

# Oracle

## SCM Cloud

# Implementing Order Management

---

**Release 12**

This guide also applies to on-premises implementations

Oracle® SCM Cloud Implementing Order Management

Part Number E74103-03

Copyright © 2011-2017, Oracle and/or its affiliates. All rights reserved.

Author: carl casey

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

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

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

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

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

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

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

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

The business names used in this documentation are fictitious, and are not intended to identify any real companies currently or previously in existence.

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

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

# Contents

<b>Preface</b>	<b>i</b>
<hr/>	
<b>1 Introduction</b>	<b>1</b>
Oracle Order Management Cloud: Overview	1
Order Management Architecture	5
<b>2 Implementing Order Management Cloud</b>	<b>13</b>
<hr/>	
Implementing Order-to-Cash	13
Enabling Offerings	28
Implementing Business-to-Business Order Management	29
Implementing Drop Ship	38
<b>3 Integrating Order Management Cloud</b>	<b>55</b>
<hr/>	
Integrating Order Management with External Systems: Overview	55
Integrating Order Management with Upstream Source Systems	56
Integrating Order Management with Downstream Fulfillment Systems	60
Using Integration Cloud Service to Integrate Order Management with Other Systems	67
More Information About Integrating Order Management	79
<b>4 Importing and Transforming Source Orders</b>	<b>85</b>
<hr/>	
Importing Orders	85
Transforming Source Orders to Sales Orders	93
<b>5 Controlling Application Behavior</b>	<b>113</b>
<hr/>	
Managing Order Management Parameters: Procedure	113
Managing Order Profiles: Procedure	116

<b>6</b>	<b>Controlling Order Processing</b>	<b>119</b>
	Controlling Statuses for Sales Orders	119
	Constraining Orders	132
	Managing Change Orders	141
	Notifying Systems When Orders Change	150
	Using Jeopardy to Manage Delays	155
	Customizing Task Types	160
	Customizing Fields	170
<b>7</b>	<b>Customizing Orchestration Processes</b>	<b>175</b>
	Defining Orchestration Processes	175
	Defining Subprocesses In Orchestration Processes	191
	Assigning Orchestration Processes	203
	Pausing Orchestration Processes	214
<b>8</b>	<b>Using Visual Information Builder to Create Rules</b>	<b>229</b>
	Using Visual Information Builder to Create Rules: Explained	229
	Using Visual Information Builder to Manage External Interface Routing Rules: Procedure	230
	Using Visual Information Builder to Manage Pretransformation Rules: Procedure	232
	Using Visual Information Builder to Manage Process Assignment Rules: Procedure	235
<b>9</b>	<b>Other Implementation Work</b>	<b>243</b>
	Administering Messaging and Social Networking	243
	Administering Collaboration Messaging	245

# Preface

This preface introduces information sources that can help you use the application.

## Oracle Applications Help

Use the help icon  to access Oracle Applications Help in the application. If you don't see any help icons on your page, click the Show Help icon  in the global header. Not all pages have help icons. You can also access Oracle Applications Help at <https://fusionhelp.oracle.com>.

## Using Applications Help

 **Watch:** This video tutorial shows you how to find help and use help features.

## Additional Resources

- **Community:** Use [Oracle Applications Customer Connect](#) to get information from experts at Oracle, the partner community, and other users.
- **Guides and Videos:** Go to the [Oracle Help Center](#) to find guides and videos.
- **Training:** Take courses on Oracle Cloud from [Oracle University](#) .

## Documentation Accessibility

For information about Oracle's commitment to accessibility, see the [Oracle Accessibility Program](#) .

## Comments and Suggestions

Please give us feedback about Oracle Applications Help and guides! You can send e-mail to: [oracle\\_fusion\\_applications\\_help\\_ww\\_grp@oracle.com](mailto:oracle_fusion_applications_help_ww_grp@oracle.com).



# 1 Introduction

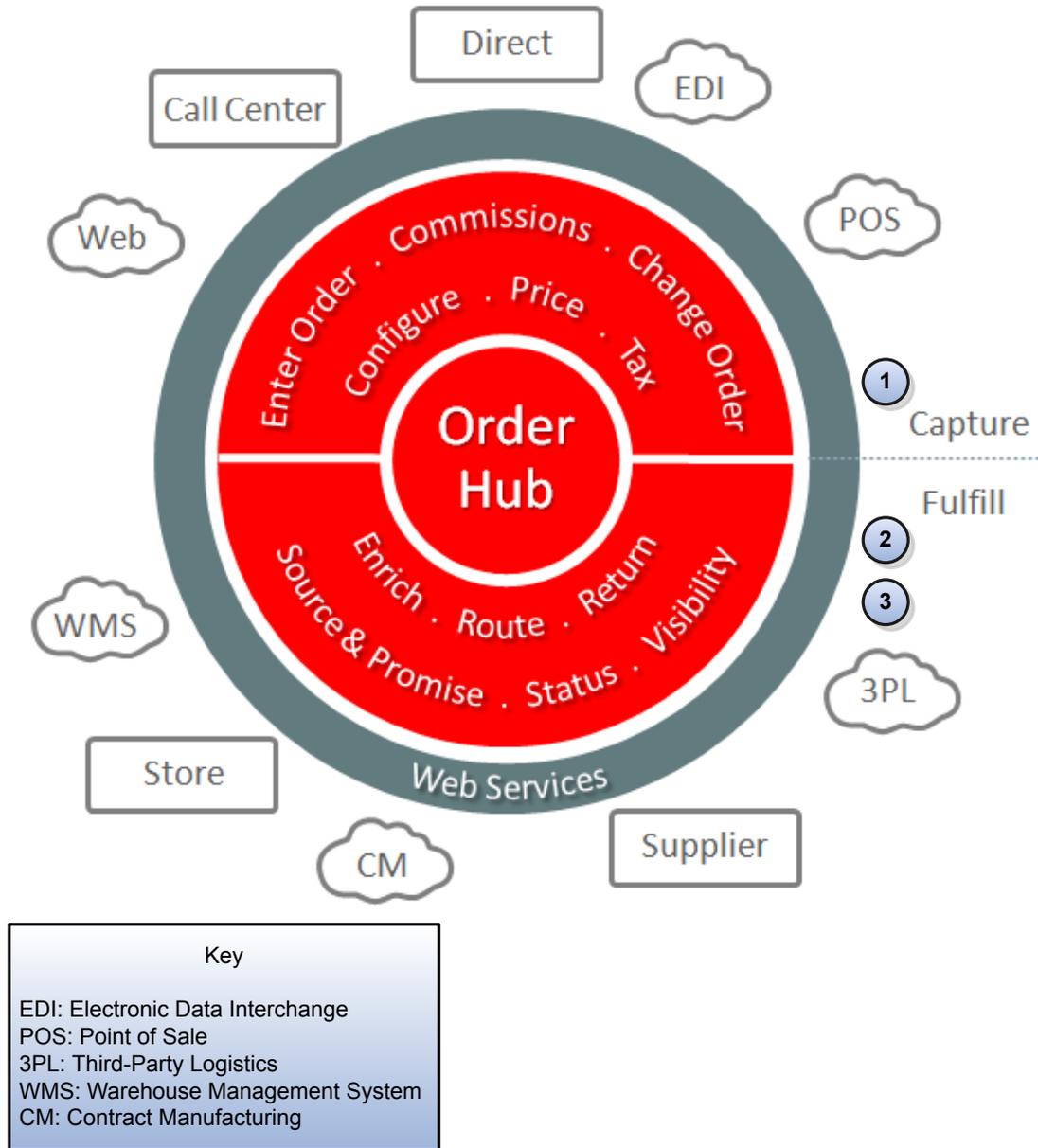
## Oracle Order Management Cloud: Overview

Order Management Cloud is a Supply Chain Management application that improves order fulfillment for your business processes. It includes predefined integration, centrally managed orchestration policies, global availability, and fulfillment monitoring that can help increase customer satisfaction and order profitability.

To get the latest details about Order Management, including What's New in each release and release content documents, see <https://cloud.oracle.com/saasreadiness/scm>.

# Capture Orders and Fulfill Them

The following diagram illustrates how you can use Order Management to capture customer demand and fulfill sales orders.



## Explanation of Callouts

You can use Order Management to capture customer demand and fulfill sales orders in the following ways:

- 1. Capture customer demand.** Capture demand from various channels, such as web, mobile, call center, direct sales, and partners. Provide the functionality that your users can use during order capture, such as pricing items, determining availability, getting the order status, and so on.
- 2. Orchestrate and monitor fulfillment across channels.** Coordinate with other Oracle Supply Chain Management Cloud applications during fulfillment, such as Purchasing, Manufacturing, Inventory Management, and other solutions.
- 3. Use web services.** Integrate Order Management with various systems that reside outside of the Order Management solution. For details, see the Integrating Order Management chapter in the guide titled Oracle SCM Cloud, Implementing Order Management.

## Manage Multichannel Orders

Order Management supports order-to-cash processes that are global and that use multiple channels. You can use it as the order hub for your organization to centralize and manage multiple order capture channels, do order promising, orchestrate fulfillment policies, monitor order status, and manage exceptions. You can enter, price, and configure sales orders directly in Order Management. You can receive source orders from external sources, modify them in Order Management, and then process them for fulfillment. You can use predefined integrations with other Oracle cloud services to centrally manage orchestration policies, get global availability, monitor statuses, and manage exceptions:

- **Enter sales orders.** Enter sales orders directly in Order Management. You can enter details for a new sales order, revise a sales order, modify order lines, view change history, place a sales order on hold, or cancel a sales order. You can also create a return order for an existing sales order.
- **Import source orders.** Import source orders from an external capture system and then edit them. For example, you can import source orders from an e-commerce system, and then modify them in Order Management.
- **Use standard definitions.** Use a standard definition of sales orders across channels so that you can view and search them in a consistent way. For example, a user can search for Item A across sales orders that other users created in Order Management, and across sales orders that an administrator imported from an e-commerce site, call center, or through electronic data interchange (EDI).
- **Use orchestration fulfillment processes.** An orchestration process routes and manages sales orders across multiple fulfillment systems. For example, you can use a schedule, ship, and bill process to route order lines to two different enterprise resource planning (ERP) systems.
- **Get status updates.** Send normalized status updates from multiple fulfillment systems to external capture systems. For example, return a SHIPPED order status to an e-commerce system even if one warehouse system returns a Shipped status for an order line, but another warehouse returns an SHP status for the same line.
- **Get a summary view of statuses and exception orders.** For example, you can view a graph that displays a summary of sales orders that are at risk of missing a promise date.
- **Configure order promising.** Use Oracle Fusion Global Order Promising to collect supply data from multiple sources and to configure rules that select the best fulfillment location for demand from any channel. You can select according to future availability, expected delivery date, and preferred delivery method. Allocate scarce supply according to customer, channel, or to resolve order exceptions.
- **Use heterogeneous deployment.** Use a mixture of cloud and on-premise environments for your capture systems and fulfillment systems. For example, you can import source orders from a cloud capture system, and then fulfill them to an external, on-premise ERP system.

