, refer to the Maximo Asset Management information center. See also the *Oracle E-Business Suite Purchasing User Guide*.

## Lotted items

The integration framework supports lotted items under the following conditions:

- Inventory costs and balances are managed in the Oracle E-Business Suite.
- Purchase orders and invoices are created in the Oracle E-Business Suite.

## Rotating items

Maximo Asset Management refers to an item under serial control as a rotating item. Both Maximo Asset Management and the Oracle E-Business Suite support items under serial control. In Oracle, the serial attributes must be configured at the master organization level, not the organization level.

Maximo Asset Management creates an ASSET record for each rotating item, using the serial number that is created in Oracle. Maximo Asset Management does not create an asset number.

## Outbound receipt processing

If you do not enter a bin number on a receipt, the integration framework uses the ITMSUBXREF integration control to determine a default subinventory code for the transaction.

The Oracle E-Business Suite import process does not recognize serial numbers or inspection transactions. Partial inspections of receipts in Maximo Asset Management are sent to the Oracle E-Business Suite as partial receipts. The Oracle E-Business Suite assumes that receipts are receipts to the final destination of the item.

In Maximo Asset Management, you can void a  transaction that was applied to a purchase order in error. A void is a reversal that brings the receipt back to its former state. A voided receipt is similar to a return, with the exception that a voided receipt cannot be considered for use on a credit invoice.

In Oracle E-Business Suite , you can also “correct” the receiving transactions. The Receipt interface maps void receipts that are void to receipt corrections in Oracle E-Business Suite.  Receipt corrections from Oracle are transferred as return transactions to asset management. A receipt correction from Oracle E-Business Suite has different business rules from the void receipts function in asset management and is mapped to the asset management return feature. For details on the rules for voiding receipt transactions in asset management, see the Maximo Asset Management Information Center.

## Service procurement line receipt

When Oracle Services Procurement is implemented, the Oracle E-Business Suite provides two additional line types: fixed price and rate value base. These line types are sent to Maximo Asset Management as service line types on an inbound purchase document.

When a fixed price line type is received in Maximo Asset Management, these lines are sent to the Oracle E-Business Suite. However, rate line types cannot be received in Maximo Asset Management since the receiving Oracle open interface does not support the import of rate-based line types.
Rate-based line types must be received in the Oracle E-Business Suite according to the process that is defined by receiving rate-based purchase order line.

## Inbound receipt processing

The inbound integration transfers receipts to Maximo Asset Management after inspection and serialization are complete and the item has reached its final destination. The inbound integration uses one of two enterprise services,

depending whether the receipt is for a service, a rotating item, or a non-rotating item.

| Line type | Rotating item | Enterprise service |
|---|---|---|
| Item | Yes | MXRCVROTITM |
| Item | No | MXRECEIPT |
| Service | N/A | MXRECEIPT |

If an item is under serial control, the integration creates a single receipt transaction for all the serial numbers. If an item is under lot control and the receipt is for multiple lots, the integration passes only the first lot number. Subsequent inventory balance transactions pass the other lot numbers.

Quantities and units of measure on inbound receipts reflect the primary quantity and primary unit of measure for the corresponding item. Conversion before the integration ensures accurate quantities in case complex conversion processes that exist in the Oracle E-Business Suite do not exist in Maximo Asset Management.

When such a conversion takes place, you can see different quantity and unit costs, but the same line costs, in the Oracle E-Business Suite and Maximo Asset Management receipts.

## Return of assets to vendors

You can return rotating assets and direct-issue items to vendors. For example, you can return an asset if it has a defect. A returned asset is removed from the inventory of assets that can be used. Maximo Asset Management applies the following rules to the return of rotating assets and items to vendors:

- The asset must be located in the same storeroom where the asset was received.
- The inventory cost for the asset must be equal to the cost of the line being reversed.
- A direct-issue item cannot be returned if it was moved.

## Filter

The integration framework transfers receipts from Maximo Asset Management to the Oracle E-Business Suite when the receipt is complete.

The integration framework transfers receipts from the Oracle E-Business Suite to Maximo Asset Management when the following conditions are met:

- The value of the Oracle integration control RCVIN is 1.
- The quantity on the receipts is greater than 0.
- The transaction type is Deliver, Reject, or Return to Receiving.
- The receipt is complete.
- The source document is a purchase order.

# Integration controls

The receiving integration uses the following Maximo Asset Management and Oracle integration controls.

| Control | Description | System |
|---|---|---|
| GENITEM | Dummy identifier for masking item number | Maximo Asset Management |
| GENSTORE | Dummy identifier for masking storeroom in outbound transactions | Maximo Asset Management |
| GENUSR | Dummy identifier for masking user ID in outbound transactions | Maximo Asset Management |
| ITMSUBXREF | Cross-reference the Oracle E-Business Suite inventory organization and default subinventory code | Maximo Asset Management |
| OACATXREF | Cross-reference Maximo Asset Management line type and the Oracle E-Business Suite item or service category identifier | Maximo Asset Management |
| ORGXREF | Cross-reference Maximo Asset Management organization and the Oracle E-Business Suite ledger ID | Maximo Asset Management |
| RCVIN | Send the Oracle E-Business Suite receipt data to Maximo Asset Management | Oracle E-Business Suite |
| SITEXREF | Cross-reference Maximo Asset Management site ID and the Oracle E-Business Suite operating unit | Maximo Asset Management |
| SUBLOCDEL | Delimiter between subinventory code and locator ID in Maximo Asset Management bin number | Maximo Asset Management |

# Invoice integration

The invoice integration transfers invoices from Maximo Asset Management to the Oracle E-Business Suite when the status of an invoice matches a status in the APSEND integration control.

The integration framework transfers validated invoices from the Oracle E-Business Suite to Maximo Asset Management when a receipt is complete and its items sent to their final destination. It also transfers invoice updates to Maximo Asset Management when an invoice is paid in the Oracle E-Business Suite.

In Maximo Asset Management, when you create an invoice, you can specify a date for the general ledger financial period. If your system administrator specified in the Organizations application that the invoice date be used, the **G/L Posting Date** field is populated with the date from the **Invoice Date** field. If the value specified is not valid, the value is updated to the start date of the next open financial period. You can specify a different value.

## Centralized invoices

Within Maximo Asset Management, you can combine payments for purchase orders from different sites within the same organization on a single invoice. However, this consolidation is not available in Oracle E-Business Suite. The invoice integration both in and outbound does not support this functionality.

## Structure of invoice line numbers

The invoice line number for an inbound invoice is a concatenation of the number 1, a four-digit Oracle invoice line number, and a four-digit distribution line number.

## Vendor pay sites

Both Maximo Asset Management and Oracle invoices can contain a vendor that differs from the vendor on the corresponding purchase order. In the Oracle E-Business Suite, you must designate a vendor site as a Pay Site if it is used on an invoice.

## Invoice matching

Maximo Asset Management supports four-way matching of an invoice when the corresponding purchase order contains rotating items or items that require

inspection. It supports three-way matching if the items are non-rotating and do not require inspection.

| Match approval level | Description |
|---|---|
| 3-way | Checks that the purchase order and invoice match within the defined tolerances:<br>• The quantity billed is less than or equal to the quantity ordered on the PO shipment.<br>• The invoice price on the PO shipment is less than or equal to the purchase order price on the PO shipment.<br>Checks that the receipt and invoice information match the quantity tolerances defined:<br>• The quantity billed on the PO shipment is less than or equal to the quantity received on the PO shipment. |
| 4-way | Same as 3-way, with the following additional criterion:<br>Checks that acceptance documents and invoice information match within the quantity tolerances defined:<br>• The quantity billed on the purchase order shipment is less than or equal to the quantity received on the purchase order shipment. |

# Reversal of invoices

In Maximo Asset Management, when an original invoice contains errors, such as unit cost mistakes, you can replace the original invoice by reversing it. In Oracle E-Business Suite, you can cancel a complete invoice. You can also cancel invoice distributions.

The status of the invoice, the credit memo, or the debit memo that is being reversed must be Approved or Paid.

You must cancel the invoice in the application in which it was created, if the invoice was created in Oracle E-Business Suite, it must be canceled in Oracle E-Business Suite.

### Inbound processing of canceled invoices

In Oracle E-Business Suite, you can cancel an invoice at the header level, this cancels the complete invoice. You can also reverse an invoice distribution, this does not cancel the complete invoice.

You can cancel a paid invoice in Oracle E-Business Suite only after the payment is voided, this is transferred to Maximo Asset Management as a reverse invoice.

In Oracle E-Business Suite, if you cancel an invoice distribution, it is transferred to Maximo Asset Management as a reverse invoice, or a reverse credit invoice, or a reverse debit invoice. Canceled invoices of type STANDARD are transfered from Oracle E-Business Suite to Maximo Asset Management as invoices of type

REVINVOICE. Canceled invoices of type CREDIT are transfered from Oracle E-Business Suite to Maximo Asset Management as invoices of type REVCREDIT. Canceled invoices of type DEBIT are transfered from Oracle E-Business Suite to Maximo Asset Management as invoices of type REVDEBIT.

Once the invoice is canceled in Oracle E-Business Suite, the accounting must be recreated to send the canceled invoice back to Maximo Asset Management.

**Outbound processing of reversed invoices**

In Maximo Asset Management, when an original invoice contains errors, such as unit cost mistakes, you can replace the original invoice by reversing it. The replacement for the original invoice contains the updated information. You can reverse an invoice without creating a return transaction.

You can transfer reversed invoices from Maximo Asset Management to Oracle E-Business Suite as a credit invoice.

Payment updates to invoices that are reversed in Maximo Asset Management are not imported back from Oracle E-Business Suite to Maximo Asset Management.

For more information, see the Invoice topics in the Maximo Asset Management Information Center.

# Filter

The integration framework transfers invoices from Maximo Asset Management to the Oracle E-Business Suite when the status of the invoice matches a status in the APSEND integration control.

The integration framework transfers invoices from the Oracle E-Business Suite to Maximo Asset Management when the following conditions are met:

- The value of the Oracle integration control APIN is 1.
- The invoice is validated, the accounting is created, and the receipt is complete.

# Integration controls

The invoice integration uses the following Maximo Asset Management and Oracle integration controls.

| Control | Description | System |
|---|---|---|
| APIN | Send the Oracle E-Business Suite invoice data to Maximo Asset Management | Oracle E-Business Suite |
| APLINETYPE | Default invoice line type in the Oracle E-Business Suite | Maximo Asset Management |
| APSEND | Statuses at which invoices are sent from Maximo Asset Management to the Oracle E-Business Suite | Maximo Asset Management |

| Control | Description | System |
|---|---|---|
| GENITEM | Dummy identifier for masking the item number | Maximo Asset Management |
| GENUSR | Dummy identifier for masking user ID in outbound transactions | Maximo Asset Management |
| GLCURNCY | Default GL currency code in Oracle E-Business Suite | Maximo Asset Management |
| GLCURNCYTYPE | Type of currency exchange rates | Maximo Asset Management |
| INVCSTATUSXREF | Translate the Maximo Asset Management internal and external invoice status, if using synonyms | Maximo Asset Management |
| INVCTYPEXREF | Cross-reference Maximo Asset Management and the Oracle E-Business invoice document types | Maximo Asset Management |
| ORGXREF | Cross-reference Maximo Asset Management organization and the Oracle E-Business ledger ID | Maximo Asset Management |
| SITEXREF | Cross-reference Maximo Asset Management site ID and the Oracle E-Business operating unit | Maximo Asset Management |

# Item and inventory integration

# 12

Maximo Asset Management and Oracle E-Business Suite can exchange the following types of item and inventory data:

| Inventory data type | Processing direction |
|---|---|
| Item | Outbound, Inbound |
| Inventory | Outbound, Inbound |
| Inventory balance | Inbound |
| Material reservation | Outbound |
| Material issue | Inbound |

There are factors that you must be aware of when integrating Maximo Asset Management and Oracle E-Business Suite item and inventory activity. Some factors include the Oracle E-Business Suite item and inventory processing considerations, item, inventory, inventory balance, material reservation, and material issue integration.

# Item and inventory processing considerations

The following components apply to item and inventory transactions.

## Ownership of items and inventory

The integration framework defines an Oracle-owned item as one that is created and managed in the Oracle s E-Business Suite, and a Maximo Asset Management-owned item as one that is created and managed in Maximo Asset Management. The integration framework similarly distinguishes between Oracle-owned and Maximo Asset Management-owned inventory.

You must change the definition of an item in the system that owns the item. If you create an item in the Oracle E-Business Suite, then transfer it to the Maximo Asset Management and want to manage it in Maximo Asset Management, you must first change the OWNERSYSID of the item and inventory to null by using a SQL statement.

## Multiple item segments

Oracle E-Business Suite users can create part numbers with multiple segments (for example, ABC-100), with intelligence built into the numbers. Maximo Asset Management users cannot.

The inbound integration concatenates the segments of the Oracle item number, inserts the delimiter specified in the Oracle E-Business Suite, and passes a single item number field to Maximo Asset Management.

Ensure that the ITEMNUM field in Maximo Asset Management is large enough to hold the concatenated item segments and delimiters.

## Material management transactions

Material management transactions, for example, item creation and item transactions, cannot cross sites. They must be associated with the same Maximo Asset Management site and the Oracle E-Business Suite operating unit.

# Item integration

The item publish channel transfers new and updated items from Maximo Asset Management to the Oracle E-Business Suite. The item enterprise service transfers new and updated items from the Oracle E-Business Suite to Maximo Asset Management.

## Structure of items and organizations

Before using the inbound item enterprise service, you must determine how to map Maximo Asset Management organizations to Oracle ledgers. You then must create and associate Maximo Asset Management item sets with the Maximo Asset Management organizations in the Organizations application. The inbound integration determines the item set from the ledger that is associated with an item.