For details about how Order Management integrates these channels with other systems, see Integrating Order Management with External Systems: Overview.

## Monitor Order Progress and Manage Exceptions

Order Management provides a single work area that you can use to view the progress of a sales order, order status, and exceptions that might cause problems in fulfilling a sales order. This work area displays a summary of statuses and exceptions according to customer, item, or supplier, and it allows you to drill into the data to view more details. If a sales order is at risk of not meeting the fulfillment dates, then a jeopardy feature makes them more prominent so that you can identify issues in time to take corrective action. Analytic information provides the data that you require to make informed choices. You can use this work area to monitor the progress of each sales order:

- View order statuses and exception conditions.
- Filter sales orders according to customer, item, fulfillment location, supplier, status, or age.
- Use a Gantt chart that includes timelines to monitor processes.
- Fix exceptions one or more lines at a time.
- Use embedded intelligence to resolve exceptions.

## Simplify Processing for Change Orders

Change order management allows you to define compensation patterns and roll back steps so that you can consistently control change order processing. You can use change order logic to make sure Order Management processes and revises sales orders consistently across all sales orders. For example, if Order Management receives a quantity change from a source order, and if it has not shipped the item, then it uses change logic that allows the change, and then rolls back the fulfillment process so that it can reschedule and send a new ship request to the external fulfillment system. You can use change order management to do the following work:

- Specify the attributes that affect a change order and that automatically recognize a change order.
- Define rules that specify how to automatically handle changes so that a user is not required to intervene in every change that occurs.
- Coordinate change order tasks with external fulfillment systems.
- Specify the tasks in the process that the change order affects.
- Adjust the fulfillment processing steps that the changes in a sales order affects.
- Cancel orders, add lines to orders, and change order quantity.
- Revise existing sales order in Order Management.

## Customize Order Processing

You can define the orchestration process that Order Management uses to fulfill a sales order. You can define a new orchestration process, or you can modify a fulfillment process that comes predefined with Order Management that already meets most of your business requirements. The process definitions allow you to define the following types of order processing that your organization requires:

- Use standard, conditional, parallel, or interrelated flows for order lines.
- Define change order rules that automatically modify order fulfillment.
- Define rules that calculate the order completion date according to the requested date.

- Define common statuses to use across your business process.
- Create draft orders in Order Management that you can later submit for processing.

You can define an orchestration process that plans fulfillment processes, calculates lead times, modifies order compensation steps, and so on. To help you visualize an orchestration process, Order Management displays it as a flowchart that includes steps, lines, decision points, and so on.

## Enrich Source Orders For Fulfillment

You can transform a source order and add information to it that improves fulfillment. For example, you can add or modify attributes or add more items so that Order Management can efficiently fulfill the sales order. You can use a rule editor to define the rules that specify how to transform a source order, how to modify order attributes, or how to define order lines. You can do the following work to determine how Order Management transforms a source order into a sales order:

- Define item transformation rules according to item structure.
- Use fulfillment item relationships to define item transformation rules.
- Define business rules that transform an item, such as product-to-product, product-to-attribute, transactional attribute-to-product, attribute-to-attribute, context-to-product, or context-to-attribute.

## Manage Problems and Recover from Errors

You can do the following work to manage problems and recover from errors:

- Use a jeopardy feature to help you monitor each sales order against a fulfillment date, predict whether or not the sales order is on schedule to meet the fulfillment date and, if not, take corrective action before the sales order misses the fulfillment date.
- Use a central work area to recover sales orders that are in an error state.
- Locate the problem, identify the root cause, and then adjust order fulfillment parameters to fix the problem.
- Modify dates and attributes that affect process planning.
- Query and examine a subset of sales orders according to a criteria that you specify.
- Take action on a single sales order, or on multiple sales orders at the same time.
- Schedule a background process that automatically recovers errors according to parameters and filters that you define.

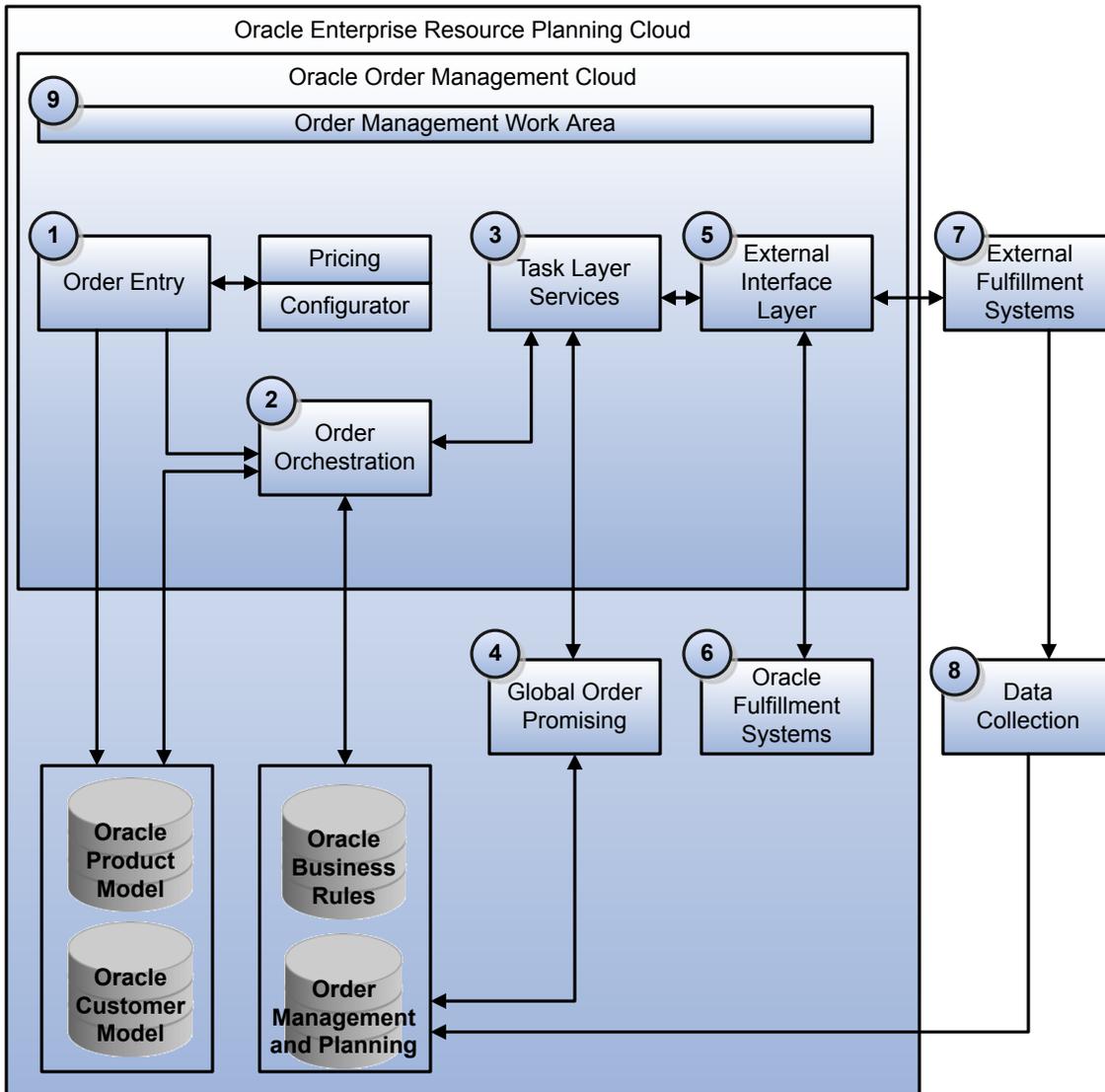
## Order Management Architecture

### Order-to-Cash: How It Works

You can use Order Management Cloud to order and fulfill an item, such as a desktop computer, configured item, or a service, such as monthly maintenance for a network of desktop computers. A user can create a new sales order in the Order Management work area. You can also integrate Order Management with one or more upstream order capture systems, and with one or more downstream fulfillment systems.

You can use various fulfillment techniques, such as drop ship, back-to-back shipping, and internal transfers. For details about some of these fulfillment techniques, see Oracle SCM Cloud, Implementing Manufacturing and Supply Chain Materials Management.

The following diagram illustrates the architecture that Order Management uses when a user creates a new sales order in the Order Management work area. To view a variation of this flow, see Drop Ship in Order Management Cloud: How it Works.



**Explanation of Callouts**

The Order Management architecture uses the following steps when it processes a new sales order. It also uses these steps when a user revises a sales order:

1. A user clicks Create Order in the Order Management work area:
  - o The user searches for a customer in the Customer field of the order header. Order Management gets information about this customer from the Oracle Customer Model, and then displays it in the order header.
  - o The user searches for an item in the Order Lines area of the order. Order Management gets information about this item from the Oracle Product Model, and then displays it on the order line when the user clicks Add.
  - o Oracle Pricing Cloud defines and manages prices for the items that the user enters.
  - o If the user adds a configured item in an order line, then Oracle Fusion Configurator Runtime allows the user to select item options to configure the item. For details, see the guide titled Oracle SCM Cloud, Configurator Modeling Guide.
  - o The user clicks Submit, Order Management validates the order, and then an assignment rule identifies the orchestration process that Order Management must run to fulfill the order. You can configure this rule. For details, see Assigning Orchestration Processes: Explained.

For information about how to get orders into Order Management from an external source system, see Implementing Order-to-Cash: Points to Consider. For information about how to import orders from a spreadsheet, see Using Files to Import Orders into Order Management: Procedure.

2. Order Orchestration uses an orchestration process that identifies and assigns the set of steps that it will use to fulfill the order. Examples of these steps include source, schedule, fulfill, pick, pack, and ship. The steps that it assigns vary so that they optimize order fulfillment according to the specific needs of the order. It determines process logic, such as determining the statuses to use for the order, how to do forward planning and backward planning, how to compensate for modifications to the order, and so on.

Order Orchestration also does the following work:

- o References data in the Oracle Product Model to determine the item data to use for the order. You can specify this item model. For details, see Getting Data from Different Product Models When Using Oracle Business Rules: Explained.
- o References Oracle Business Rules that specify how to orchestrate each order. You can define a variety of rules. For example, you can define a rule that uses a different lead time depending on whether or not the inventory organization resides in Denver or in San Francisco, or a rule that makes sure Order Management does not attempt to ship a nonshippable item, such as a video that your customers can only stream from the cloud. For details, see Defining Lead-Times for Orchestration Process Steps: Procedure.
- o References data that resides in the Order Management and Planning Data Repository to cross-reference and validate data. To collect this data, you define data collection parameters, enable data for collections, and so on. For details, see Managing Planning Source Systems for Data Collections: Explained.
- o Sequences the fulfillment steps that process the order.
- o Calls the task layer services. You can customize how an orchestration process calls a task layer service. For details, see Defining Orchestration Processes: Examples.

For details about how to define a custom orchestration process that meets your business needs, see Defining Orchestration Processes: Explained.

3. Task layer services run the orchestration process steps and manage fulfillment tasks. The Schedule task layer service communicates with Global Order Promising so that it can cross-reference items before it sends the message to downstream fulfillment systems through the External Interface Layer. Task layer services interpret replies and updates from these systems. For example, to send a shipment request to the shipping system, an orchestration process can call the Shipment Task Layer service to create a shipment request. For details, see Task Layer: Explained.

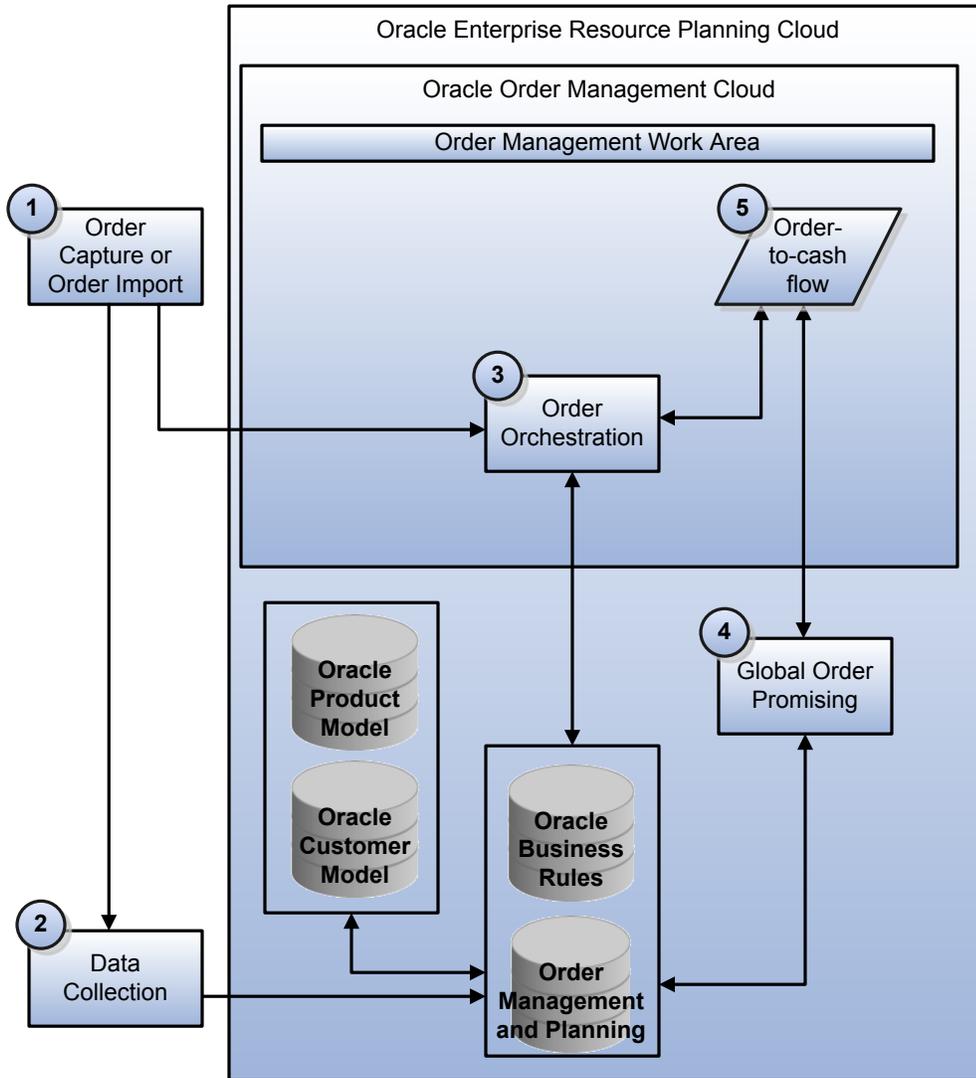
4. An orchestration process interacts with Oracle Global Order Promising Cloud to determine availability and to promise the order. Global Order Promising Cloud does the following work:
  - o Uses supply and demand data from the Order Management and Planning Data Repository to determine availability in the supply chain, such as inventory levels in warehouses.
  - o Schedules the best fulfillment options for each fulfillment line. A user can also manually use the Order Management work area to determine and choose these options.
  - o Communicates with Oracle Shipping to ship the order. For details, see Shipping: How it Works.
  - o Updates the fulfillment status.
  - o Sends order information to order billing.
  - o If the order must reserve the item, then Order Management interacts with Oracle Inventory Management.

Each orchestration process that comes predefined with Order Management includes a scheduling step that uses Global Order Promising. However you can define an orchestration process that does not include a scheduling step and that does not use Global Order Promising. For example, if your organization sells downloadable software or publications that do not require a warehouse or shipping.
5. The External Interface Layer communicates order details between Order Management and each fulfillment system. It primarily routes the fulfillment request and converts the data so that each fulfillment system can correctly use this data.
6. Oracle fulfillment systems do the following work:
  - o Oracle Materials Management Cloud manages logistics and inventory for the order, including schedule, reserve, receive, and ship each item.
  - o Accounts Receivables processes the billing information, including one-time charges and recurring charges, and then sends this information to Oracle Financials Cloud.
  - o Oracle Financial Cloud performs the financial transactions for the order. It creates an invoice, manages accounts receivable, processes payments, and manages revenue.
7. The External Interface Layer sends a request to external fulfillment systems to fulfill the order. These external systems process the request and send completion updates to the External Interface Layer so the orchestration process can move to the next step. The user can use the Order Management work area to specify fulfillment options that affect these fulfillment systems.
8. Order Management Cloud and Global Order Promising Cloud collect data, and then sends it to the Order Management and Planning repository. For example, Global Order Promising collects the following types of data:
  - o **Static.** For example, structures, routing rules, suppliers, transit times, and so on.
  - o **Dynamic.** For example, on-hand inventory, purchase orders, and data from other Cloud solutions.
9. An end user uses the Order Management work area to manage orders during the fulfillment process. For example, to monitor order line status, modify fulfillment options, revise orders, and so on.

## Order-to-Cash with Order Capture Systems: How it Works

You can configure an order capture system that resides outside of Order Management Cloud, such as Oracle Configure, Price and Quote Cloud, to send a source order to Order Management that includes standard items, kits, and pick-to-order configurations. This topic describes the order-to-cash flow that Order Management Cloud uses when it receives source orders from an order capture system that resides upstream of Order Management.

Order Management can transform a source order to a sales order, reply with a confirmation, and then orchestrate fulfillment tasks for this sales order, such as schedule, reserve, ship, and bill. It also updates the fulfillment line status while it does this work, and then sends it periodically to the order capture system.



**Explanation of Callouts**

Order Management uses the same flow with an external order capture system that it uses when the user creates a sales order directly in Order Management, with the following important differences:

1. An order capture system that resides outside of Order Management captures a source order. For example, a user might use a legacy application that your company owns to enter a source order. This capture system then

uses a connector to send the source order to Order Management. You must add this connector. For details, see [Connecting Order Management to Source Systems: Procedure](#).

2. Data Collection collects data from your order capture system. Order Management uses this data to create cross-references for various items, such as warehouses, units of measure, carriers, currencies, shipping methods, payment terms, accounting rules, invoicing rules, service levels, tax classification codes, and so on. For details about how Data Collection uses this data, see [Data Collections, Order Orchestration, and Order Promising: How They Fit Together](#).

The data that you collect into the Order Management and Planning repository references data that the Oracle Customer Model and the Oracle Product Model use. For details about how to set up data collection, see the guide titled [Oracle SCM Cloud, Implementing Supply Chain Planning](#).

3. Order Orchestration uses connector services and references rules in the Oracle Business Rules repository to transform and orchestrate each order:
  - o Performs pretransformation.
  - o Uses transformation rules to transform the source order into a sales order.
  - o Performs posttransformation cleanup.

For details, see [Transforming Source Orders to Sales Orders: How It Works](#).

4. Global Order Promising uses data in the Order Management and Planning repository that you populated from your order capture system to do promising.
5. Order Management uses the order-to-cash flow to continue, with the difference occurring primarily during order entry and order transformation. No transformation occurs when the user enters a sales order directly in the Order Management work area. For details, see [Order-to-Cash: How it Works](#).

Order Management communicates order status and invoice status to your order capture system throughout processing.

A single source order might contain order lines that include one-time charges, and other order lines that include recurring charges. If Order Management receives a source order that includes a recurring charge, then Oracle Receivables creates a recurring billing invoice for these lines.