## Lotted items

The integration framework supports lotted items in the following circumstances:

- Inventory costs and balances are managed in the Oracle E-Business Suite.
- Purchase orders and invoices are created in the Oracle E-Business Suite.

## Attributes of Oracle-owned items

The Oracle-owned items have the following attributes. These attributes apply to items that are created in Maximo Asset Management by the inbound integration framework processing.

**Costs**

The integration framework supports Average, Standard, LIFO, and FIFO cost methods in the Oracle E-Business Suite, for setting item costs in Maximo Asset Management. The cost attributes in Oracle E-Business Suite Item Master control whether the LIFO/FIFO layer cost is created. A Maximo Asset Management item that is costed as either LIFO or FIFO has the Cost Enabled attribute and the Inventory Asset Value attribute set to Yes in Oracle E-Business Suite.

**The internal order enabled attribute**

The Internal Order Enabled attribute on an Oracle item identifies whether the item can be ordered internally in the Oracle E-Business Suite. Its value appears in the OA_IO_FLAG field in the Maximo Asset Management INVENTORY record. The outbound integration does not use this field. If necessary, you can write a user exit procedure to validate its value.

**The purchasable Attribute**

The PURCHASEABLE attribute on an Oracle item identifies whether the item can be ordered on a purchase order in the Oracle E-Business Suite. You can change its value by unhiding the OA_PURCH_FLAG in the Maximo Asset Management INVENTORY table. The outbound integration does not use this field. If necessary, you can write a user exit procedure to validate its value.

**Unit of measure**

When the integration framework creates an item in Maximo Asset Management, it uses the Oracle primary unit of measure as the Maximo Asset Management order and issue units of measure.

# Attributes of Maximo Asset Management owned items

Maximo Asset Management owned items have the following attributes. These attributes apply to items that are created in the Oracle E-Business Suite by the outbound integration framework processing.

**Inspection items**

If a Maximo Asset Management item requires inspection (Inspect on Receipt is selected), the value of the Receipt Routing field of the item created in the Oracle E-Business Suite is Inspection Required.

**Rotating items**

Maximo Asset Management refers to an item under serial control as a rotating item. Both Maximo Asset Management and Oracle E-Business Suite support items under serial control. If a Maximo Asset Management item is a rotating item, the serial control field of the item that the integration framework creates in Oracle E-Business Suite is set to predefined.

**Stock or non-stock items**

In Maximo Asset Management, stocked items are items that you stock continually because they are frequently needed. A stocked item can be added to a storeroom, they are part of the inventory. If the Reorder flag is set at inventory level, then stocked items can be reordered.

Non-stock items are direct-issue items that are not stocked on a regular basis and cannot be added to a storeroom. Non-stock items are expense items.

The category flag indicates if an item is a stocked item or a non-stocked item

# Stock and non-stock item processing

In Oracle E-Business Suite, the Stockable flag indicates if an item is stocked or non-stocked. The Stockable flag works with the Inventory Item flag. A stocked item is included in the inventory. Stocked and non-stocked items can be either inventory items or expense items, this is indicated by the Inventory Asset Value flag.

In Maximo Asset Management, stocked items are usually inventory items and are included in valuations. Non-stocked items are expense items and are excluded from the inventory valuation.

In Oracle E-Business Suite a stocked or a non-stocked item must be part of the inventory organization (Storeroom). In Maximo Asset Management, non-stocked items do not exist in a storeroom and are defined only at the item master level, the planning cost information is in the item vendor catalog (INVVENDOR).

Due to the differences between stocked and non-stocked items in the two applications, the outbound transfer of Maximo Asset Management non-stocked items to Oracle E-Business Suite is not supported.

Maximo Asset Management stocked items are transferred to Oracle E-Business Suite as stocked inventory items.

Items in Oracle E-Business Suite are transferred to Maximo Asset Management as stocked items. The Stockable flag setting in Oracle E-Business Suite controls how the Reorder flag in Maximo Asset Management is set.

| Oracle E-Business Suite | Maximo Asset Management |
|---|---|
| Stockable Flag= no | Reorder flag= no |
| Stockable flag= yes | Reorder flag= yes |

## Inbound item processing from the continuous queue

An error can occur if separate threads simultaneously process inbound transactions for the same item from the continuous queue. If two threads try to insert the same item, an error occurs in the console. This is a temporary error and can be ignored. When the integration framework reprocesses the transaction in error, it finds the existing item record, and performs an update instead of an insert.

## Outbound lot item mapping

If you want to pass lot items from Maximo Asset Management to Oracle E-Business Suite, you must map a lot type to an Oracle open interface table. You also must populate the Starting Prefix and Starting Number fields in the Oracle open interface table for System Items, when you use the Full Control value as the lot control code.

To map a lot prefix and auto lot number, you can add the ITEM.OA_LOT_PREFIX and ITEM.OA_LOT_NUMBER columns in the Item application in Maximo Asset Management by using the Application Designer application. You also must populate the lot prefix and lot start numbers. For more information about adding fields to a Maximo Asset Management application, see the *IBM Maximo Asset Management Application Developer Guide*.

Alternatively, you can use the following integration controls for the lot item mappings:

| IFACECONTROL | Control type | Default value | Description |
|---|---|---|---|
| OAITMLOTPREFIX | Value | NULL | Lot prefix for a lot controlled item |
| OAITMLOTSTARTNUM | Value | NULL | Lot start number for a lot controlled item |

## Effectivity of items

The values in the Enabled, From and To fields in the OracleE-Business Suite Segment Values window indicate whether an item segment is active or inactive. If the Enabled check box is selected and the system date falls between the From and To dates, the item segment is active.

The integration framework uses these values to set the EXT_ACTIVE flag in Maximo Asset Management, MXITEM_IFACE table to 0 (inactive) or 1 (active), but it does not use the field to validate the status of the item in outbound transactions. If necessary, you can create a processing rule or write a user exit procedure to check the value in this field, and use the Application Designer application to unhide and view the value of the field.

The item integration does not update the Maximo Asset Management items when the status of an Oracle item segment changes to active or inactive based on the system date. To ensure that Maximo Asset Management uses only active items, you must run the SYNCITM synchronization script frequently.

## Filter

The integration framework transfers items from the Oracle E-Business Suite to Maximo Asset Management when the value of the Oracle integration control ITEMIN is 1.

## Integration controls

The item integration uses the following Maximo Asset Management and Oracle integration controls.

| Control | Description | System |
|---|---|---|
| CATEGORYXREF | Cross-reference the Maximo Asset Management item category and the Oracle E-Business Suite stock-enabled flag | Maximo Asset Management |
| ITEMIN | Send the Oracle E-Business Suite item data to Maximo Asset Management | Oracle E-Business Suite |
| LANGXREF | Cross-reference the Maximo Asset Management language code and the Oracle E-Business Suite process set ID | Maximo Asset Management |
| LOTTYPEXREF | Cross-reference the Maximo Asset Management and the Oracle E-Business Suite lot control indicators | Maximo Asset Management |
| ORGXREF | Cross-reference the Maximo Asset Management organization and Oracle ledger ID | Maximo Asset Management |
| OAITMLOTPREFIX | Lot prefix for lot controlled item | Maximo Asset Management |
| OAITMLOTSTARTNUM | Lot start number  for lot controlled item | Maximo Asset Management |

The LOTTYPEXREF integration control cross-references the Maximo Asset Management and the Oracle E-Business Suite lot control values as follows:

| Maximo Asset Management LOTTYPE values | Oracle lot control value |
|---|---|
| NOLOT | 1 (No Control) |
| LOT | 2 (Full Control) |

## Multilingual support

For information about multilingual support for items, see the multiple language support information.

# Inventory integration

The inventory integration transfers new and updated inventory records from Maximo Asset Management to Oracle E-Business Suite and from the Oracle E-Business Suite to Maximo Asset Management.

## Costing method

In Maximo Asset Management, the costing method can be defined at the storeroom level. Different items in the storeroom can have different cost types. The supported costing methods are Average, Standard, FIFO, LIFO.

In Oracle E-Business Suite , the costing method is defined at the inventory organization level. All items under one inventory organization have the same costing method. Therefore, the inventory integration transfers the costing method linked to the inventory organization to Maximo Asset Management.

The inventory integration framework transfers the costing method information from Oracle E-Business Suite into Maximo Asset Management.

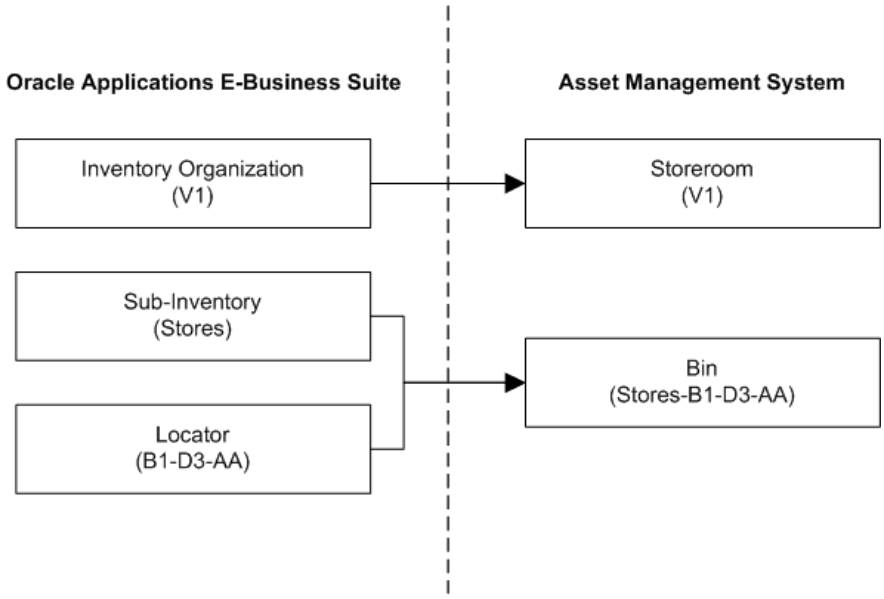
If you want to transfer costing method information from Maximo Asset Management to Oracle E-Business Suite, you can have only one costing method for each storeroom. Therefore, all items within a storeroom share the same costing method.

## Inventory mapping

Inventory is the combination of an item and a storeroom location. A storeroom in Maximo Asset Management is equivalent to an inventory organization in the Oracle E-Business Suite.

The Oracle E-Business Suite inventory organization equates to the Maximo Asset Management storeroom, and the concatenation of subinventory and locator equate to the Maximo Asset Management bin.

The following diagram shows the mapping of the Oracle inventory data to Maximo Asset Management:




If the inbound integration transfers inventory data to a storeroom that does not exist in Maximo Asset Management, an error occurs.

## Default bins

The ITMSUBXREF integration control contains a default subinventory code for each Oracle E-Business Suite inventory organization. The integration can use this value to determine the Oracle subinventory code and derive the Maximo Asset Management bin numbers.

## Filter

The integration framework transfers inventory from the Oracle E-Business Suite to Maximo Asset Management when the value of the Oracle integration control ITEMIN is 1.

# Integration controls

The inventory integration uses the following Maximo Asset Management and Oracle integration controls.

| Control | Description | System |
|---|---|---|
| CATEGORYXREF | Cross-reference the Maximo Asset Management item category and the Oracle E-Business Suite stock-enabled flag | Maximo Asset Management |
| GENSTORE | Dummy identifier for masking storeroom in outbound transactions | Maximo Asset Management |
| ITEMIN | Send the Oracle E-Business Suite item data to Maximo Asset Management | Oracle E-Business Suite |
| LANGXREF | Cross-reference the Maximo Asset Management language code and the Oracle E-Business Suite process set ID | Maximo Asset Management |
| ORGXREF | Cross-reference the Maximo Asset Management organization and the Oracle E-Business Suite ledger ID | Maximo Asset Management |
| SITEXREF | Cross-reference the Maximo Asset Management site ID and the Oracle E-Business Suite operating unit | Maximo Asset Management |

The CATEGORYXREF integration control cross-references the Maximo Asset Management item category and the Oracle item Stockable attribute, as follows:

| Maximo Asset Management item category | Oracle stockable attribute |
|---|---|
| NS | N |
| STK | Y |

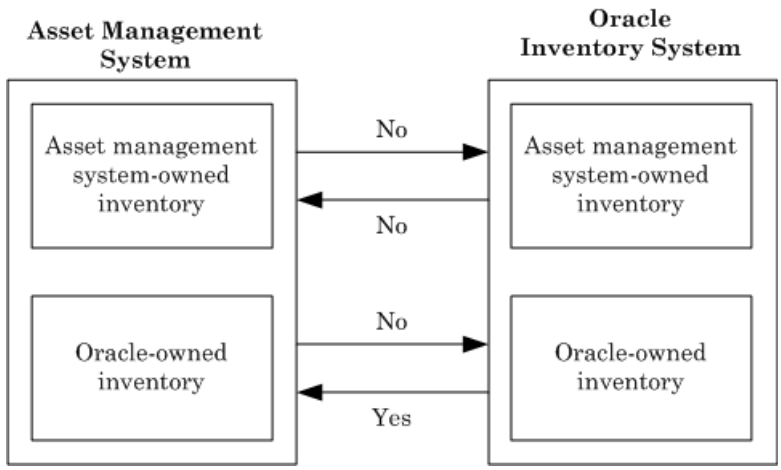
# Inventory balance integration

The inventory balance enterprise service transfers inventory balance updates, including physical inventory adjustments and LIFO/ FIFO cost information from the Oracle E-Business Suite to Maximo Asset Management. It does not transfer balance updates for rotating items.

## Updates to inventory balances

The adapter transfers inventory balance and cost updates from the Oracle E-Business Suite to Maximo Asset Management when the inventory is owned by Oracle E-Business Suite and the costing process in Oracle E-Business Suite is complete.

The following diagram shows the allowable inventory balance updates:




The adapter synchronizes inventory balance and cost information between the two applications by refreshing the Maximo Asset Management inventory balance and cost data with values transferred from the Oracle E-Business Suite.

**ATTENTION** If you update costs in the Oracle E-Business Suite but do not send the transaction information to Maximo Asset Management, a discrepancy might exist between the costs in the two systems.

The following restrictions apply to inventory balance and cost updates:

- You cannot use Maximo Asset Management to perform inventory balance and cost adjustments on Oracle-owned inventory.
- You cannot transfer inventory balance and cost adjustments, for Maximo Asset Management-owned inventory, from the Oracle E-Business Suite to Maximo Asset Management.

The current balance on the inbound transaction is the total of the on-hand quantities for the combination of item, inventory organization, subinventory, locator (optional), and lot number (optional). The first three levels must exist in order for the integration framework to pass an inventory balance.

Standard, average, and last cost values are transferred in the inventory integration, not the inventory balance integration.

LIFO/ FIFO cost information is transferred through the inventory balance integration from Oracle E-Business Suite to Maximo Asset Management

## Filter

The integration framework transfers inventory balances from the Oracle E-Business Suite to Maximo Asset Management when the following conditions are met:

- The value of the Oracle integration control INVBALIN is 1.
- The item is not a rotating item.
- The item is Oracle-owned.

## Integration controls

The inventory balance integration uses the following Maximo Asset Management and Oracle integration controls.

| Control | Description | System |
|---|---|---|
| INVBALIN | Send the Oracle E-Business Suite inventory balance data to Maximo Asset Management | Oracle E-Business Suite |
| ORGXREF | Cross-reference the Maximo Asset Management organization and the Oracle E-Business Suite ledger ID | Maximo Asset Management |
| SITEXREF | Cross-reference the Maximo Asset Management site ID and the Oracle E-Business Suite operating unit | Maximo Asset Management |
| SUBLOCDEL | Delimiter between subinventory code and locator ID in the Maximo Asset Management bin number | Maximo Asset Management |

# Material reservation integration

The material reservation publish channel transfers new item reservations from Maximo Asset Management to the Oracle E-Business Suite.

In order that the reservation can be transfered from Maximo Asset Management to Oracle E-Business Suite, the reservation request number must be numeric

Maximo Asset Management supports several reservation types, hard reservations, soft reservations, and backorder reservations. When you create a reservation request, you assign the reservation type.

You can set a flag to assign the reservation type automatically. If this flag is set, Maximo Asset Management assigns the reservation types of APSOFT, APHARD, or Backorder and uses the procurement lead time and the required date to determine the reservation type.

In Oracle E-Business Suite, when an inventory reservation is created, it is a guaranteed allocation of material. Therefore, new and updated Maximo Asset Management reservations of types HARD and APHARD are transferred to Oracle E-Business Suite. Deleted Maximo Asset Management reservations are also transfered to Oracle E-Business Suite.

## Default bins

The ITMSUBXREF integration control contains a default subinventory code for each Oracle E-Business Suite inventory organization. The integration framework can use this value to determine the Oracle subinventory code and derive the Maximo Asset Management bin numbers.

## Integration controls

The reservations integration uses the following Maximo Asset Management integration controls.

| Control | Description | System |
|---|---|---|
| GENSTORE | Dummy identifier for masking storeroom in outbound transactions | Maximo Asset Management |
| GENUSR | Dummy identifier for masking user ID in outbound transactions | Maximo Asset Management |
| ITMSUBXREF | Cross-reference the Oracle E-Business Suite inventory organization and default subinventory code | Maximo Asset Management |
| ORGXREF | Cross-reference the Maximo Asset Management organization and the Oracle s E-Business Suite ledger ID | Maximo Asset Management |
| SITEXREF | Cross-reference the Maximo Asset Management site ID and the Oracle E-Business Suite operating unit | Maximo Asset Management |
| SUBLOCDEL | Delimiter between subinventory code and locator ID in the Maximo Asset Management bin number | Maximo Asset Management |

# Material issue integration

The material issue enterprise services transfers material issue transactions from the Oracle E-Business Suite to Maximo Asset Management.

Material issue transfers are only supported from Oracle E-Business Suite to Maximo Asset Management. Material reservation is supported from Maximo Asset Management to Oracle E-Business Suite.

## Creation of material issues

When creating a material issue in the Oracle E-Business Suite, enter the following information in the Demand Source field:

| Destination | Demand source value |
|---|---|
| Work order | WO:`work order number` |
| Reservation | MISC:`Asset management system request number` |

## Rotating items

The following considerations apply to the issue and return of rotating items.

Oracle users can assign multiple serial numbers on a single material issue transaction, while Maximo Asset Management users can enter only one issue transaction per rotating item. If a material issue or return that is created in the Oracle E-Business Suite contains a quantity that is greater than one and multiple serial numbers, the integration framework creates an inbound issue transaction with a quantity of one for each serial number.

If you create an Oracle Miscellaneous Receipt to return an item under serial control to the storeroom, you must assign the item a serial number that has already been issued out of the storeroom. Otherwise an error occurs in Maximo Asset Management since the serial number does not exist as an asset number in Maximo Asset Management.

Maximo Asset Management requires a location when issuing a rotating item. You can create a reservation against a rotating item in Maximo Asset Management without referencing a work order. However, if you create the corresponding issue transaction in the Oracle E-Business Suite, an error occurs when the integration record does not contain a location, since Maximo Asset Management requires a location. You can write a user exit procedure or use processing rules to assign a default location.

## Filter

The integration framework transfers material issues from the Oracle E-Business Suite when the following conditions are met:

- The value of the Oracle integration control ISUIN is 1.
- The issue is a material issue or a receipt.

# Integration controls

The material issue integration uses the following Maximo Asset Management and Oracle integration controls.

| Control | Description | System |
|---|---|---|
| GENSTORE | Dummy identifier for masking storeroom in outbound transactions | Maximo Asset Management |
| GENUSR | Dummy identifier for masking user ID in outbound transactions | Maximo Asset Management |
| ISUIN | Send the Oracle E-Business Suite material issue data to Maximo Asset Management | Oracle E-Business Suite |
| SITEXREF | Cross-reference the Maximo Asset Management site ID and the Oracle E-Business Suite operating unit | Maximo Asset Management |
| SUBLOCDEL | Delimiter between subinventory code and locator ID in the Maximo Asset Management bin number | Maximo Asset Management |

# Oracle Project accounting integration 13

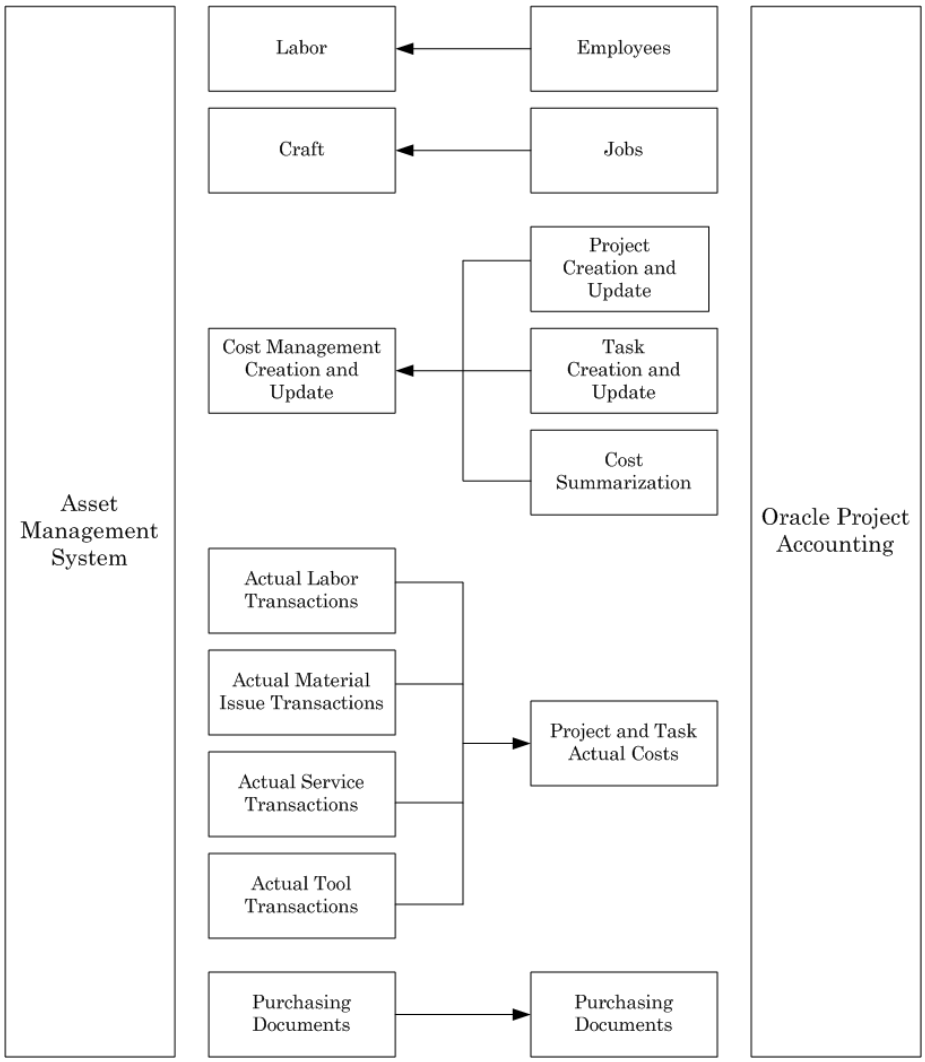
When you integrate Maximo Asset Management and Oracle Project Accounting, you can update and track project costs in both systems. The two systems can exchange the following types of data.

| Data type | Processing direction |
|---|---|
| Project and Task | Inbound |
| Actual Costs | Outbound |

There are factors that you must be aware of when you integrate Maximo Asset Management and the Oracle Project Accounting activity. Some factors include object structures, integration framework processing, labor, craft, and project integration, integration control use, and mapping capabilities.

# Maximo Asset Management and Oracle Project Accounting integration

The following diagram shows how Maximo Asset Management and Oracle Project Accounting exchange data using the Projects integration. The arrows point to the destination system.



# Performing projects integration prerequisites

To use the Projects integration in Maximo Asset Management:

1. Perform the labor and craft integration.
2. Define the transaction sources in the Oracle Project Accounting.
3. Depending on the value of the RESLEVELITEM and RESLEVELTOOL integration controls, define expenditure types in Oracle Project Accounting.
4. Depending on the value of the RESLEVELLABOR integration control, define labor or craft codes as employees.
5. Add the fields to the user interfaces in the following applications:
    - Cost Management
    - Invoice
    - Purchase Order
    - Purchase Requisition
    - Receiving
    - Work Order Tracking
6. Using the System Administrator Responsibility, define the operating unit mode for the Oracle Project Accounting concurrent jobs.
    a) Go to **Concurrent>Program>Define**.
    b) Select the Project Account concurrent job.
    c) Select the **Update** icon.
    d) In the Request tab, enter a value in the Operating Unit Mode field.
    e) Save your changes.

    In the Oracle E-Business Suite, you must specify a mode for the operating unit in the concurrent request set for some of the Projects concurrent jobs.

# Processing overview for projects integration

The are activities that are associated with the Projects integration. The Projects integration involves the following activities:

1. In Oracle Project Accounting application:
    a Create or update a work breakdown structure.
    b) If necessary, assign budget amounts to the project.
    c) Run the following process:

PRC: Update Project Summary Amounts

**d)** Run the following process to extract and pass the work breakdown structure and budget data to the Maximo Asset Management Cost Management application:

PRC: Maximo Asset Management Financial Control Update from Oracle

**2.** In the Maximo Asset Management Work Order Tracking application:

**a)** Create a work order.

**b)** Assign the work order to a project and lowest-level task.

**c)** Approve the work order.

**d)** Record labor, material, service receipt, and tool (actuals) transactions against the work order.

**3.** In the Oracle Project Accounting application, run the following process to import the actuals transactions into Oracle Projects:

PRC: Transaction Import

**4.** In the Oracle Project Accounting application:

**a)** Run the following processes to distribute and summarize project costs:

- PRC: Distribute Labor Costs
- PRC: Distribute Usage and Miscellaneous Costs
- PRC: Update Project Summary Amounts

**b)** Run the following process to extract and pass work breakdown structure, budget, and summarized cost data to the Maximo Asset Management Cost Management application:

PRC: Financial Control Update from Oracle

# Labor and craft integration

Labor and craft integration is a prerequisite to the Projects integration. The projects integration uses this information to pass the labor and craft rates that are defined in Oracle Projects to the Maximo Asset Management LABORCRAFT records. These rates are used in the work order actuals.

The Projects integration uses the labor and craft components that are provided within the Oracle adapter. There are no separate labor and craft integration components for the Projects integration.

# The PROJPAY integration control

The labor and craft integration can retrieve pay rates from the Oracle Human Resources module or cost rates from the Oracle Project Accounting module. The value of the PROJPAY integration control indicates the module to be used.

| PROJPAY value | Cost rate source |
|---|---|
| 0 | Oracle Human Resources |
| 1 | Oracle Project Accounting |

The default value of PROJPAY is 1 when the Projects integration is installed.

# Cost rates in Oracle Project accounting

In Oracle Project Accounting application, an employee can be assigned a fixed rate for all projects or defined in a rate schedule.