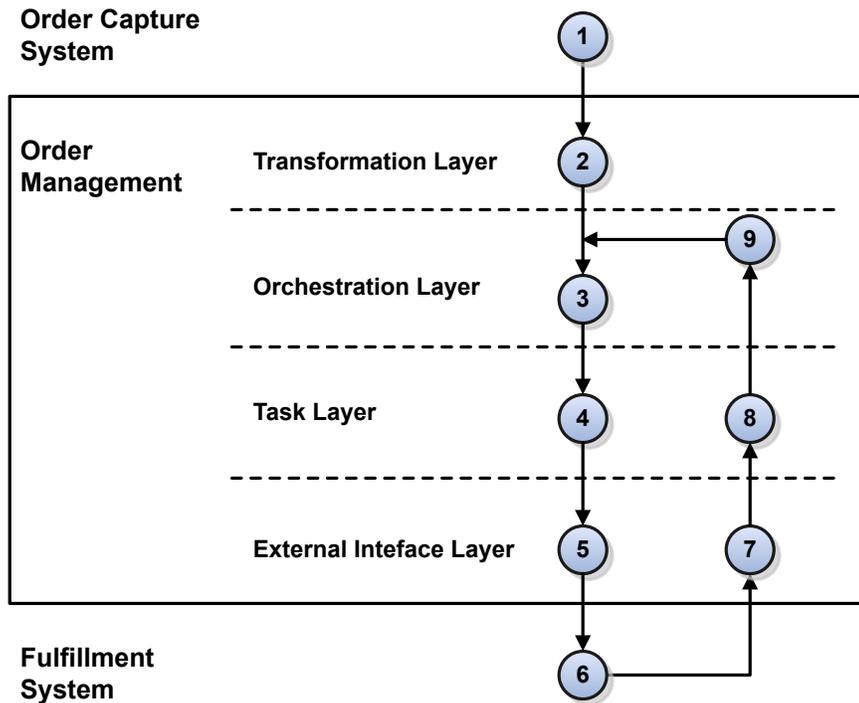
Order import typically uses the same flow that Order Management uses when it gets source orders from an external source system, except you import source orders from a spreadsheet. For details, see [Using Files to Import Orders into Order Management: Procedure](#).

#### *Related Topics*

- [Using Files to Import Orders into Order Management: Procedure](#)

## Architectural Layers in Order Management: How They Work

The following diagram illustrates the flow through architectural layers that Order Management Cloud uses when it fulfills a sales order.



### Explanation of Callouts

Order Management uses the following steps when it fulfills a sales order that it receives from an order capture system. The flow for a sales order that a user enters directly in Order Management is similar, except it starts primarily at the Orchestration Layer in step 3:

1. An external order capture system sends a source order to Order Management.
2. The transformation layer transforms the source order to a sales order, and then assigns an orchestration process to this sales order.
3. The orchestration layer runs the fulfillment step, and then sends it to the task layer. Each fulfillment request starts here.
4. The task layer sends the request to the external interface layer. This layer includes services that can send requests for fulfillment tasks to fulfillment systems. Order Management provides services for some typical fulfillment system functions, such as shipping and invoicing. You can use the Template task layer services to add functionality. For details, see [Task Layer: Explained](#).
5. The external interface layer does the following work:
  - o Cross-references the data that the request contains.
  - o Uses routing rules to identify the fulfillment system to use for fulfillment lines. For details about these rules and how to create them, see [External Interface Routing Rules: Explained](#).

- A connector sends the request to the fulfillment system.
- 6. The fulfillment system accepts the request, and then sends a reply.
- 7. The external interface layer converts the reply from the fulfillment system.
- 8. The task layer processes the reply from the fulfillment system.
- 9. The orchestration layer runs the next fulfillment step.

## External Interface Layer

The external interface layer manages communication between Order Management and an external fulfillment system. Note the following:

- The fulfillment system does not communicate directly with the orchestration process. Instead, the external interface layer provides an intermediary that Order Management uses to route requests.
- Order Management uses a web service that routes requests from the fulfillment task layer to the fulfillment system, and from the fulfillment system to the fulfillment task layer.
- The connector modifies the structure and content of the outbound message so that it matches the inbound interface that the fulfillment system uses.
- You configure web services, configure routing rules, and then register the connector web service. This work integrates the fulfillment system with the external interface layer.

The external interface layer provides the following benefits:

- Keeps the external system separate from the orchestration process to minimize the modifications that you must make when you add a new order capture system or fulfillment system. This way, it is not necessary to modify the orchestration process every time you integrate a fulfillment system.
- Provides a flexible integration that you can use in a Service Oriented Architecture (SOA) with an external system or application.
- Provides a complete, open, and integrated solution that lowers cost of ownership.

### *Related Topics*

- [Task Layer: Explained](#)
- [Routing Requests to Fulfillment Systems: Explained](#)
- [Connecting Order Management to Fulfillment Systems: Procedure](#)

# 2 Implementing Order Management Cloud

## Implementing Order-to-Cash

### Implementing Order-to-Cash: Points to Consider

It is recommended that you identify the features that you must implement to support the order-to-cash flow in your business environment, and then estimate the effort required to set them up.

#### Consider Some of the Features You Must Set Up

Perform the preparation work described in *Implementing Order-to-Cash: Roadmap*. This work will provide a solid understanding of the features that you need to implement, and the work involved in setting them up.

#### Estimate Effort Required to Set Up Features

Use the following table to create a rough estimate of the amount of effort that your implementation will require. For example, if your implementation will require the configurator and order import, then include the setup effort for them in your project plans, and consider whether or not you must administer the job roles that support them.

Requirement	Comes Predefined	Setup Required
Dashboard	Yes	No
Create Order	Yes	Yes
Revise Order	Yes	No
Return Order	Yes	Yes
Search and View Orders	Yes	No
Configurator	No	Yes
Order Import	No	Yes
Order Transformation	No	Yes
Custom Order Status Behavior	No	Yes
Custom Constraint Behavior	No	Yes
Custom Change Order Behavior	No	Yes
Custom Notifications	No	Yes

Requirement	Comes Predefined	Setup Required
Custom Jeopardy Setup	No	Yes
Custom Fields	No	Yes
Custom Orchestration Process Behavior	No	Yes
Receive and Process Orders from Trading Partners	No	Yes
Send Advanced Shipment Notice	No	Yes
Collaboration Messaging Framework	No	

## Determine to Modify the Predefined Integration or Create a Custom One

### Modifying the Predefined Implementation

Order Management comes predefined with most of the setup tasks already completed for you. You can modify this predefined implementation.

To get started, do the following work:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, notice the functional areas that display above the Orders functional area, such as Initial Users, Enterprise Profile, and so on. These functional areas are common areas. For details about how to set them up, see the guide titled Oracle SCM Cloud, Implementing Common Features for Oracle SCM Cloud.
4. In the Functional Areas list, click **Orders**.
5. In the Orders area, click **Required Tasks**, and then click **All Tasks**.
6. In the Orders list, drill into and complete each task as necessary, depending on your business requirements. If you find that modifying the predefined implementation does not meet your business requirements, then create a implementation.

### Creating a Custom Implementation

It is recommended that you modify the predefined implementation. However, if the predefined implementation does meet your business requirements for some reason, then you can create a new, custom implementation.

To get started, do the following work:

1. In the **Navigator**, click **Setup and Maintenance**, click **Getting Started**, and then click **Order Management**. Read the documents to get detailed information about the following items that you can configure and the work you must perform during implementation:
  - o Reports that are available for each offering
  - o Lists of setup tasks that you must perform
  - o Descriptions of the options and features that you can choose when you configure Order Management
  - o Lists of business objects and enterprise applications that you can use with Order Management

2. In the Setup and Maintenance work area, click **Implementation Projects**.
3. On the Implementation Projects page, click **Actions**, and then click **Create**.
4. On the Create Implementation Project page, modify the field values, as necessary, and then click **Next**.
5. On the Select Offerings to Implement page, in the Name list, expand **Order Management**.
6. Add a check mark in the Include column for the Order Management row and for the Pricing row, and then click **Save and Open Project**.

You can create one or more implementation projects for the offerings and options that you must implement. Each Oracle application creates the task list that you must complete for each project that you create. An Application Implementation Manager can customize these task lists, and can assign and track each task that these lists contain.

7. On the Implementation Project page, in the Task list, expand **Order Management**, and then perform the tasks that your business flow requires.

## Implementing Order-to-Cash: Roadmap

This road map provides a high-level procedure that you can use to implement order-to-cash. You perform the common setup tasks, set up integration, set up parameters, and so on.

Note the following:

- It is recommended that you perform the tasks that this topic references in the same sequence that this topic displays them.
- Steps after step 3 in this topic primarily reference tasks that reside in the Orders functional area. Complete each of these tasks, as necessary, depending on your business requirements.
- To perform a task, click each task in the Orders area. If the Orders area does not display a task, then search for it in the Search Tasks field.
- The work you must perform depends on your business requirements. For example, a typical implementation requires that you perform most or all of the tasks that integrate Order Management. However, whether or not you perform other tasks depend on your business requirements, such as tasks that control order status, constrain changes, or that customize orchestration processes.
- The Topic column references topics that reside in the guide titled Oracle SCM Cloud, Implementing Order Management. It is recommended that you use these topics when you perform the task that they describe.
- The Prepare column describes work that you can do to prepare for the implementation. It is recommended that you perform this work before you administer order-to-cash. Doing so will help to avoid interruption and downtime during administration, and will help to make sure it goes smoothly. You can also use the Prepare column to check off preparation work as you complete it.
- To help monitor your progress, add a check mark to the Done column when you finish each functional area or task.
- To implement drop ship, see the Implementing Drop Ship section in the Implementing Order Management guide.

To implement order-to-cash, do the following work:

1. Read the Implementing Order-to-Cash: Points to Consider topic. It contains information that help to make the implementation go smoothly.
2. In the Navigator, click **Setup and Maintenance**.
3. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.

This topic assumes that you will modify the predefined implementation that is available through Setup. For details about using a custom implementation, see Implementing Order-to-Cash: Points to Consider.

4. On the Setup page, in the Functional Areas list, complete the following functional areas that display above the Orders functional area. These areas reference common tasks. A typical implementation requires that you complete most

or all common tasks. For details about how to set them up, see the guide titled Oracle SCM Cloud, Implementing Common Features for Oracle SCM Cloud.