A rate schedule is a list of cost rates for cost projects by either employee or job. When a rate schedule lists cost rates per job, the integration framework uses the cost rate for the job that is associated with the employee primary assignment in Oracle Human Resources. Rate schedules can be shared by projects that are assigned to different operating units.

# Factors in retrieving cost rate

The following factors determine how the labor and craft integrations retrieve cost data.

| Factor | Determination |
|---|---|
| Whether using Oracle Human Resources pay rates or Oracle Project Accounting cost rates | The value of the PROJPAY control<br><br>Value 0: retrieve costs from Oracle Human Resources<br><br>Value 1: retrieve costs from Oracle Project Accounting |
| Whether the cost rate is determined by fixed rate or cost schedule | The Oracle Project Accounting override type |
| Whether Oracle Project Accounting cost rates are set by employee or by job | The Oracle Project Accounting cost schedule |

The following table shows how the integration framework uses these factors to retrieve cost rates. All rates are based on the job of the employee primary assignment.

If the value of PROJPAY is 1 and no Oracle Project Accounting cost rate exists, the integration framework skips the transaction.

| PROJPAY value | Oracle Project Accounting override type | Oracle Project Accounting costing schedule | Rate source | Labor cost rate (LABORCRAFTRATE) |
|---|---|---|---|---|
| 0 | N/A | N/A | HR Payroll | HR pay rate for the job of the primary assignment<br><br>(PER_PAY_PROPOSALS.PROPOSED_SALARY * PER_PAY_BASES.PAY_ANNUALIZATION_FACTOR) / CONVHRS |
| 1 | Rate | Employee | Oracle Project Accounting Labor Costing | Hourly cost rate from the Oracle Project Accounting labor costing table<br><br>PA_COMPENSATION_DETAILS_ALL.HOURLY_COST_RATE |
| 1 | Schedule | Employee | Oracle Project Accounting Rate Schedule | Cost rate per employee from the Oracle Project Accounting rate schedule table<br><br>PA_BILL_RATES_ALL.RATE where PERSON_ID = PA_COMPENSATION_DETAILS_ALL.PERSON_ID and BILL_RATE_SCH_ID = PA_COMPENSATION_DETAILS_ALL.RATE_SCHEDULE_ID and ORG_ID = PA_COMPENSATION_DETAILS_ALL.ORG_ID and SYSDATE BETWEEN START_DATE_ACTIVE AND END_DATE_ACTIVE |
| 1 | Schedule | Job | Oracle Project Accounting Rate Schedule | Cost rate per job from the Oracle Project Accounting rate schedule table<br><br>PA_BILL_RATES_ALL.RATE where BILL_RATE_SCH_ID = PA_COMPENSATION_DETAILS_ALL.RATE_SCHEDULE_ID and JOB_ID = PER_ALL_ASSIGNMENTS_F.JOB_ID and ORG_ID = PA_COMPENSATION_DETAILS_ALL.ORG_ID and SYSDATE BETWEEN START_DATE_ACTIVE AND END_DATE_ACTIVE |

## Multiplication of cost rates

In Oracle Projects, you can configure cost rates for an employee by operating unit. In Maximo Asset Management, cost rates and the crafts are stored at the organization level.

When the value of PROJPAY is 1 and cost rate records exist for the same employee in multiple Oracle operating units, the integration framework passes all the records with a valid date range to Maximo Asset Management.

To filter out unnecessary records, customize the user exit processing or create a processing rule. To include all the rates, customize the user exit processing or create a processing rule to concatenate the operating unit (site) and craft code to create a unique identifier for each cost rate.

For example, to include cost rates for organization 204 only, add the following logic to oflpa.usp:

```
If PACOMPREC.ORG_ID NOT IN (204) Then
        SKIP_TRANSACTION := True;
End If;
```

An employee Human Resources pay rate is multiplied across all Maximo Asset Management organizations. The project rate is specific to a Maximo Asset Management site.

## Labor rates without a craft

When you use the Oracle E-Business Suite, you can associate an assignment with an employee for whom a rate, but no job, exists. Maximo Asset Management requires both labor (employee) and craft (job).

You can customize the user exit processing to supply a dummy craft code for the Maximo Asset Management LABORCRAFTRATE record; otherwise, the integration framework skips the record.

## Integration controls for labor and craft integration

The labor and craft integration use the following Projects integration-specific Oracle integration control.

| Control | Description |
|---|---|
| PROJPAY | Source of cost rates |

# Project and task integration

The project and task integration transfers work breakdown structures and summarized project and task costs from the Oracle Project Accounting to the Maximo Asset Management Cost Management application.

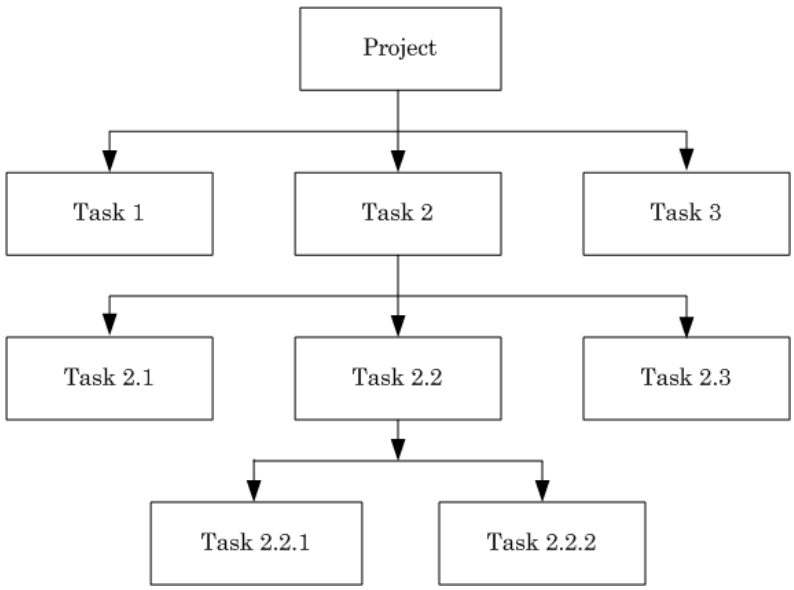
## The work breakdown structure

A work breakdown structure (WBS) is a hierarchy that consists of a top-level project and top-level, mid-level, and lowest-level tasks. A lowest-level task is a task that does not have any child tasks.

You can create work breakdown structures in the Oracle Project Accounting. Creating the work breakdown structure includes the following activities:

- Defining a project and related tasks
- Assigning budget information to the project
- Assigning labor, material, and tools to the tasks

The following diagram shows the typical work breakdown structure:




For more information about creating work breakdown structures, see the *Oracle Project Accounting User Guide.*

## Maximo Asset Management System cost management application

The Maximo Asset Management Cost Management application stores work breakdown structures and summarized project and task costs at the site level. The integration framework process assigns a unique financial control ID to each project and task that it adds to Maximo Asset Management.

# Transfer project and task Data to Maximo Asset Management

To transfer project and task data from Oracle Project Accounting to the Maximo Asset Management Cost Management application, you must first run three processes to distribute costs and generate rolled-up costs for each entry within the work breakdown structure. You then run the financial extract concurrent process (PRC: Financial Update from Oracle) to extract and transfer updated work breakdown structure and summarized cost data to Maximo Asset Management.

The latter process does not create actuals against the Maximo Asset Management work orders. Expenditures entered as actuals against a task within the Oracle Project Accounting are part of the rolled-up costs.

Depending on how you configure project types and task types in Oracle Projects, the integration framework passes burdened costs or raw costs to Maximo Asset Management.

Perform the project and task integration as often as necessary, after updating work breakdown structures or costs in Oracle Project Accounting.

| Process | Description |
|---|---|
| PRC: Distribute Labor Costs | Distributes labor costs to general ledger accounts |
| PRC: Distribute Usage and Miscellaneous Costs | Distributes non-labor costs to general ledger accounts<br><br>You can run the first two processes in any order. |
| PRC: Update Project Summary Amounts | Generates rolled-up inception-to-date budget, burdened, and committed costs for each entry in a project work breakdown structure |

| Process | Description |
|---|---|
| PRC: Maximo Asset Management Financial Update from Oracle | Transfers updated work breakdown structure and summarized cost data to Maximo Asset Management<br><br>You can enter the following optional parameters to filter projects:<br><br>• Operating Unit ID<br>• Project Number<br>• Budget Type<br>• Start Date<br><br>The current operating unit is the default operating unit ID.<br><br>If you enter a budget type, the process passes all budget amounts for that budget type to Maximo Asset Management.<br><br>If you do not enter a budget type and the project has one budget type, the process passes the budget amount for that type to Maximo Asset Management. If the project has multiple budget types, the process sets the budget amount in Maximo Asset Management to null.<br><br>The process extracts records updated on or after the start date. If you do not enter a start date, the screen displays the date in the Oracle FCEXTRACTDATE control. If you erase that value, the process extracts all projects that meet the project number and operating unit ID criteria. |

# Project and task information updates

Add and modify projects and tasks in Oracle Project Accounting, not in Maximo Asset Management. If you delete a task in Oracle Project Accounting, manually delete it in Maximo Asset Management, as the integration framework does not delete project information. For more information about deleting a task, refer to the online help for the Cost Management application.

# Project and task integration controls

The project and task integration uses the following Projects integration-specific Maximo Asset Management and Oracle integration controls.

| Control | Description | System |
|---|---|---|
| FCEXTRACTDATE | Date financial extract program (PRC: Maximo Asset Management Financial Update from Oracle) last run | Oracle E-Business Suite |
| FCSTATUSXREF | Cross-reference Oracle project status and Maximo Asset Management financial control status | Maximo Asset Management |

# Actual costs integration

Actual costs are the costs of labor, material, and tools that are associated with a work order. Whenever a Project-related general ledger transaction is saved in Maximo Asset Management, the integration framework generates a project actuals transaction. The integration framework identifies the associated project and task from the corresponding work order and includes that information in the outbound transaction.

The following types of transactions can generate actuals data:

| Transaction description | Transaction type |
|---|---|
| Labor transactions | LABTRANS |
| Material issues and returns | MATUSETRANS |
| Material receipts with direct issues | MATRECTRANS |
| Service receipts with direct issues | SERVRECTRANS |
| Tool transactions | TOOLTRANS |

The integration framework sends the actuals transactions when the associated data meets the following criteria:

- The corresponding work order is associated with a project and task.
- The corresponding task is chargeable.
- The transaction type is included in the PROJSEND integration control.

## Associating projects and tasks with work orders

All outbound project-related transactions that are passed must be assigned to a project or task for costing. When transactions are entered against a work order, the integration framework uses the corresponding financial control record to retrieve and add the project and task number to the actuals transaction.

Before you can associate a task with a work order, the task must meet the following conditions:

- The status of the project must be APPR (approved).
- The end date of the project or task must be equal to or later than the current date.
- The Is Chargeable check box for the task must be selected, indicating that the task accepts charges.

To associate a work order with a project or task, complete the following steps in the Work Order Tracking application.

1. Create a work order, and enter the project and task numbers in the appropriate fields.

   For actuals reporting, associate work orders with only the lowest-level tasks (tasks without child tasks) within a work breakdown structure.

2. Approve the work order.

3. Record labor, material, and tool actuals transactions against the task.

   For more information about these activities, refer to the online help for the Work Order Tracking application.

# Project attributes on actuals transactions

For Oracle costing purposes, Project-related actuals transactions must include the following information:

- Project ID
- Task ID
- Charge organization (project expenditure organization)
- Non-labor resource organization
- Expenditure type
- Employee number

The integration framework also adds this information to Maximo Asset Management purchase requisitions, purchase orders, and invoices that are associated with a project.

## Project and task ID

Every outbound actuals transaction includes the project and task number that is associated with the work order.

## Charge organization and non-labor resource organization

The following actuals transactions include the charge organization (project expenditure organization) and non-labor resource organization:

| Transaction description | Transaction type |
|---|---|
| Material issues and returns | MATUSETRANS |
| Material receipts with direct issues | MATRECTRANS |
| Service receipts with direct issues | SERVRECTRANS |
| Tool transactions | TOOLTRANS |

The integration framework obtains the charge organization in one of the following ways:

- If you enter the charge organization directly into a Maximo Asset Management actuals transaction, the integration framework uses that value.

  To enter a charge organization directly, you must add the charge organization field to the user interface in the appropriate applications.

- If you do not enter the charge organization directly, the integration framework uses the value in the CHARGEORG integration control.

The integration framework uses the value in the NLRORG integration control as the non-labor resource organization. Any Oracle Project Accounting charge organization can be a non-labor resource organization.

### Expenditure type

Every transaction that is sent to Oracle Projects must include an expenditure type. The integration framework retrieves the project expenditure type in one of the following ways:

- If you enter an expenditure type directly into a Project-related Maximo Asset Management transaction, the integration framework uses that value.

  To enter an expenditure type directly, you must add the expenditure type field to the user interface in the appropriate applications.

- If you do not enter the expenditure type directly, the integration framework uses the values in various integration controls to determine the value.

| Transaction type | Condition | Expenditure type value |
|---|---|---|
| LABTRANS | | The value of EXPENDLABOR control |
| MATRECTRANS<br>MATUSETRANS | RESLEVELITEM = 1 | The value of EXPENDITEM control |
| MATRECTRANS<br>MATUSETRANS | RESLEVELITEM = 2 and GENITEM not null | The value of GENITEM control |
| MATRECTRANS<br>MATUSETRANS | RESLEVELITEM = 2 and GENITEM null | The value of ITEMNUM |
| SERVRECTRANS | | The value of EXPENDITEM control |
| TOOLTRANS | RESLEVELTOOL = 1 | The value of EXPENDTOOL control |
| TOOLTRANS | RESLEVELTOOL = 2 | The value of TOOLNUM |

### Employee number

The integration framework retrieves the employee number in one of the following ways:

- If the value of the RESLEVELLABOR integration control is 1, the integration framework retrieves the craft code.

- If the value of the RESLEVELLABOR integration control is 2, the integration framework retrieves the labor code.

## Transfer actuals transactions to Oracle general ledger

You can send actuals to Oracle Projects as project actuals, to the Oracle general ledger as financial journals, or to both. The following integration controls

determine how the integration framework processes project-related and non-project-related actuals.

| Control | Description | Processing |
|---|---|---|
| GLSOURCE | Sends non-project related transactions to Oracle GL | If the transaction type is listed in this control, the integration framework skips (does not send) the transaction. |
| JEPROJSEND | Sends project-related transactions to Oracle GL | If the transaction type is listed in this control, the integration framework sends the transaction to Oracle GL. |
| PROJSEND | Sends project-related transactions to Oracle Project Accounting | If the transaction type is listed in this control, the integration framework sends the transaction to the Oracle Project Accounting if the transaction also meets the following criteria:<br>• The work order is associated with a chargeable task.<br>• The work order is associated with a financial control ID. |

# Import actuals transactions into Oracle project accounting

The Projects integration writes the project actuals to the Oracle open interface table PA_TRANSACTION_INTERFACE. To import the actuals from the interface table into Oracle Project Accounting, you must run the concurrent process for the Transaction Import (PRC: Transaction Import).

When defining the transaction source, do not select the Import Burdened Amounts check box.

# Integration controls for actual cost

The actuals integration uses the following Projects integration-specific integration controls.

| Control | Description |
|---|---|
| CHARGEORG | Default charge (project expenditure) organization |
| EXPENDITEM | Oracle Project Accounting item expenditure type |
| EXPENDLABOR | Oracle Project Accounting labor expenditure type |
| EXPENDTOOL | Oracle Project Accounting tool expenditure type |
| NLRORG | Default Oracle Project Accounting non-labor resource organization |
| PROJSEND | Transaction types for which actuals transactions are sent to Oracle Project Accounting |
| RESLEVELITEM | Resource level for items |
| RESLEVELLABOR | Resource level for labor |
| RESLEVELTOOL | Resource level for tools |
| SRCTIM | Transaction source for labor actuals |
| SRCUSE | Transaction source for non-labor actuals |

# Project attributes on purchasing transactions

The Projects integration also includes the following project attributes at the line level of some outbound purchasing (PR, PO, and invoice) documents.

- Project ID
- Task ID
- Expenditure type
- Charge organization

Use the following controls to specify whether the integration framework includes project attributes on the outbound purchasing transactions. If the value of the control is 1 (true), the transaction includes the project attributes.

| Integration control | Controls |
|---|---|
| PROJAP | Invoices |
| PROJPO | Purchase orders |
| PROJPR | Purchase requisitions |

The integration framework retrieves project attributes from the work order that is referenced by the purchasing document. As with actuals transactions, you can enter the charge organization and expenditure type directly or let the integration framework use the default value in the related integration controls.

If the value of any of these controls is 1, the integration framework assumes that the corresponding publish channel is enabled and it generates an outbound transaction even when that publish channel is disabled.

The Projects integration uses the purchasing integration components provided within the Oracle adapter. There are no separate purchasing components for the Projects integration.

## Purchase integration controls

The Projects integration uses the following Projects integration-specific integration controls when processing outbound purchasing documents.

| Control | Description |
|---|---|
| CHARGEORG | Default charge (project expenditure) organization |
| EXPENDITEM | Oracle Project Accounting item expenditure type |
| PROJAP | Include project accounting data on outbound invoice lines |
| PROJPO | Include project accounting data on outbound purchase order lines |
| PROJPR | Include project accounting data on outbound purchase requisition lines |
| RESLEVELITEM | Resource level for items |

# Project-specific integration controls

The following tables list the Oracle processing control and Maximo Asset Management integration controls that are specific to the Projects integration.

## Oracle processing control

| Control | Description | Default value |
|---|---|---|
| FCEXTRACTDATE | Date financial extract program (PRC: Maximo Asset Management Financial Update from Oracle) last run | Null |
| PROJPAY | Source of cost rates<br><br>Value 0: retrieve pay rates from Oracle HR<br>Value 1: retrieve cost rates from Oracle Project Accounting | 1<br><br>(when Projects integration installed) |

## Maximo Asset Management integration controls

The table on the following page lists the Projects-specific integration controls. The default value of every control is Null. The columns within the table are:

| Column | Description |
|---|---|
| Control Name | Name of the integration control |
| Description | A brief description of the control |
| Type | Control type:<br><br>B = Boolean<br>L = List<br>V = Value<br>X = Cross-reference |
| Domain | If applicable, the name of the domain that contains the valid values for the control.<br><br>N/A = Domains are not applicable to the control<br>N = No domain |
| Org or Site | Indicates if the system-level control value can be overridden at the organization or site level<br><br>N = Neither<br>O = Organization<br>S = Site |

| Control Name | Description | Type | Domain | Org or Site |
|---|---|---|---|---|
| CHARGEORG | Default charge (project expenditure) organization | V | N | S |
| EXPENDITEM | Oracle Project Accounting item expenditure type | V | N | S |
| EXPENDLABOR | Oracle Project Accounting labor expenditure type | V | N | S |
| EXPENDTOOL | Oracle Project Accounting tool expenditure type | V | N | S |
| FCSTATUSXREF | Cross-reference Oracle project status and Maximo Asset Management financial control status | X | FCSTATUS | S |
| NLRORG | Default Oracle Project Accounting non-labor resource organization | V | N | S |
| PROJAP | Include project accounting data on outbound invoice lines | B | N/A | S |
| PROJPO | Include project accounting data on outbound PO lines | B | N/A | S |
| PROJPR | Include project accounting data on outbound PR lines | B | N/A | S |
| PROJSEND | Transaction types for which actuals transactions are sent to Oracle Project Accounting | L | GLSOURCE | S |
| RESLEVELITEM | Resource level for items | V | N | S |
| RESLEVELLABOR | Resource level for labor | V | N | S |
| RESLEVELTOOL | Resource level for tools | V | N | S |
| SRCTIM | Transaction source for labor actuals | V | N | S |
| SRCUSE | Transaction source for non-labor actuals | V | N | S |

# Data transfer specifications

For information about labor and purchasing transactions, see the Oracle Adapter Interface Specifications information.

# Integration specifications

# A

Maximo Enterprise Adapter for Oracle Applications performs processing that is specific to the integration of Maximo Asset Management with the Oracle E-Business Suite.

The components reside in the following directories under the Maximo Asset Management root directory:

| Processing class or script | Directory |
|---|---|
| Java processes | psdi\iface\app |
| Processing classes | psdi\iface\oa11i |
| Synchronization scripts | ORACLEAPIS\install |
| PL/SQL stored procedures | ORACLEAPIS\interfac |
| PL/SQL user exit procedures | ORACLEAPIS\apisuser |
| PL/SQL procedure that the concurrent jobs runs | ORACLEAPIS\concurrent |
| The log directory | ORACLEAPIS\log |

Maximo Enterprise Adapter for Oracle Applications does not install any predefined Java user exit classes. If you develop your own, they can reside in any directory.

# General ledger component

| | |
|---|---|
| Detail Description | This component allows for the inbound synchronization of general ledger component data. |
| Object Structure | MXGLCOMP |
| Interface Table | MXGLCOMP_IFACE |
| Integration Controls | COAXREF |

## Inbound Implementation Details

| | |
|---|---|
| Synchronization Script | syncglcomp.sql |
| Oracle Control | COAIN |
| Triggering Tables | • FND_FLEX_VALUES<br>• FND_FLEX_VALUES_TL |
| PL/SQL Stored Procedure | api_glcomp_sp |
| PL/SQL User Exit Procedure | MOF_USR_GLCOMP_SP |
| Enterprise Service | MXGLCOMP_FROA12 |
| Concurrent job request | <*MAXORA_schema_name*>: Maximo GL Component Master Data |
| Processing Class | GLCompInExt |
| Java Process | None |
| Filtering | Bypass if the value of the COAIN integration control is 0. |

## Outbound Implementation Details

Not applicable

# Chart of accounts component

| | |
|---|---|
| Detail Description | This component allows for the inbound synchronization of chart of accounts data. |
| Object Structure | MXCOA |
| Interface Table | MXCOA_IFACE |
| Integration Controls | COAXREF |

## Inbound Implementation Details

| | |
|---|---|
| Synchronization Script | synccoa.sql |
| Oracle Integration Control | COAIN |
| Triggering Tables | GL_CODE_COMBINATIONS |
| PL/SQL Stored Procedure | api_glcc_sp |
| PL/SQL User Exit Procedure | MOF_USR_COA_SP |
| Enterprise Service | MXCOA_FROA12 |
| Concurrent job request | <*MAXORA_schema_name*>: Maximo GL COA Master Data |
| Processing Class | COAInExt |
| Java Process | coa\MaxCOAProcess |
| Filtering | Bypass if the value of the COAIN integration control is 0. |

## Outbound Implementation Details

Not applicable

# General ledger journal component

| | |
|---|---|
| Detail Description | This component allows for the outbound synchronization of general ledger journal transactions. |
| Object Structure | MXGLTXN |
| Interface Table | MXGLTXN_IFACE |
| Integration Controls | • GLCURNCY<br>• GLSOURCE<br>• JECATXREF<br>• JEPROJSEND<br>• ORGXREF<br>• SITEXREF |

## Outbound Implementation Details

| | |
|---|---|
| Publish Channel | MXGLTXN_TOOA12 |
| Java Process | gl\MoutGLProcess |
| Processing Class | GLTxnOutExt |
| PL/SQL Stored Procedure | MXE_GL_SP |
| PL/SQL User Exit Procedure | MXE_USR_GL_SP |
| Oracle Open Interface Tables | GL_INTERFACE |
| Filtering | Bypass if any following condition is true:<br><br>• The transaction type is included in the GLSOURCE integration control.<br>• The GL debit or credit account is null.<br>• The GL debit or credit account is not fully specified.<br>• The GL debit and credit accounts are the same.<br>• The transaction cost is zero. |

## Inbound Implementation Details

Not applicable

## Comments

The integration framework excludes journals that have a source business object listed in the GLSOURCE integration control.

# Labor component

| | |
|---|---|
| Detail Description | This component allows for the inbound synchronization of labor data. |
| Object Structure | MXLABOR |
| Interface Table | MXLABOR_IFACE |
| Integration Controls | • CONVHRS<br>• LABXREF<br>• ORGXREF |

## Inbound Implementation Details

| | |
|---|---|
| Synchronization Script | synclabor.sql |
| Oracle Integration Control | LCIN |
| Triggering Tables | • PER_ALL_ASSIGNMENTS_F<br>• PA_COMPENSATION_DETAILS_ALL |
| PL/SQL Stored Procedure | api_lc_pa_sp<br>api_lc_sp |
| PL/SQL User Exit Procedure | • MOF_USR_LC_SP<br>• MOF_USR_LC_PA_SP (applies if PROJPAY value equals 1) |
| Enterprise Service | MXLABOR_FROA12 |
| Concurrent job request set | <*MAXORA_schema_name*>: Maximo Labor and Craft Master Data |
| Processing Class | LaborInExt |
| Java Process | labor\MaxLaborProcess |
| Filtering | Bypass if either following condition is true:<br><br>• The value of the LCIN integration control is 0.<br>• The assignment is not the employee primary assignment. |

## Outbound Implementation Details

Not applicable

# Craft component

| | |
|---|---|
| Detail Description | This component allows for the inbound synchronization of craft data. |
| Object Structure | MXCRAFT |
| Interface Table | MXCRAFT_IFACE |
| Integration Controls | • CONVHRS<br>• CRAFTXREF<br>• ORGXREF |

## Inbound Implementation Details

| | |
|---|---|
| Synchronization Script | synclabor.sql |
| Oracle Integration Control | LCIN |
| Triggering Tables | PER_ALL_ASSIGNMENTS_F |
| PL/SQL Stored Procedure | api_lc_sp |
| PL/SQL User Exit Procedure | MOF_USR_LC_SP |
| Enterprise Service | MXCRAFT_FROA12 |
| Concurrent job request set | <MAXORA_schema_name>: Maximo Labor and Craft Master Data |
| Processing Class | None |
| Java Process | None |
| Filtering | Bypass if the value of the LCIN integration control is 0. |

## Outbound Implementation Details

Not applicable

# Vendor component

| | |
|---|---|
| Detail Description | This component allows for the inbound synchronization of organization-level vendor data. |
| Object Structure | MXVENDOR |
| Interface Table | MXVENDOR_IFACE |
| Integration Controls | • GENUSR<br>• ORGXREF |

## Inbound Implementation Details

| | |
|---|---|
| Synchronization Script | syncvnd.sql |
| Oracle Integration Control | COMIN |
| Triggering Tables | • AP_SUPPLIERS<br>• AP_SUPPLIERS_CONTACTS<br>• AP_SUPPLIER_SITES_ALL<br>• HZ_PARTIES |
| PL/SQL Stored Procedure | • api_vnd_sp<br>• api_vndcont_sp<br>• api_vndcontupd_sp |
| PL/SQL User Exit Procedure | • MOF_USR_VND_SP<br>• MOF_USR_VNDCONT_SP |
| Enterprise Service | MXVENDOR_FROA12 |
| Concurrent job request set | <MAXORA_schema_name>: Maximo Companies and Contact Master Data |
| Processing Class | VendorInExt |
| Java Process | company\MaxComProcess |
| Filtering | Bypass if either of the following conditions are true:<br>• The value of the COMIN integration control is 0.<br>• The vendor is not a purchasing site or a pay site. |

## Outbound Implementation Details

Not applicable

## Comments

As a prerequisite, select the Automatically Add Companies to Company Master check box for the set associated with the organization to which the vendor record is added.