Functional Area	Done
Initial Users	_____
Enterprise Profile	_____
Organization Structures	_____
Users and Security	_____
Items	_____
Catalogs	_____
Customers	_____

5. In the Functional Areas list, click **Orders**.
6. In the Orders area, click **Required Tasks**, and then click **All Tasks**.
7. Integrate Order Management. Complete the following tasks. For details, see the Integrating Order Management chapter in the Implementing Order Management guide.

Task	Topic	Prepare
Manage Web Service Details	Creating User Credential Keys: Procedure	_____ Gained access to the administrator privilege and administrator role so that we can use Oracle Wallet Manager.
Manage Trading Community Source Systems	Connecting Order Management to Source Systems: Procedure	_____ Located the time zone where the server that the source system uses is located.
Manage Upstream and Fulfillment Source Systems		_ _____ Identified the URL that locates the connector service that resides on the source system.
Manage External Interface Web Service Details		_ _____ Identified the User Name and Password that the Status Update service requires.
Manage External Interface Web Service Details	Connecting Order Management to Fulfillment Systems: Procedure	_ _____ Identified an integrated development environment we can use to create a transformation style sheet.  _____ Acquired the user credentials that the service provider requires when calling their web service.  _____ Acquired contact information for the IT administrator who works for the service provider.

Task	Topic	Prepare
		<p>_____ Identified the URL that locates the web service that resides on the fulfillment system.</p> <p>_____ Acquired the security certificate from the service provider.</p>
Manage External Interface Routing Rules	Routing Requests to Fulfillment Systems: Procedure	_____ Created a list that includes the names of the items that we must route to fulfillment systems, the unique identifier for each item, and the name of each fulfillment system where Order Management must route the request.

8. Control application behavior. Complete the following tasks. For details, see the Controlling Application Behavior chapter in the Implementing Order Management guide.

Task	Topic	Prepare
Manage Order Management Parameters	Managing Order Management Parameters: Procedure	<p>_____ Determined whether or not our business requirements will allow or disallow the Configurator to choose items in a configuration and to modify a configuration after the user adds a configured item.</p> <p>_____ Determined the date that Order Management must use when it determines the components that it displays for a configured item.</p> <p>_____ Determined how Order Management will display the Ship-to Address and the Bill-to Location for each customer during order entry.</p> <p>_____ Determined whether or not the Configurator stops processing when it encounters an error.</p> <p>_____ Identified the item validation organization that Order Management must use to validate and display items.</p> <p>_____ Identified an Order Management user that the buyer can contact to resolve a problem that occurs with a drop ship.</p> <p>_____ Determined whether or not the Configurator validates each order that includes a configured item during order import.</p>
Manage Order Profiles	Managing Order Profiles: Procedure	<p>_____ Identified the value to use when converting a currency.</p> <p>_____ Identified the currency to display in the Order Management work area.</p> <p>_____ Identified the customer to use when filtering status data on the Overview page.</p>

Task	Topic	Prepare
		<p>____ Determined whether or not to use the source order number as the order number during transformation.</p> <p>____ Determined the number of seconds to wait after an action finishes.</p>

9. Administer pricing. For details, see the guide titled Oracle SCM Cloud, Administering Pricing.
10. Import source orders. If your implementation includes a source system, then you must import source orders. Complete the following tasks. For details, see the Importing Orders section in the Implementing Order Management guide.

Task	Topic	Prepare
Not applicable	Using Files to Import Orders into Order Management Cloud: Procedure	<p>____ Acquired access to our source order data.</p> <p>____ Identified a tool we can use to manipulate source order data, such as SQL (Structured Query Language), or ODI (Oracle Data Integrator).</p>
Create Source System	Using Web Services to Import Source Orders: Explained	____ Determined whether or not our implementation will use the product model or the product hub.
Manage Upstream and Fulfillment Source Systems		____ Identified the attributes that we will use in the Request Payload.
Manage Standard Lookups		
Manage Item Relationships		

11. Transform source orders. If your implementation includes a source system, then you must transform source orders. Complete the following tasks. For details, see the Transforming Source Orders to Sales Orders section in the Implementing Order Management guide.

Task	Topic	Prepare
Manage Product Transformation Rules	Setting Up Transformation: Procedure	Acquired login user and password to the Oracle Product Model.
Manage Product Transformation Rules	Creating Transformation Rules: Procedure	<p>____ Described the structure of the source order data.</p> <p>____ Identified the type of transformation that must occur so that Order Management can support our source data.</p> <p>____ Described business requirements for pretransformation rules.</p> <p>____ Described business Requirements for transformation rules.</p> <p>____ Described business requirements for posttransformation rules.</p>

Task	Topic	Prepare
Manage Product Transformation Rules	Creating Advanced Product Transformation Rules: Procedure	____ Described business requirements for advanced posttransformation rules, if necessary.

- 12.** Customize statuses. For details, see the Controlling Order Statuses section in the Implementing Order Management guide.

Task	Topic	Prepare
Manage Task Status Conditions	Managing Task Status Conditions: Procedure	____ Described our business requirements for managing the statuses that our fulfillment systems provide.
Manage Status Values	Customizing Fulfillment Lines Statuses: Procedure	____ Described our business requirements for managing the statuses for each fulfillment line.
Manage Orchestration Process Definitions	Adding Status Conditions to Fulfillment Lines: Procedure	
Manage Status Values	Creating Orchestration Process Classes: Procedure	____ Described our business requirements for grouping orchestration process statuses so that they are meaningful.
Manage Orchestration Process Definitions	Adding Status Conditions to Orchestration Processes: Procedure	____ Described our business requirements that specify when to set statuses for orchestration processes.

- 13.** Customize processing constraints. For details, see the Constraining Orders section in the Implementing Order Management guide.

Task	Topic	Prepare
Manage Processing Constraints	Managing Processing Constraints: Procedure	____ Described business requirements regarding who can change a sales order, what can change in this sales order, and when the change can occur.
Manage Constraint Entities	Constraining Changes to Attributes: Procedure	____ Described business requirements regarding the order attributes that can change.

- 14.** Control change orders. For details, see the Managing Change Orders section in the Implementing Order Management guide.

Task	Topic	Prepare
Manage Order Attributes That Identify Change	Managing Order Attributes That Identify Change: Procedure	____ Described business requirements regarding what changes Order Management will allow on an existing sales order, and when it allows the change, including order attributes, users, and timing.

Task	Topic	Prepare
Manage Orchestration Process Definitions	Measuring the Cost of Change: Procedure	___ Estimated the business cost associated with each change.
Manage Orchestration Process Definitions	Compensating Orders That Change: Procedure	___ Described business requirements regarding the order compensation that we will allow Order Management to perform.

- 15.** Notify systems when orders change. For details, see the Notifying Systems When Orders Change section in the Implementing Order Management guide.

Task	Topic	Prepare
Manage Business Event Trigger Points	Sending Notifications from Order Management Cloud to External Systems: Procedure	___ Described business requirements regarding when Order Management must notify external systems that a sales order changed.

- 16.** Use jeopardy to manage delays. For details, see the Using Jeopardy to Manage Delays section in the Implementing Order Management guide.

Task	Topic	Prepare
Manage Jeopardy Priorities	Defining Jeopardy and Lead Time to Manage Delay: Procedure	___ Described business requirements regarding when Order Management must notify an order manager that a sales order is in jeopardy, including the threshold at which to notify for each step in an orchestration process.
Manage Orchestration Process Definitions		
Manage Jeopardy Thresholds		

- 17.** Manage task types. For details, see the Customizing Task Types section in the Implementing Order Management guide.

Task	Topic	Prepare
Manage Task Types	Creating Custom Task Types: Procedure	___ Described business requirements regarding custom task types that our implementation might require.

- 18.** Customize fields. For details, see the Customizing Fields section in the Implementing Order Management guide.

Task	Topic	Prepare
Manage Order Extensible Flexfields	Deploying Extensible Flexfields in Order Management: Procedure	___ Described business requirements regarding what information we must display in custom fields, including the data and location in the user interface.

Task	Topic	Prepare
------	-------	---------

19. Customize orchestration processes. For details, see the Customizing Orchestration Processes chapter in the Implementing Order Management guide.

Task	Topic	Prepare
Manage Orchestration Process Definitions	Defining Orchestration Processes: Points to Consider	<p>___ Confirmed that the predefined orchestration processes do not meet our business requirements.</p> <p>___ Identified configuration details for each custom orchestration process that we must create, such as task type to use, change logic, process planning, jeopardy, statuses, and so on.</p>
Manage Orchestration Process Definitions	Defining Orchestration Processes: Procedure	<p>___ Described the IF/THEN rules that our business requires for each custom orchestration process.</p> <p>___ Described orchestration planning and replanning that our business requires for each custom orchestration process.</p> <p>___ Specified the behavior that we require for each orchestration process step, such as the task type to use, the service to call, lead-times, and so on.</p> <p>___ Created a flowchart mock-up for each custom orchestration process.</p>
Manage Orchestration Process Definitions	Deploying Orchestration Processes: Procedure	Not applicable.
Manage Orchestration Process Definitions	Defining Lead-Times for Orchestration Process Steps: Procedure	___ Described the IF/THEN rules and conditions that determine the lead-time to use for each orchestration process step.
Manage Orchestration Process Definitions	Selecting the Fulfillment Lines That Run for Orchestration Process Steps: Procedure	___ Described the IF/THEN rules that select the fulfillment line to run for each orchestration process step.
Manage Orchestration Process Definitions	Adding Branches to Orchestration Processes: Procedure	___ Specified subprocesses and parallel processing that our custom orchestration process requires.
Manage Orchestration Process Definitions	Pausing Orchestration Processes: Explained	___ Specified the IF/Then rules that Order Management must use when it pauses each of our custom orchestration processes and each custom orchestration process step.
Manage Orchestration Process Assignment Rules	Assigning Orchestration Processes: Procedure	___ Specified the IF/Then rules that Order Management must use when it assigns each of our custom orchestration processes

Task	Topic	Prepare
Not applicable	Automatically Resuming Paused Orchestration Processes: Explained	and each custom orchestration process step.  ____ Identified the pause tasks that we will release to resume a paused orchestration process.

*Related Topics*

- [Oracle Order Management Cloud: Overview](#)

## Implementing Order-to-Cash: Quick Start

This topic describes the minimum steps that you must perform to implement Oracle Order Management Cloud if you do not use the all the tasks that are available in the Order Management offering. You can use it to set up a test instance of Order Management.