# Purchase requisition component

| | |
|---|---|
| Detail Description | This component allows for the outbound synchronization of purchase request data. It also allows for inbound status changes when an error occurs during the Oracle E-Business Suite import process. |
| Object Structure | MXPR |
| Interface Table | MXPR_IFACE |
| Integration Controls | • DTC_EXP<br>• DTC_INV<br>• GENITEM<br>• GENSTORE<br>• GENUSR<br>• GLCURNCY<br>• GLCURNCYTYPE<br>• LINETYPEXREF<br>• OACATXREF<br>• ORGXREF<br>• PRSEND<br>• PRSTATUSXREF<br>• SITEXREF<br>• USEPRNUM |

## Outbound Implementation Details

| | |
|---|---|
| Publish Channel | MXPR_TOOA12 |
| Java Process | None |
| Processing Class | PROutExt |
| PL/SQL Stored Procedure | MXE_PR_SP |
| PL/SQL User Exit Procedure | MXE_USR_PR_SP |
| Oracle Open Interface Tables | PO_REQUISITIONS_INTERFACE |
| Filtering | Bypass if the status of the PR does not equal a status in the PRSEND control. |

## Inbound Implementation Details

| | |
|---|---|
| Synchronization Script | Not applicable |
| Oracle Integration Control | Not applicable |
| Triggering Tables | • PO_ACTION_HISTORY<br>• PO_REQUISITION_HEADERS_ALL<br>• PO_REQUISITIONS_INTERFACE_ALL |
| PL/SQL Stored Procedure | api_pr_sp |
| PL/SQL User Exit Procedures | MOF_USR_PR_SP |

| | |
|---|---|
| Enterprise Service | MXPR_FROA12 |
| Concurrent job request | • <*MAXORA_schema_name*>: Maximo Purchase Documents Rejected Import<br>• <*MAXORA_schema_name*>: Maximo Requisitions Rejected Import |
| Processing Class | PRInExt |
| Java Process | pr\MaxPRProcess |
| Filtering | None |

# Purchase order component

| | |
|---|---|
| Detail Description | This component allows for the outbound synchronization of purchase order data. It also allows for inbound status changes when an error occurs during the Oracle E-Business Suite import process. |
| Object Structure | MXPO |
| Interface Table | MXPO_IFACE |
| Integration Controls | • DTC_EXP<br>• DTC_INV<br>• GENITEM<br>• GENSTORE<br>• GENUSR<br>• GLCURNCYTYPE<br>• LINETYPEXREF<br>• OACATXREF<br>• ORGXREF<br>• POSTATUSXREF<br>• POTYPEXREF<br>• POSEND<br>• SITEXREF |

## Outbound Implementation Details

| | |
|---|---|
| Publish Channel | MXPO_TOOA12 |
| Java Process | None |
| Processing Class | POOutExt |
| PL/SQL Stored Procedure | MXE_PO_SP |
| PL/SQL User Exit Procedure | • MXE_USR_POH_SP<br>• MXE_USR_POL_SP<br>• MXE_USR_POT_SP |
| Oracle Open Interface Tables | • PO_HEADERS_INTERFACE<br>• PO_LINES_INTERFACE<br>• PO_DISTRIBUTION_INTERFACE |
| Filtering | Bypass if either following condition is true:<br>• The status of the PO does not equal a status in the POSEND integration control.<br>• The purchase order is an internal purchase order. |

## Inbound Implementation Details

| | |
|---|---|
| Synchronization Script | None |
| Oracle Integration Control | POIN |

| | |
|---|---|
| Triggering Tables | • PO_HEADERS_INTERFACE<br>• PO_ACTION_HISTORY<br>• PO_HEADERS_ALL<br>• PO_RELEASES_ALL |
| PL/SQL Stored Procedure | api_pox_sp |
| PL/SQL User Exit Procedure | MOF_USR_POX_SP |
| Enterprise Service | MXPO_FROA12 |
| Concurrent job request sets | • <*MAXORA_schema_name*>: Maximo Purchase Documents Rejected Import<br>• <*MAXORA_schema_name*>: Maximo Requisitions Rejected Import<br>• <*MAXORA_schema_name*>: Maximo Requisitions & Purchase Documents |
| Processing Class | POInExt |
| Java Process | po\MaxPOProcess |
| Filtering | Bypass if either following condition is true:<br><br>• The value of the POIN integration control is 0.<br>• The purchase order is not approved. |

## Comments

The inbound processing described here applies to the status change sent to Maximo Asset Management when the Oracle E-Business Suite import process fails.

# Contract component

| | |
|---|---|
| Detail Description | This component allows for the synchronization of purchase contract data. |
| Object Structure | MXPC |
| Interface Table | MXPC_IFACE |
| Integration Controls | • GENITEM<br>• GENSTORE<br>• GENUSR<br>• GLCURNCYTYPE<br>• LINETYPEXREF<br>• OACATXREF<br>• OAPCDEFORDERUNIT<br>• ORGXREF<br>• PCSEND<br>• SITEXREF |

## Outbound Implementation Details

| | |
|---|---|
| Publish Channel | MXPC_TOOA12 |
| Java Process | pc\MoutPCProcess |
| Processing Class | PCOutExt |
| PL/SQL Stored Procedure | MXE_PC_SP |
| PL/SQL User Exit Procedure | • MXE_USR_PCH_SP<br>• MXE_USR_PCL_SP<br>• MXE_USR_PCT_SP |
| Oracle Open Interface Tables | • PO_HEADERS_INTERFACE_ALL<br>• PO_LINES_INTERFACE_ALL<br>• PO_DISTRIBUTION_INTERFACE_ALL |
| Filtering | Bypass if either of the following conditions are true:<br>• The status of the contract does not equal a status in the PCSEND integration control.<br>• The contract is not a blanket contract. |

## Inbound Implementation Details

| | |
|---|---|
| Synchronization Script | None |
| Oracle Integration Control | PCIN |
| Triggering Tables | • PO_ACTION_HISTORY<br>• PO_HEADERS_ALL<br>• PO_HEADERS_INTERFACE<br>• PO_LINES_ALL |

| | |
|---|---|
| PL/SQL Stored Procedure | • api_pc_sp<br>• api_pcx_sp<br>• api_polx_sp |
| PL/SQL User Exit Procedure | • MOF_USR_PC_SP<br>• MOF_USR_PCX_SP<br>• MOF_USR_POLX_SP |
| Enterprise Service | MXPC_FROA12 |
| Concurrent job request | <*MAXORA_schema_name*>: Maximo Requisitions & Purchase Documents |
| Processing Class | PCInExt |
| Java Process | pc\MaxPCProcess |
| Filtering | Bypass if any of the following conditions are true:<br><br>• The value of the PCIN integration control is 0.<br>• The contract is not a blanket purchase agreement, contract purchase agreement, or planned purchase order.<br>• The status of the contract is not Approved. |

# Comments

The inbound processing described here applies to the status change sent to the Maximo Asset Management when the Oracle E-Business Suite import process fails.

# Receipt component

| | |
|---|---|
| Detail Description | This component allows for the bidirectional synchronization of receipts of non-serialized (non-rotating) items and the outbound synchronization of receipts of serialized (rotating) items. |
| Object Structure | MXRECEIPT |
| Interface Table | MXRECEIPT_IFACE |
| Integration Controls | • GENITEM<br>• GENSTORE<br>• GENUSR<br>• ITMSUBXREF<br>• OACATXREF<br>• ORGXREF<br>• SITEXREF<br>• SUBLOCDEL |

## Outbound Implementation Details

| | |
|---|---|
| Publish Channel | MXRECEIPT_TOOA12 |
| Java Process | rcv\MoutRCVProcess |
| Processing Class | ReceiptOutExt |
| PL/SQL Stored Procedure | • MXE_RCV_SER_SP<br>• MXE_RCV_SP |
| PL/SQL User Exit Procedure | • MXE_USR_RCV_SER_SP<br>• MXE_USR_RCV_SP |
| Oracle Open Interface Tables | • MTL_SERIAL_NUMBERS_INTERFACE<br>• MTL_TRANSACTION_LOTS_INTERFACE<br>• RCV_HEADERS_INTERFACE<br>• RCV_TRANSACTIONS_INTERFACE |
| Filtering | Bypass if the receipt is incomplete. |

## Inbound Implementation Details

| | |
|---|---|
| Synchronization Script | None |
| Oracle Integration Control | RCVIN |
| Triggering Tables | RCV_TRANSACTIONS |
| PL/SQL Stored Procedure | api_rcv_sp |
| PL/SQL User Exit Procedure | MOF_USR_RCV_SP |
| Enterprise Service | MXRECEIPT_FROA12 |
| Concurrent job request | <*MAXORA_schema_name*>: Maximo Receipts, Returns & Corrections |

| | |
|---|---|
| Processing Class | ReceiptInExt |
| Java Process | rcv\MaxRCVProcess |
| Filtering | Bypass if any of the following conditions are true:<br><br>• The value of the RCVIN integration control is 0.<br>• The quantity on the receipt is not greater than 0.<br>• The transaction type is not Deliver, Reject, or Return to Receiving.<br>• The receipt is incomplete.<br>• The source document is not a purchase order. |

# Rotating item receipt component

| | |
|---|---|
| Detail Description | This component allows for the inbound synchronization of receipts of serialized (rotating) items. |
| Object Structure | MXRCVROTITM |
| Interface Table | MXRCVROTITM_IFACE |
| Integration Controls | • GENITEM<br>• GENSTORE<br>• GENUSR<br>• ITMSUBXREF<br>• OACATXREF<br>• ORGXREF<br>• SITEXREF<br>• SUBLOCDEL |

## Inbound Implementation Details

| | |
|---|---|
| Synchronization Script | None |
| Oracle Integration Control | RCVIN |
| Triggering Tables | RCV_SERIAL_TRANSACTIONS |
| PL/SQL Stored Procedure | api_rcvrot_sp |
| PL/SQL User Exit Procedure | MOF_USR_RCVROT_SP |
| Enterprise Service | MXRCVROTITM_FROA12 |
| Concurrent job request | <*MAXORA_schema_name*>: Maximo Receipts, Returns & Corrections |
| Processing Class | RcvRotItmInExt |
| Java Process | rcv\MaxRcvRotItmMProcess |
| Filtering | Bypass if any of the following conditions are true:<br>• The value of the RCVIN integration control is 0.<br>• The quantity on the receipt is not greater than 0.<br>• The transaction type is not Deliver, Reject, or Return to Receiving.<br>• The receipt is incomplete.<br>• The source document is not a purchase order. |

## Outbound Implementation Details

Not applicable

# Invoice component

| | |
|---|---|
| Detail Description | This component allows for the bidirectional synchronization of invoice data. |
| Object Structure | MXINVOICE |
| Interface Table | MXINVOICE_IFACE |
| Integration Controls | • APLINETYPE<br>• APSEND<br>• GENUSR<br>• GLCURNCY<br>• GLCURNCYTYPE<br>• INVCSTATUSXREF<br>• INVCTYPEXREF<br>• ORGXREF<br>• SITEXREF |

## Outbound Implementation Details

| | |
|---|---|
| Publish Channel | MXINVOICE_TOOA12 |
| Java Process | None |
| Processing Class | InvoiceOutExt |
| PL/SQL Stored Procedure | MXE_AP_SP |
| PL/SQL User Exit Procedure | MXE_USR_AP_SP |
| Oracle Open Interface Table(s) | • AP_INVOICES_INTERFACE<br>• AP_INVOICE_LINES_INTERFACE |
| Filtering | Bypass if the status of the invoice doe not match a status in the APSEND integration control. |

## Inbound Implementation Details

| | |
|---|---|
| Synchronization Script | None |
| Oracle Integration Control | APIN |
| Triggering Tables | • AP_INVOICES_ALL<br>• AP_INVOICE_DISTRIBUTIONS_ALL |
| PL/SQL Stored Procedure | • api_invcappr_sp<br>• api_apx_sp |
| PL/SQL User Exit Procedure | • MOF_USR_AP_SP<br>• MOF_USR_APX_SP |
| Integration Point | MXINVOICE_FROA12 |
| Concurrent job request set | <*MAXORA_schema_name*>: Maximo Invoices & Payments |

| | |
|---|---|
| Processing Class | InvoiceInExt |
| Java Process | ap\MaxAPProcess |
| Filtering | Bypass if either of the following conditions are true:<br>• The value of the APIN integration control is 0.<br>• The invoice is not validated, the receipt is incomplete, or the items are not sent to their final destination. |

# Item component

| | |
|---|---|
| Detail Description | This component provides for bidirectional synchronization of item data between Maximo Asset Management and Oracle E-Business Suite. |
| Object Structure | MXITEM |
| Interface Table | MXITEM_IFACE |
| Integration Controls | • CATEGORYXREF<br>• LANGXREF<br>• LOTTYPEXREF<br>• OAITMLOTPREFIX<br>• OAITMLOTSTARTNUM<br>• ORGXREF |

## Outbound Implementation Details

| | |
|---|---|
| Publish Channel | MXITEM_TOOA12 |
| Java Process | item\MoutItemProcess |
| Processing Class | ItemOutExt |
| PL/SQL Stored Procedure | MXE_ITM_SP |
| PL/SQL User Exit Procedure | MXE_USR_ITM_SP |
| Oracle Open Interface Tables | MTL_SYSTEM_ITEMS_INTERFACE |
| Filtering | None |

## Inbound Implementation Details

| | |
|---|---|
| Synchronization Script | syncitm.sql |
| Oracle Integration Control | ITEMIN |
| Triggering Tables | • MTL_SYSTEM_ITEMS<br>• MTL_SYSTEM_ITEMS_TL |
| PL/SQL Stored Procedure | api_item_sp |
| PL/SQL User Exit Procedure | MOF_USR_ITM_SP |
| Enterprise Service | MXITEM_FROA12 |
| Concurrent job request set | <*MAXORA_schema_name*>:Maximo Item & Inventory Master Data |
| Processing Class | ItemInExt |
| Java Process | item\MaxItemProcess |
| Filtering | Bypass if the value of the ITEMIN integration control is 0. |

# Inventory component

| | |
|---|---|
| Detail Description | This component provides for the bidirectional synchronization of inventory (item-storeroom) data. |
| Object Structure | MXINVENTORY |
| Interface Table | MXINVENTORY_IFACE |
| Integration Controls | • CATEGORYXREF<br>• GENSTORE<br>• LANGXREF<br>• ORGXREF<br>• SITEXREF |

## Outbound Implementation Details

| | |
|---|---|
| Publish Channel | MXINVENTORY_TOOA12 |
| Java Process | inventory\MoutInvProcess |
| Processing Class | InventoryOutExt |
| PL/SQL Stored Procedure | MXE_INV_SP |
| PL/SQL User Exit Procedure | MXE_USR_INV_SP |
| Oracle Interface Tables | MTL_SYSTEM_ITEMS_INTERFACE |
| Filtering | None |

## Inbound Implementation Details

| | |
|---|---|
| Synchronization Script | syncitm.sql |
| Oracle Integration Control | ITEMIN |
| Triggering Tables | • MTL_SYSTEM_ITEMS<br>• MTL_MATERIAL_TRANSACTIONS |
| PL/SQL Stored Procedure | api_inv_sp |
| PL/SQL User Exit Procedure | MOF_USR_INV_SP |
| Enterprise Service | MXINVENTORY_FROA12 |
| Concurrent job request set | <*MAXORA_schema_name*>:Maximo Item & Inventory Master Data |
| Processing Class | InventoryInExt |
| Java Process | None |
| Filtering | Bypass if the value of the ITEMIN integration control is 0. |

# Inventory balance component

| | |
|---|---|
| Detail Description | This component provides for inbound synchronization of Oracle-owned inventory balances. |
| Object Structure | MXINVBAL |
| Interface Table | MXINVBAL_IFACE |
| Integration Controls | • ORGXREF<br>• SITEXREF<br>• SUBLOCDEL |

## Inbound Implementation Details

| | |
|---|---|
| Synchronization Script | syncbal.sql |
| Oracle Integration Control | INVBALIN |
| Triggering Tables | MTL_MATERIAL_TRANSACTIONS |
| PL/SQL Stored Procedure | api_bal_sp |
| PL/SQL User Exit Procedure | MOF_USR_BAL_SP |
| Enterprise Service | MXINVBAL_FROA12 |
| Concurrent job request | <*MAXORA_schema_name*>: Maximo Inventory Balance Master Data |
| Processing Class | InvBalInExt |
| Java Process | invbalances\MaxInvBalancesProcess |
| Filtering | Bypass if any following conditions is true:<br>• The value of the INVBALIN integration control is 0.<br>• The item is a rotating item.<br>• The item is Maximo Asset Management-owned. |

## Outbound Implementation Details

Not applicable

# Material reservation component

| | |
|---|---|
| Detail Description | This component provides for outbound synchronization of inventory reservations data. |
| Object Structure | MXINVRES |
| Interface Table | MXINVRES_IFACE |
| Integration Controls | • GENSTORE<br>• GENUSR<br>• ITMSUBXREF<br>• ORGXREF<br>• SITEXREF<br>• SUBLOCDEL |

## Outbound Implementation Details

| | |
|---|---|
| Publish Channel | MXINVRES_TOOA12 |
| Java Process | inventory\MoutRSVProcess |
| Processing Class | InvResOutExt |
| PL/SQL Stored Procedure | MXE_INVRESERVE_SP |
| PL/SQL User Exit Procedure | MXE_USR_INVRESERVE_SP |
| Oracle Integration Tables | MTL_RESERVATIONS_INTERFACE |
| Filtering | None |

## Inbound Implementation Details

Not applicable

# Material issue component

| | |
|---|---|
| Detail Description | This component provides for inbound synchronization of material issue data. |
| Object Structure | MXINVISSUE |
| Interface Table | MXINVISSUE_IFACE |
| Integration Controls | • GENSTORE<br>• GENUSR<br>• SITEXREF<br>• SUBLOCDEL |

## Inbound Implementation Details

| | |
|---|---|
| Synchronization Script | None |
| Oracle Integration Control | ISUIN |
| Triggering Tables | MTL_MATERIAL_TRANSACTIONS |
| PL/SQL Stored Procedure | api_mtlisu_sp |
| PL/SQL User Exit Procedure | MOF_USR_ISU_SP |
| Enterprise Service | MXINVISSUE_FROA12 |
| Concurrent job request | <*MAXORA_schema_name*>: Maximo Inventory Balance Master Data |
| Processing Class | InvIssueInExt |
| Java Process | isu\MaxIsuProcess |
| Filtering | Bypass if either of the following conditions are true:<br>• The value of the ISUIN integration control is 0.<br>• The issue is not a material issue or a receipt. |

## Outbound Implementation Details

Not applicable

# Project and task component

| | |
|---|---|
| Detail Description | This component provides for inbound synchronization of project and task data. |
| Object Structure | MXPROJ |
| Interface Table | MXPROJ_IFACE |
| Integration Controls | • FCSTATUSXREF<br>• ORGXREF<br>• SITEXREF |
| Oracle Integration Controls | FCEXTRACTDATE |
| Oracle Open Interface Table | None |
| User Action (in Oracle Project Accounting) | **1** Create or update work breakdown structure and project costs in Oracle Project Accounting.<br>**2** Run the following programs:<br>• PRC: Distribute Labor Cost<br>• PRC: Distribute Usage and Miscellaneous Costs<br>• PRC: Update Project Summary Amounts<br>• PRC: Maximo Asset Management Financial Update from Oracle |
| Conditions | • If a user enters the following optional parameters, the data must match the specified values:<br>• Operating Unit ID<br>• Project Number<br>• Budget Type<br>• Start Date<br>The current operating unit is the default operating unit ID.<br>If you enter a budget type, the process passes all budget amounts for that budget type to Maximo Asset Management.<br>If you do not enter a budget type and the project has one budget type, the process passes the budget amount for that type to Maximo Asset Management. If the project has multiple budget types, the process sets the budget amount in Maximo Asset Management to null.<br>The process extracts records updated on or after the start date. If you do not enter a start date, the screen displays the date in the Oracle FCEXTRACTDATE control. If you erase that value, the process extracts all projects that meet the project number and operating unit ID criteria.<br>• The current project accounting period must be open.<br>• The budget associated with the project or task must be base-lined and valid. |

| | |
|---|---|
| Integration Action | **1** The Maximo Asset Management Financial Control Update from Oracle concurrent process extracts updated the work breakdown structure and summarized cost data.<br><br>**2** The integration framework calls the MOF_USR_FC_SP user exit procedure.<br><br>**3** The integration framework writes the extracted data to the MXPROJ_IFACE table and updates the MXIN_INTER_TRANS queue table.<br><br>**4** The IFACETABLECONSUMER cron task polls the MXIN_INTER_TRANS table for new records, retrieves the MXPROJ_IFACE record, and writes it to the inbound JMS queue.<br><br>**5** The integration framework calls the following procedures:<br><br>**a** User exit procedure preprocessing method (optional)<br><br>**b** ERP exit procedure ProjInExt<br><br>**c** User exit procedure postprocessing method (optional)<br><br>**6** The integration framework applies any processing rules to the object structure.<br><br>**7** The adapter builds the FINCTRLID business object from the object structure, and assigns a unique financial control ID to every new project and task.<br><br>**8** The integration framework passes the business object to Maximo Asset Management for standard processing. |

# Project transactions component

| | |
|---|---|
| Detail Description | This component provides for outbound synchronization of actuals data. |
| Object Structure | MXGLTXN |
| Interface Table | MXPROJTXN_IFACE |
| Integration Controls | • CHARGEORG<br>• EXPENDITEM<br>• EXPENDLABOR<br>• EXPENDTOOL<br>• NLRORG<br>• ORGXREF<br>• PROJAP<br>• PROJPO<br>• PROJPR<br>• PROJSEND<br>• RESLEVELITEM<br>• RESLEVELLABOR<br>• RESLEVELTOOL<br>• SITEXREF<br>• SRCTIM<br>• SRCUSE |
| Oracle Integration Controls | None |
| Oracle Open Interface Table | PA_TRANSACTION_INTERFACE |
| User Action (in Maximo Asset Management) | Enter labor, material, services, or tool actuals, or a receipt. |
| Conditions | • The transaction type is listed in PROJSEND control.<br>• The work order is associated with a chargeable task.<br>• The work order is associated with a financial control ID. |

| | |
|---|---|
| Integration Framework Action | **1** The integration framework builds the object structure from the applicable business objects<br><br>**2** The integration framework applies any processing rules to the object structure.<br><br>**3** The integration framework calls the following Java procedures:<br><br>**a** User exit procedure preprocessing method (optional)<br><br>**b** ERP exit procedure ProjTxnOutExt<br><br>**c** User exit procedure postprocessing method (optional)<br><br>**4** The integration framework writes the transaction to the outbound JMS queue.<br><br>**5** The outbound JMS consumer retrieves and passes the transaction from the outbound queue to the router, which invokes interface table handler.<br><br>**6** The interface table handler writes the transaction to the MXPROJTXN_IFACE table and updates the MXOUT_INTER_TRANS queue table.<br><br>**7** The integration framework calls the MXE_USR_PA_ACT_SP user exit procedure.<br><br>**8** The integration framework moves the record from the interface table to the Oracle open interface table PA_TRANSACTION_INTERFACE. |
| User Action (in Oracle) | Run the following program:<br><br><*MAXORA_schema_name*>PRC: Transaction Import<br><br>This activity imports the data into the PA_EXPENDITURES_ALL table in the Oracle E-Business Suite database. |

# Oracle-specific object structure fields

# B

Maximo Asset Management has standard object structures that you can use in the integration with the Oracle E-Business Suite.

The value in the Pers (persistent) column indicates whether the field is persistent (value 1) or non-persistent (0). Depending on that status, you can see the field in the Object Structures application.

The table that begins on the next page contains the following information about each new field:

| Column | Description |
|---|---|
| Table | Table name |
| Field | Field name |
| Title | Title |
| Remarks | Description or comments |
| Val | Default value |
| Type | Field type |
| Len | Length |
| Sc | Scale |
| Pers | Persistent |
| Reqd | Required |
| Same as Attribute | Same as attribute |
| Same as Object | Same as object |

### *Oracle-specific Object Structure Fields*

| Table | Column | Title | Remarks | Val | Type | Len | Sc | Pers | Reqd | Same As Attribute | Same As Object |
|---|---|---|---|---|---|---|---|---|---|---|---|
| COMPANIES | OA_OLD_VNDSITECODE | Old Vendor Site Code | Used when vendor site code is changed in Oracle | | ALN | 15 | 0 | 0 | 0 | | |
| COMPANIES | OA_PAYSITE_FLG | Pay Site | Pay site flag | 0 | YORN | | 0 | 1 | 1 | | |
| COMPANIES | OA_PURCHSITE_FLG | Purchasing Site | Purchasing site flag | 0 | YORN | | 0 | 1 | 1 | | |
| CONTRACTLINE | OA_CATEGORYID | Category ID | Oracle category ID | | Integer | | 0 | 0 | 0 | | |
| CONTRACTLINE | OA_IFACEMESSAGE | Interface Message | A general message for Oracle interface transaction | | ALN | 2000 | 0 | 0 | 0 | | |
| CONTRACTLINE | OA_IFACESTATUS | Interface Status | Oracle interface transaction status | | ALN | 40 | 0 | 0 | 0 | | |
| CONTRACTLINE | OA_IFACETIMESTAMP | Interface Timestamp | The date and time the transaction completed or failed | | Datetime | | 0 | 0 | 0 | | |
| CONTRACTLINE | OA_ITEM_ID | Item ID | Oracle item ID | | Integer | | 0 | 0 | 0 | | |
| CONTRACTLINE | OA_STORELOC | Inventory Org | Oracle inventory organization | | | | | 1 | 0 | LOCATION | LOCATIONS |
| INVBALANCES | OA_LOC_SEGMENTS | Locator Segments | Oracle locator segments | | ALN | 40 | 0 | 1 | 0 | | |
| INVENTORY | EXT_ACTIVE | Item Active | Is external item active | 1 | YORN | | 0 | 1 | 1 | | |
| INVENTORY | OA_IFACEMESSAGE | Interface Message | A general message for Oracle interface transaction | | ALN | 2000 | 0 | 0 | 0 | | |