**⚠ Warning:** This topic does not include all the steps and security tasks that are required to fully implement Order Management. It includes only the minimum implementation steps that you must perform so that your implementation can receive and fulfill test orders.

For details about how to use the predefined implementation, see [Implementing Order-to-Cash: Points to Consider](#).

### Summary of the Work

To get started with implementing Order Management, do the following work:

1. Prepare to start your implementation.
2. Configure common components.
3. Set up the product model and import items.
4. Define source systems and import customers.
5. Collect data.
6. Configure Global Order Promising.
7. Set up Order Management and test your work.

This topic references several guides, such as the Oracle Fusion Product Information Management Implementation guide. To access each guide, navigate to the Technology Library for Oracle Cloud Applications at <http://www.oracle.com/technetwork/documentation/default-1577155.html>.

## Prepare to Start Your Implementation

Prepare to start your implementation:

1. Get access to the Security Console and Oracle Cloud Applications.
2. Get the URLs that locate Oracle Cloud Applications and the Security Console.

For example, the URL for Oracle Cloud Applications is:

`http://host/homePage/faces/AtkHomePageWelcome`

where:

**host** is the name of the host location, such as abc.oracleoutsourcing.com.

For example:

**`http://abc.oracleoutsourcing.com/homePage/faces/AtkHomePageWelcome`**

If you do not know the URLs, contact the person who installed the systems at your company. The summary page displays these URLs when the provisioning process that Oracle Cloud Applications uses finishes. The default file name of the file that stores these URLs is Provisioning Summary. The person who installs the software can choose a different file name. This file resides in the same folder where the provisioning plan resides.

3. Get the user name and password of the Oracle Cloud Applications super user and the Security Console system administrator.

The default user name of the Oracle Cloud Applications super user is FAADMIN. Contact the person who installed the systems to get the user names and passwords that they used or specified when they installed and provisioned the application.

4. Determine the following for your test orders:
  - o Identify the items that the test orders will contain and the customers who will order them.
  - o Identify the item organizations that you will associate with these items.
  - o Identify the units of measure (UOMs) and currencies that these test orders will use.

## Configure Common Components

Perform the implementation steps that are common to an Oracle application. For details about how to perform each of the following steps, see the table that the step mentions in Article ID 1387777.1 (Getting Started with Oracle Fusion Applications: Common Implementation) on My Oracle Support at <https://support.oracle.com>. For example, for step 1, see Table 1 in Getting Started with Oracle Fusion Applications. For step 2, see Table 2 in Getting Started with Oracle Fusion Applications:

Configure common components:

1. Prepare the super user.
2. Prepare the IT Security Manager Role.
3. Create the setup task list for the Order Management offering.

This offering includes the tasks that you perform.

4. Define users.
5. Define the basic enterprise structures.

Note the following:

- o You can use the default enterprise structure for a pilot implementation.
  - o The Order Management work area displays sales orders for the business units that the current user can access. You must create a separate business unit for each business unit that will receive sales orders.
  - o A set is a collection of business units. Order Management uses sets to restrict access to holds and orchestration processes. You must specify a default set when you create a business unit. You can use the predefined Common Set for the default set.
6. Define the users who perform functional testing.

An implementation user can access a wide range of privileges. To test with users who have fewer privileges, you can define users with the following roles that are specific to Order Management:

Role	Name
Job	Order Entry Specialist
Job	Order Manager
Job	Order Administrator
Job	Supply Chain Application Administrator
Abstract	Error Recovery

To examine how these roles implement security, you can create at least one user for each of the following roles:

- o Only the Order Manager role
- o The Order Manager role and the Error Recovery role
- o The Order Administrator role and the Supply Chain Application Administrator role

The Supply Chain Application Administrator role provides the Order Administrator role.

7. In the Navigator, click **Setup and Maintenance**.
8. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
9. On the Setup page, click **Organization Structures**, search for, and then use the Manage Item Organizations page to configure your item organizations so that the Oracle Product Model can use them.

Note the following points:

- o You will need at least one organization that represents a warehouse where your implementation can collect the supply data that it uses to fulfill each order.
- o You must configure each warehouse from a fulfillment system as an item organization in Oracle.
- o You must not associate an item organization with a business unit.

For details, open the Oracle Fusion, Product Information Management Implementation Guide, and then see Define Enterprise Structures, Define Facilities, Manage Item Organizations.

## Set Up the Product Model and Import Items

The Oracle Fusion Product Model must contain the items that each test order references.

Set up the product model and import items:

1. On the Setup page, click **Items**, search for, and then use the Manage Units of Measure page to configure your item organizations so that the Oracle Fusion Product Model can use them.

For details, see the Manage Units of Measure section in the Oracle Fusion, Product Information Management Implementation guide.

2. Set up the Oracle Product Model.

For details, see Define Product Information Management section in the Oracle Fusion, Product Information Management Implementation guide.

3. Define items, catalogs, and tool kits.
4. Use an order import template to import the items that your test orders will reference. Transform orders, as necessary.

For details, see the chapter that describes Importing and Transforming Source Orders in the Implementing Order Management Guide.

## Connect Source Systems and Import Customers

Connect the source system and import customers:

1. Connect the source system. For details, see Connecting Order Management to Source Systems: Procedure.
2. Import customers:
  - o On the Setup page, click **Customers**, search for, and then open the Import Person and Organization page.
  - o On the Manage Data Import Batches page, click **Actions**, and then click **Create** to create an import batch.
  - o Use an Extract, Transform, and Load tool to load your data into the interface tables.

You can use this import process to import a batch from the interface tables into the Trading Community Model registry. The batch you import must include the customers that your orders will reference. For details, see Defining the Import Process for Customers and Consumers: Points to Consider, and Defining the Import Process for Customers and Consumers: Worked Example.

## Collect Data

Order Management references units of measure, currency, and currency conversions from the Order Orchestration and Planning Data repository. You must perform the tasks in this table to collect data for these entities so that your implementation can receive source orders. Global Order Promising sourcing rules reference the Organization Parameter entity. You must collect Organization Parameters before you define sourcing rules.

Collect data:

1. Manage upstream and fulfillment source systems:
  - o In the Navigator, click **Setup and Maintenance**.
  - o On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
  - o On the Setup page, search for, and then open Manage Upstream and Fulfillment Source Systems.
  - o On the Manage Upstream and Fulfillment Source Systems page, in the Source Systems list, locate your Oracle system, and then set the following values for it.

Field	Value
Version	Fusion
Collections Allowed	Contains a check mark
Order Orchestration Type	Order Orchestration
Enable Data Cross-Reference	Contains a check mark

- o Set the following values for each of the systems that your implementation must integrate, such as an order capture system or fulfillment system.

Field	Value
Version	Others
Collections Allowed	Contains a check mark
Enable Data Cross-Reference	Contains a check mark

For details, see *Managing Planning Source Systems for Data Collections: Explained*.

2. Allow the currencies and units of measure to cross-reference data. On the Setup page, search for, and then open the Manage Upstream and Fulfillment Source Systems page.

For details, see *Collecting Planning Data: Explained*.

3. Collect the organization parameters from your external source system.

To identify your organization, the collected data concatenates the source system to the organization ID. Other entities, such as Sourcing Rules, also use this identification. To collect this data, you load it into staging tables, and then use a scheduled process named Perform Data Load from Staging Tables.

4. Collect units of measure, currencies, and currency conversions from your order capture system.

Note the following points:

- o Order Management validates the units measures and currencies against data that it already collected from your order capture system when it receives source orders from this system.
- o You must collect calendars from your fulfillment system so that Order Management can use them during scheduling. A calendar specifies when a facility, such as a warehouse, is open or closed.
- o To collect this data, you load it into staging tables, and then use a scheduled process named Perform Data Load from Staging Tables.
- o Use the review pages to verify the data that you collect.
- o A pilot implementation expects your test orders to use the same values for the unit of measure, currency, and currency conversion that you collect into the data repository. During a full implementation, if multiple order capture systems use different values for these entities, then you must perform more implementation steps. For example, if one order capture system uses Ea for each item, and if another order capture system uses Each for each item, then you must perform more steps to implement the data cross-referencing that these order capture systems require.
- o Your implementation can collect reference data from multiple systems. However, a reference data entity is a global object, so Order Management and Global Order Promising uses the most recent data that your implementation collects. It is recommended that you identify the source system that contains the master data list, and then collect data from this source system after you collect data from all other source systems. For example, if source system B contains the master list of currencies, then collect currencies from source system B so that Order Management can cross-reference currencies to system B, and then collect currencies from system A.
- o If Order Promising must consider the transit time that occurs during shipping between the warehouse and a destination, then you must collect shipping methods. If you do not collect them, then Order Management uses a transit time of zero days.

For details, see *Data Collections, Order Orchestration, and Order Promising: How They Fit Together*.

## Configure Global Order Promising

Oracle Fusion Global Order Promising uses sourcing rules, assignment sets, and ATP Rules when it determines availability.

Configure Global Order Promising:

1. In the Navigator, click **Order Promising**, click **Tasks**, click **Manage Sourcing Rules**, and then create a global sourcing rule that specifies the warehouse that Order Management must use when it fulfills a sales order.

For details, see Sourcing Rules and Bills of Distribution: Explained.

2. On the Manage Sourcing Rules page, click **Tasks**, click **Manage Assignment Sets**, and then create an assignment set that assigns the sourcing rule that you created in step 1 to the Global assignment level.

For details, see Sourcing Assignment Levels: Explained.

3. Set the administrator profile value:
  - o Navigate to the Manage Administrator Profile Values page, and then, in the Application field, select **Global Order Promising**, and click **Search**.
  - o In the Search results list, set the following value for the MSP\_DEFAULT\_ASSIGNMENT\_SET profile.

Field	Description
Profile Value	Choose the sourcing rule that you created in step 1.

4. Create an ATP rule:
  - o In the Navigator, click **Order Promising**, click **Tasks**, click **Manage ATP Rules**, and then create an ATP rule that uses the Infinite-Availability-Based promising mode.
  - o Assign the ATP rule to each of the organizations that you created for your fulfillment warehouses.