| Table | Column | Title | Remarks | Val | Type | Len | Sc | Pers | Reqd | Same As Attribute | Same As Object |
|---|---|---|---|---|---|---|---|---|---|---|---|
| INVENTORY | OA_IFACESTATUS | Interface Status | Oracle interface transaction status | | ALN | 40 | 0 | 0 | 0 | | |
| INVENTORY | OA_IFACETIMESTAMP | Interface Timestamp | The date and time the transaction completed or failed | | Datetime | | 0 | 0 | 0 | | |
| INVENTORY | OA_IO_FLAG | Internal Order Enabled | Oracle internal order enabled flag | 0 | YORN | | 0 | 1 | 1 | | |
| INVENTORY | OA_PURCH_FLAG | Purchasing Enabled | Oracle purchasing enabled flag | 1 | YORN | | 0 | 1 | 1 | | |
| INVENTORY | OA_INV_ITEM_FLAG | Oracle Inventory Item Flag | Oracle inventory item flag | 1 | YORN | | 0 | 0 | 0 | | |
| INVENTORY | OA_ITEMDESC | Description | Item description | | | | | 0 | 0 | DESCRIPTION | ITEM |
| INVENTORY | OA_LONGDESCRIPTION | | Long description | | | | | 0 | 0 | LDTEXT | LONGDESCRIPTION |
| INVENTORY | OA_SERIAL_CTRL | | Is item rotating item | | | | | 0 | 0 | ROTATING | ITEM |
| INVENTORY | OA_RCV_RTE_ID | | Inspection required | | | | | 0 | 0 | INSPECTION REQUIRED | ITEM |
| INVENTORY | OA_SET_PROCESS_ID | Set Process ID | Oracle set process ID | | Integer | | 0 | 0 | 0 | | |
| INVOICE | OA_GLCURNCY | GL Currency | Oracle GL currency | | ALN | 15 | 0 | 0 | 0 | | |
| INVOICE | OA_GLCURNCYTYPE | Currency Type | Oracle GL currency type | | ALN | 25 | 0 | 0 | 0 | | |
| INVOICELINE | OA_APLINETYPE | Payables Line Type | Oracle payables default line type | | ALN | 25 | 0 | 0 | 0 | | |

| Table | Column | Title | Remarks | Val | Type | Len | Sc | Pers | Reqd | Same As Attribute | Same As Object |
|---|---|---|---|---|---|---|---|---|---|---|---|
| INVOICELINE | OA_CONTRACTLINENUM | Contract Linenum | Line number of each contract line | | | | | 0 | 0 | CONTRACT LINENUM | CONTRACTLINE |
| INVOICELINE | OA_IFACEMESSAGE | Interface Message | A general message for Oracle interface transaction | | ALN | 2000 | 0 | 0 | 0 | | |
| INVOICELINE | OA_IFACESTATUS | Interface Status | Oracle interface transaction status | | ALN | 40 | 0 | 0 | 0 | | |
| INVOICELINE | OA_IFACETIMESTAMP | Interface Timestamp | The date and time the transaction completed or failed | | Datetime | | 0 | 0 | 0 | | |
| INVOICELINE | OA_OWNERSYSID | PO Owner | PO owner system ID | | | | | 0 | 0 | OWNER1 SYSID | MXCOLLAB |
| INVOICELINE | OA_POTYPE | PO Type | PO type | | | | | 0 | 0 | POTYPE | PO |
| INVRESERVE | OA_CHANGEBY | Change By | Change by | | | | | 0 | 0 | PERSONID | PERSON |
| INVRESERVE | OA_IFACEMESSAGE | Interface Message | A general message for Oracle interface transaction | | ALN | 2000 | 0 | 0 | 0 | | |
| INVRESERVE | OA_IFACESTATUS | Interface Status | Oracle interface transaction status | | ALN | 40 | 0 | 0 | 0 | | |
| INVRESERVE | OA_IFACETIMESTAMP | Interface Timestamp | The date and time the transaction completed or failed | | Datetime | | 0 | 0 | 0 | | |
| INVRESERVE | OA_ISSUEUNIT | Issue Unit | Issue unit | | | | | 0 | 0 | MEASURE UNITID | MEASUREUNIT |
| INVRESERVE | OA_ITEM_ID | Inventory Item ID | Oracle inventory item ID | | Integer | | 0 | 0 | 0 | | |

| Table | Column | Title | Remarks | Val | Type | Len | Sc | Pers | Reqd | Same As Attribute | Same As Object |
|---|---|---|---|---|---|---|---|---|---|---|---|
| INVRESERVE | OA_LOCATOR_ID | Locator ID | Oracle locator ID | | Integer | | 0 | 0 | 0 | | |
| INVRESERVE | OA_ORDERUNIT | Order Unit | Order unit | | | | | 0 | 0 | MEASURE UNITID | MEASUREUNIT |
| INVRESERVE | OA_SUBINV_CODE | Sub Inventory Code | Oracle subinventory code | | ALN | 10 | 0 | 0 | 0 | | |
| ITEM | OA_SET_OF_BKS_IDS | Sets of Books | All SET_OF_ BOOKS_IDs separated by # | | ALN | 2000 | 0 | 0 | 0 | | |
| ITEM | OA_SET_PROCESS_ID | Set Process ID | Oracle Set Process ID | | Integer | | 0 | 0 | 0 | | |
| ITEM | OA_IFACEMESSAGE | Interface Message | A general message for Oracle interface transaction | | ALN | 2000 | 0 | 0 | 0 | | |
| ITEM | OA_IFACESTATUS | Interface Status | Oracle interface transaction status | | ALN | 40 | 0 | 0 | 0 | | |
| ITEM | OA_IFACETIMESTAMP | Interface Timestamp | The date and time the transaction completed or failed | | Datetime | | 0 | 0 | 0 | | |
| ITEM | OA_ITM_LOT_NUMBER | Starting Lot Number | Starting Lot Number | | ALN | 30 | 0 | 0 | 0 | | |
| ITEM | OA_ITM_LOT_PREFIX | Starting Lot Prefix | Starting Lot Prefix | | ALN | 30 | 0 | 0 | 0 | | |
| MATRECTRANS | OA_CATEGORYID | Category ID | Oracle category ID | | Integer | | 0 | 0 | 0 | | |
| MATRECTRANS | OA_CONTRACTLINENUM | Contract Linenum | Line number of each contract line | | | | | 0 | 0 | CONTRACT LINENUM | CONTRACTLINE |
| MATRECTRANS | OA_CONTRACTNUM | Contract Number | Contract number | | | | | 0 | 0 | CONTRACT NUM | CONTRACT |

| Table | Column | Title | Remarks | Val | Type | Len | Sc | Pers | Reqd | Same As Attribute | Same As Object |
|---|---|---|---|---|---|---|---|---|---|---|---|
| MATRECTRANS | OA_IFACEMESSAGE | Interface Message | A general message for Oracle interface transaction | | ALN | 2000 | 0 | 0 | 0 | | |
| MATRECTRANS | OA_IFACESTATUS | Interface Status | Oracle interface transaction status | | ALN | 40 | 0 | 0 | 0 | | |
| MATRECTRANS | OA_IFACETIMESTAMP | Interface Timestamp | The date and time the transaction completed or failed | | Datetime | | 0 | 0 | 0 | | |
| MATRECTRANS | OA_ITEM_ID | Item ID | Oracle item ID | | Integer | | 0 | 0 | 0 | | |
| MATRECTRANS | OA_LOCATOR_ID | Locator ID | Oracle locator ID | | Integer | | 0 | 0 | 0 | | |
| MATRECTRANS | OA_OWNERSYSID | PO Owner | PO owner system ID | | | | | 0 | 0 | OWNER1 SYSID | MXCOLLAB |
| MATRECTRANS | OA_POTYPE | PO Type | PO type | | | | | 0 | 0 | POTYPE | PO |
| MATRECTRANS | OA_REQUIREDDATE | Required Date | Required date to Oracle | | | | | 0 | 0 | REQUIRED DATE | PO |
| MATRECTRANS | OA_SHIPTO | Ship To | Oracle ship to location | | | | | 0 | 0 | ADDRESS CODE | ADDRESS |
| MATRECTRANS | OA_SUBINV_CODE | Sub Inventory Code | Oracle subinventory code | | ALN | 10 | 0 | 0 | 0 | | |
| MATRECTRANS | OA_VENDOR | Company | Oracle vendor | | | | | 0 | 0 | COMPANY | COMPANIES |
| MATUSETRANS | OA_GLCURNCY | GL Currency | Oracle GL currency | | ALN | 15 | 0 | 0 | 0 | | |
| MATUSETRANS | OA_JE_CATEGORY | GL Category | Oracle journal category | | ALN | 25 | 0 | 0 | 0 | | |
| PO | OA_GLCURNCYTYPE | Currency Type | Oracle GL currency type | | ALN | 25 | 0 | 0 | 0 | | |
| PO | OA_POAPI | PO API | Oracle PO API | | Upper | 3 | 0 | 0 | 0 | | |

| Table | Column | Title | Remarks | Val | Type | Len | Sc | Pers | Reqd | Same As Attribute | Same As Object |
|---|---|---|---|---|---|---|---|---|---|---|---|
| POLINE | OA_CATEGORYID | Category ID | Oracle category ID | | Integer | | 0 | 0 | 0 | | |
| POLINE | OA_CHARGEORG | Charge Org | Oracle Projects charge organization | | ALN | 1 | 0 | 0 | 0 | | |
| POLINE | OA_DEST_TYPE_CODE | Destination Type | Oracle destination type code | | ALN | 25 | 0 | 0 | 0 | | |
| POLINE | OA_EXPENDITEM | Expenditure Item | Oracle Projects expenditure item | | ALN | 1 | 0 | 0 | 0 | | |
| POLINE | OA_IFACEMESSAGE | Interface Message | A general message for Oracle interface transaction | | ALN | 2000 | 0 | 0 | 0 | | |
| POLINE | OA_IFACESTATUS | Interface Status | Oracle Interface transaction status | | ALN | 40 | 0 | 0 | 0 | | |
| POLINE | OA_IFACETIMESTAMP | Interface Timestamp | The date and time the transaction completed or failed | | Datetime | | 0 | 0 | 0 | | |
| POLINE | OA_ITEM_ID | Item ID | Oracle item ID | | Integer | | 0 | 0 | 0 | | |
| POLINE | OA_RATELINE | Rate Type PO Line | Indicates if PO Line is of Rate Type | | YORN | 1 | 0 | 1 | 0 | | |
| PR | OA_GLCURNCY | GL Currency | Oracle GL currency | | ALN | 15 | 0 | 0 | 0 | | |
| PR | OA_GLCURNCYTYPE | GL Currency Type | GL currency type | | ALN | 30 | 0 | 0 | 0 | | |
| PR | OA_USEPRNUM | Use PR Number | Use PR number flag | 0 | YORN | | 0 | 0 | 1 | | |
| PRLINE | OA_CATEGORYID | Category ID | Oracle category ID | | Integer | | 0 | 0 | 0 | | |

| Table | Column | Title | Remarks | Val | Type | Len | Sc | Pers | Reqd | Same As Attribute | Same As Object |
|---|---|---|---|---|---|---|---|---|---|---|---|
| PRLINE | OA_CHARGEORG | Projects Charge Org | Oracle Projects charge organization | | ALN | 50 | 0 | 0 | 0 | | |
| PRLINE | OA_DEST_TYPE_CODE | Destination Type | Oracle destination type code | | ALN | 25 | 0 | 0 | 0 | | |
| PRLINE | OA_EXPENDITEM | Expenditure Item | Oracle Projects expenditure item | | ALN | 50 | 0 | 0 | 0 | | |
| PRLINE | OA_IFACEMESSAGE | Interface Message | A general message for Oracle interface transaction | | ALN | 2000 | 0 | 0 | 0 | | |
| PRLINE | OA_IFACESTATUS | Interface Status | Oracle interface transaction status | | ALN | 40 | 0 | 0 | 0 | | |
| PRLINE | OA_IFACETIMESTAMP | Interface Timestamp | The date and time the transaction completed or failed | | Datetime | | 0 | 0 | 0 | | |
| PRLINE | OA_ITEMNUM | Item Number | Oracle item number | | ALN | 30 | 0 | 0 | 0 | | |
| PRLINE | OA_ITEM_ID | Item ID | Oracle item ID | | Integer | | 0 | 0 | 0 | | |
| PRLINE | OA_PROJECTID | Project ID | Oracle Projects ID | | Integer | | 0 | 0 | 0 | | |
| PRLINE | OA_STORELOC | Storeroom | Oracle storeroom location | | ALN | 12 | 0 | 0 | 0 | | |
| PRLINE | OA_TASKID | Task ID | Oracle task ID | | Integer | | 0 | 0 | 0 | | |
| PURCHVIEW | OA_GLCURNCYTYPE | Currency Type | Oracle GL currency type | | ALN | 25 | 0 | 0 | 0 | | |
| PURCHVIEW | OA_POAPI | PO API | Oracle PO API | | Upper | 3 | 0 | 0 | 0 | | |

# Index

# E



# F



# G



# H



# I

## J



## K



## L

## M



## N



## O



## P

## Q



## R



## S

# T



# U



# V



# W

# Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing

IBM Corporation

North Castle Drive

Armonk, NY 10504-1785

U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing

Legal and Intellectual Property Law

IBM Japan Ltd.

19-21, Nihonbashi-Hakozakicho, Chuo-ku

Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new

editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation

2Z4A/101

11400 Burnet Road

Austin, TX 78758

U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

## Trademarks

IBM®, the IBM logo, and ibm.com® are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at www.ibm.com/legal/copytrade.shtml.

## Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

**Applicability**

These terms and conditions are in addition to any terms of use for the IBM website.

**Personal use**

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

**Commercial use**

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

**Rights**

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

**Privacy Policy Considerations**

IBM Software products, including software as service solutions, (“Software Offerings”) may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering’s use of cookies is set forth in the following section.

Depending on the configuration that is deployed in the application server, this Software Offering may use session cookies for session management. These cookies can optionally be enabled or disabled, but disabling the cookies will also

disable the functionality that they enable. See the application server documentation for additional information.

Optionally, you can also use functionality, such as typeahead, to store data locally on the browser. The data that is stored locally could contain personally identifiable information: Data validation.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM's Privacy Policy at http://www.ibm.com/privacy and IBM's Online Privacy Statement at http://www.ibm.com/privacy/details in the section entitled "Cookies, Web Beacons and Other Technologies" and the "IBM Software Products and Software-as-a-Service Privacy Statement" at http://www.ibm.com/software/info/product-privacy.