For details, see ATP Rule Promising Modes: Explained.

## Set up Order Management and Test Your Work

You must set up connections to any order capture systems and fulfillment systems that your implementation uses, and then deploy the predefined data that specifies how Order Management fulfills each order.

Set up Order Management, and then test your work:

1. Set up system wide parameters for Order Management. For details, see Managing Order Management Parameters: Procedure.
2. In the Navigator, click **Setup and Maintenance**.
3. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
4. On the Setup page, click **Orders**, search for and open Manage Order Profiles, and then define values for the following profile options:
  - o Display Currency
  - o Currency Conversion Type

For details, see Setting Orchestration Profile Options: Procedure.

5. Define and deploy the following predefined orchestration processes:
  - o ShipOrderGenericProcess

- o ReturnOrderGenericProcess

For details, see Defining Orchestration Processes: Procedure.

6. Click **Save and Close**.

7. Apply the required, predefined constraints:

- o On the Setup page, search for, and then open the Manage Processing Constraints page.
- o On the Manage Processing Constraints page, click **Validation Rule Sets**, click **Generate Packages**, wait a moment, and then click **OK** in the confirmation dialog.

For details, see Managing Processing Constraints: Procedure.

8. Optional. Define the custom orchestration that your implementation requires:

- o Define and deploy custom orchestration processes.
- o Define status codes.
- o Manage change orders.
- o Set up jeopardy and planning.
- o Release and deploy orchestration processes.
- o Create custom processing constraints.

For details, see the chapter that describes Defining Orchestration in the Implementing Order Management guide.

9. Use a tool of your choice, such as SOAP (Simple Object Access Protocol), to simulate order processing.

10. Update the Global Order Promising data:

- o In the Navigator, click **Scheduled Processes**.
- o On the Scheduled Processes page, click **Actions**, and then click **Schedule New Process**.
- o In the Schedule New Process dialog, set the following field, and then click **OK**.

Field	Value
Name	Refresh and Start the Order Promising Server

- o In the Process Details dialog, add a check mark to every parameter, and then click **Close**. Internal Transfer Order is an example of a parameter. You must add a check mark to every parameter even if you have not collected data for them.

For details, see Refreshing the Global Order Promising Engine: Explained.

11. Test your work:

- o Use a tool of your choice, such as SOAP, to simulate sending a test order. Note the order number.
- o In the Order Management work area, use the source order number to search for the order.
- o Confirm that Order Management received the order and is processing it.

### Related Topics

- [Data Collections, Order Orchestration, and Order Promising: How They Fit Together](#)
- [Security Console: Overview](#)

## Enabling Offerings

### Enabling Offerings: Explained

When planning your implementation, you decide what business processes your organization or company performs or supports. These decisions determine the offerings and functional areas you want to implement. You then configure the offerings and functional areas that support the activities your organization or company performs. During the configuration process, you specifically enable offerings and functional areas for use before you implement them.

#### Enabling Offerings and Functional Areas

Use the Setup and Maintenance work area to help decide which offerings to enable for implementation. Once you decide to use an offering, you can select the Configure button to choose the configuration details and enable the offering, associated functional areas, and features. All the base functional areas of an offering are automatically enabled for implementation when you enable the parent offering. You choose which optional functional areas to enable. The functional areas appear in an expandable and collapsible hierarchy to facilitate progressive decision making for implementation.

#### Enabling Features

Features are optional or alternative business rules or methods used to fine-tune business processes and activities supported by an offering or a functional area. If features are available for the offering or functional areas, you can enable them to help meet your business requirements, if desired. In general, the features are set with a default configuration based on their typical usage in most implementations. You should always review the available features for the offering and functional areas and select them as appropriate. Dependent features appear visible when the feature choice they depend on is selected for implementation.

### Enabling Offerings: Procedure

You enable offerings to customize the functionality that matches the services you plan on implementing.

#### Enabling Offerings

To enable offerings, follow these steps.

1. Open the Setup and Maintenance work area (**Navigator > Setup and Maintenance**).
2. In the Setup and Maintenance Offerings page, select the offering you're using, then click **Configure**.
3. In the Configure page, select the **Enable** check box for the offering. Also select the **Enable** check box for each of the functional areas you want to use.
4. Click the Features icon for the offering or functional area you have enabled, then enable any features you require. Select **Done** when complete.
5. Select **Done** to return to the Offerings page then repeat the same steps for each of the offerings you are using.

## Implementing Business-to-Business Order Management

## Business-to-Business Order Management: Overview

Order Management Cloud provides a central order hub in a multichannel environment that you can use to improve order capture and order fulfillment in the order-to-cash process when communicating sales orders between businesses. Business-to-business messaging uses Collaboration Messaging Framework to automate message flow so that Order Management can receive and process source orders from trading partners, and then reply with an advance shipment notice after shipping successfully finishes.

Web services use Open Applications Group Integration Specification (OAGIS) messages in the payload that it uses to handle the interactions that occur between Oracle Cloud Applications and trading partners. You can use your existing Electronic Data Interchange (EDI) infrastructure with OAGIS. This configuration provides you the flexibility to receive each source order and allow a trading partner and supplier to process it through order fulfillment.

Business-to-business order management allows you to achieve the following results:

- Process purchase order
- Change purchase order
- Cancel purchase order
- Acknowledge purchase order
- Acknowledge a change in the purchase order

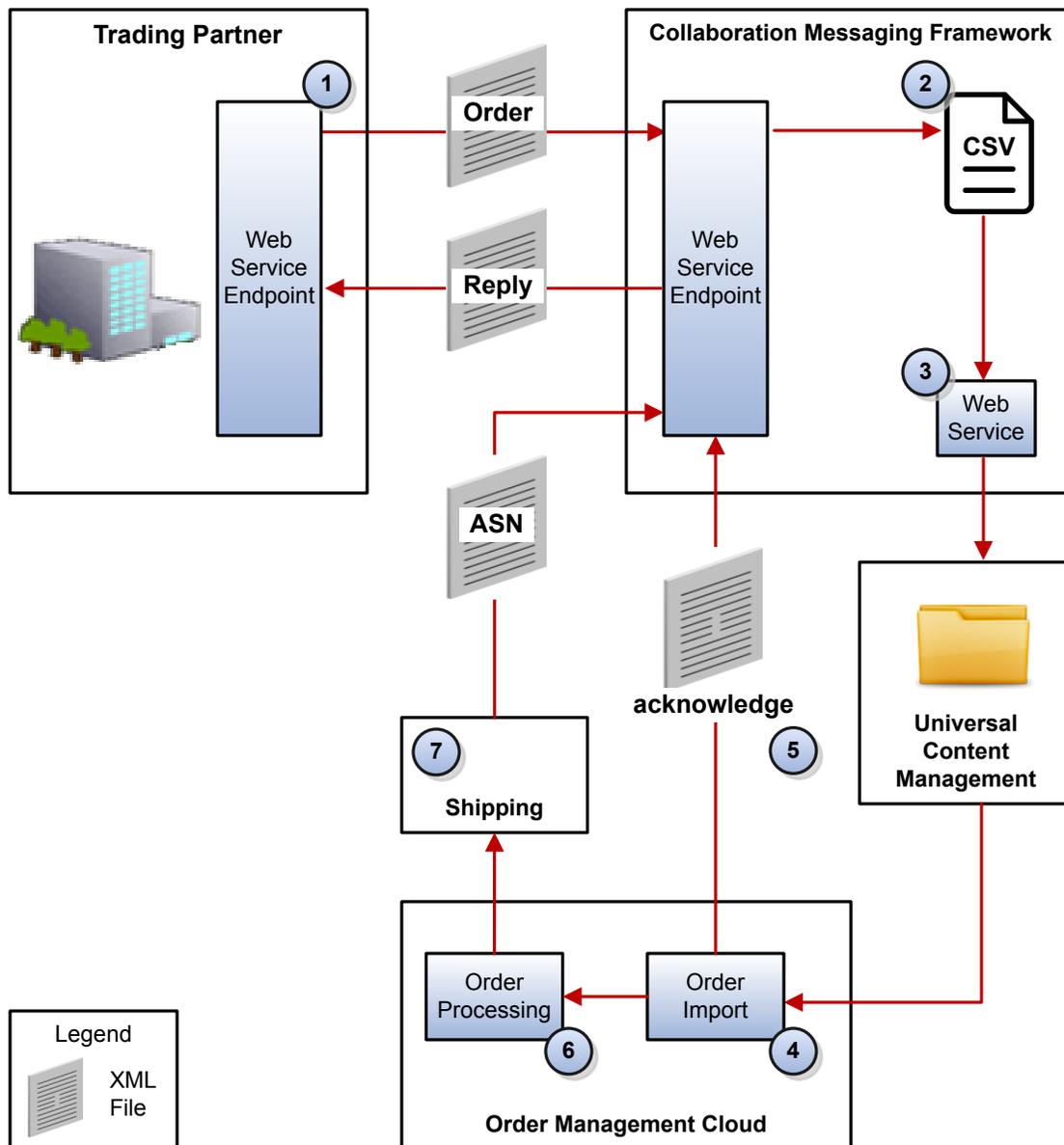
Business-to-business order management provides the following benefits:

- Reduce cost
- Increase processing speed and accuracy
- Improve relationships between business partners
- Simplify setup and management

## Business-to-Business Order Management: How it Works

The Collaboration Messaging Framework uses a web service to communicate sales order information between the trading partner and Order Management Cloud. It communicates this information in XML payloads.

The following diagram illustrates how Order Management uses Collaboration Messaging Framework to communicate sales orders between businesses.



### Explanation of Callouts

Order Management performs the following steps to communicate sales orders between businesses:

1. A trading partner uses a web service endpoint that they have set up and enabled on their server to send an XML document that contains source orders to the CollaborationMessage web service on the Collaboration Messaging Framework.
2. Collaboration Messaging Framework converts the source orders to a CSV format (comma separated value) that Order Management supports.

3. Collaboration Messaging Framework uses a web service to upload these CSV files to a Universal Content Management folder.
4. You use the order import template and a scheduled process to convert the CSV files into sales orders.
5. Collaboration Messaging Framework subscribes to a business event that the scheduled process raises when it finishes. Order Management recognizes this event, and then sends an order acknowledgment in an OAGIS message through Collaboration Messaging Framework to the trading partner.
6. Order Management processes the sales order, and then sends it to shipping for order fulfillment.
7. Oracle Fusion Shipping creates, and then processes the shipment. When shipping finishes delivery, it sends the ship confirm and advanced shipment notice in an OAGIS message, through Collaboration Messaging Framework, to the trading partner.

## Types of Messages That Collaboration Messaging Can Send

Collaboration Messaging can receive the following types of messages from a trading partner:

Collaboration Message	Description
OAGIS_10.1_CANCEL_PO_COLLAB_MSG_IN	Cancel purchase orders received from the trading partner.
OAGIS_10.1_CHANGE_PO_COLLAB_MSG_IN	Change purchase orders received from the trading partner.
OAGIS_10.1_PROCESS_PO_COLLAB_MSG_IN	Process purchase orders received from the trading partner.

Collaboration Messaging can send the following types of messages to a trading partner:

Collaboration Message	Description
OAGIS_10.1_ACK_PO_COLLAB_MSG_OUT	Send acknowledgment to the trading partner that Collaboration Messaging received the purchase order from.
OAGIS_10.1_ACK_CHANGE_PO_COLLAB_MSG_OUT	Send acknowledgment to the trading partner that Collaboration Messaging received the purchase order change from.
OAGIS_10.1_PROCESS_SHIPMENT_COLLAB_MSG_OUT	Send details to the trading partner about shipments.
OAGIS_10.1_PROCESS_RCV_DEL_COLLAB_MSG_OUT	Send details to the trading partner about purchase order deletions.

## Administering Business-to-Business Order Management: Procedure

You can use the Collaboration Messaging work area to set up business-to-business order management, and the Order Management and the Shipping work areas to test your set up.

In this example, assume you must create a relationship between a customer named Computer Service and Rentals who resides in Oracle Cloud Applications, and a trading partner named Computer Associates. You create an integration that achieves the following results:

- Allow Computer Service and Rentals to receive sales orders, updates, and cancellations from Computer Associates
- Allow Computer Service and Rentals to send acknowledgments and shipments to Computer Associates

## Summary of the Work

To administer business-to-business order management, do the following work:

1. Identify the trading partner and the messages to send and receive.
2. Create a relationship between the trading partner and the customer.
3. Simulate collaboration messaging.
4. Import source orders.
5. Process the sales order in shipping.
6. Examine the results in collaboration messaging.

This topic includes example values. You might use different values, depending on your business requirements.

## Identify the Trading Partner and the Messages to Send and Receive

Identify the trading partner and the messages to send and receive:

1. Log into Oracle Cloud Applications with a job role that includes the B2B Messaging Administration Duty role.

You can use one of the following job roles:

- Order Entry Specialist
- Order Manager
- Order Administrator
- Supply Chain Application Administrator
- Warehouse Manager

2. In the Navigator, click **Collaboration Messaging**.
3. Click **Tasks**, and then click **Manage Trading Partners**.
4. On the Manage Trading Partners page, click **Actions**, and then click **Create**.

You can use Manage Trading Partners to communicate with a trading partner directly or through a service provider. For example, you can communicate through a service provide to implement a solution where the trading partner must use EDI (Electronic Data Interchange), a custom XML format, a proprietary format, and so on. In this example, you set up the configuration to communicate directly with the trading partner.

5. In the Create Trading Partner dialog, set the following values, and then click **Save and Close**.

Attribute	Value
Service Provider	None
	Computer Service and Rentals will communicate directly with Computer Associates instead of going through a service provider, so you set this attribute to None.
Trading Partner ID	ComputerAssociates

Attribute	Value
Partner ID Type	Generic

6. In the Delivery Methods tab, click **Actions**, click **Add Row**, set the following values, and then click **Save and Close**. The Delivery Methods tab allows you to specify how to send messages to the trading partner.

Attribute	Value
Name	CMKDelivery00
Delivery Method	Web Service
Service Name	CollaborationMessage. Process
	This integration uses the Process method of the CollaborationMessage web service. You must make sure that the trading partner has deployed this web service on their server.
Security Policy	None
Endpoint	Enter the URL that locates the server and port that the trading partner uses at their location as the end point for their web services. For example:  http: // ComputerAssociatescom7012
User Name	Enter the user name that the trading partner server requires to access the endpoint.
Password	Enter the password that the trading partner server requires to access the endpoint.

7. Click **Outbound Collaboration Messages**, add the following rows, and then click **Save**.

Name	Collaboration Message
CMKORDERACK001	OAGIS_ 10.1_ ACK_ PO_ COLLAB_ MSG_OUT
CMKORDERACK002	OAGIS_ 10.1_ ACK_ CHANGE_ PO_ COLLAB_ MSG_OUT
CMKSHIPCONFIRM001	OAGIS_ 10.1_ PROCESS_ SHIPMENT_ COLLAB_ MSG_OUT

Note the following points:

- o The Outbound Collaboration Messages tab allows you to specify the messages that Collaboration Messaging sends to the trading partner.
  - o Set the Status to Active for each message.
  - o You can enter any text for the name. It is recommended that you enter text that describes the message contents.
8. Click **Inbound Collaboration Messages**, add the following messages, and then click **Save and Close**.

Name	Collaboration Message
CMKORDERIN001	OAGIS_10.1_CANCEL_PO_COLLAB_MSG_IN
CMKORDERIN002	OAGIS_10.1_CHANGE_PO_COLLAB_MSG_IN
CMKORDERIN003	OAGIS_10.1_PROCESS_PO_COLLAB_MSG_IN

Note the following points:

- o The Inbound Collaboration Messages tab allows you to specify the messages that Collaboration Messaging receives from the trading partner.
- o Set the Status to Active for each message.

## Create a Relationship Between the Trading Partner and the Customer

In this section, you create a relationship between Computer Service and Rentals, who is a customer that you have already defined in Oracle Cloud Applications, and the trading partner, Computer Associates. You also identify the documents that you must enable for this partner.

Create a relationship between the trading partner and the customer:

1. On the Overview page, click **Tasks**, and then click **Manage Customer Collaboration Configuration**.
2. On the Manage Customer Collaboration Configuration page, search for the customer who will receive communications from the trading partner.

For this example, enter **Computer Service and Rentals**. Assume that you already set up Computer Service and Rentals as a customer. If you have not set up the customer, then you must do so now. For details, see Customer Data Management Offering: Overview.

3. In the search results, if more than one row exists for this customer, then click the row that includes a check mark in the Ship to Party and in the Collaboration Configured options, and then click **Edit Collaboration Configuration**.
4. On the Edit Customer Collaboration Configuration page, in the Associated Service Providers area, click **Actions**, click **Add Row**, and then set the following values.

Attribute	Value
Service Provider	None
Trading Partner ID	ComputerAssociates Note that you specified this ID earlier in this topic when you identified the trading partner.
Order Processing Business Unit	Specify the business unit that processes each sales order that Order Management Cloud receives. For this example, Computer Service and Rentals resides in the Vision Operations business unit, so choose <b>Vision Operations</b> .
Application Partner Code	Accept the default value.

5. In the Collaboration Documents for Service Provider area, add the following documents, click **Save and Close**, and then click **Done**:
  - o PROCESS\_PO\_IN
  - o CHANGE\_PO\_IN
  - o CANCEL\_PO\_IN
  - o ACKNOWLEDGE\_PO\_OUT
  - o ACKNOWLEDGE\_CHANGE\_PO\_OUT
  - o PROCESS\_SHIPMENT\_OUT

Note the following points:

- o You use this area to specify the documents that this customer will communicate with the trading partner.
- o Set the Association Status to Enabled for each document.
- o The customer can communicate with the trading partner after you finish this step.

## Simulate Collaboration Messaging

Simulate collaboration messaging:

1. Click **Tasks**, and then click **Validate Inbound Collaboration Messaging Setup**.
2. On the Validate Inbound Collaboration Messaging Setup page, set the following values.

Attribute	Value
Service Provider	None
From Partner ID	ComputerAssociates
	You will be testing a flow that sends a new source order from the trading partner, Computer Associates, so you use the Partner ID that you specified earlier in this topic.
External Message ID	Enter the number that identifies the message you are testing, such as 08192016_001.
External Message Name	OAGIS_10.1_PROCESS_PO_COLLAB_MSG_IN
Processing Service	CollaborationMessage.Process

3. Click **Create Message Payload**.

Notice that the Message Payload area uses the settings that you specified in step 2 to create, and then display the XML that this test will use to communicate the message. You can modify this XML, if necessary.

4. Click **Process**.

A separate web service in Collaboration Messaging simulates the communication from the trading partner. It receives the XML document, validates it, and then displays a Processing Confirmation dialog that includes the XML result of the test. If the test is successful, then this XML will include the following code:

```
<ProcessingResultCode><Success>/<ProcessingResultCode>
```

5. Make a note of the line that includes the message ID. It will resemble the following:

```
<BODID>IN_8001</BODID>
```

In this example, **IN\_8001** is the message ID.

## Import Source Orders

The simulation that you performed earlier in this topic created a simulated source order. In this section, you import this source order into Order Management, and then view it. Note that you must also import source orders in a production environment.

Import source orders:

1. Use a scheduled process to import the source orders. For details, see *Using Files to Import Orders into Order Management: Procedure*.

Note the following points:

- o When you run the scheduled process named Load Interface File for Import, set the Data File to the file that the Messaging Framework created when you simulated collaboration messaging earlier in this topic. The data file will include a concatenation of the document name plus the message ID. For example, **PROCESS\_PO\_IN-1.0-IN\_8001**.
  - o When you run the Import Sales Order scheduled process, set the Source System to **ORA\_ELECTRONIC\_DOCUMENTS**. Order Management uses this predefined source system for each source order that it receives from the trading partner.
  - o As an option, you can specify the unique ID for the message that you received, such as **08192016\_001**, in the Batch Name parameter.
  - o When the Import Sales Order scheduled process finishes, Order Management can process the sales orders through order fulfillment.
2. In the Navigator, click **Order Management**.
  3. On the Overview page, search for the simulated order, such as **Demo\_Order**.
  4. In the Search Results, note that the value in the Source Order attribute includes the order number that you simulated, such as **Demo\_Order\_0819\_001**.
  5. In the Order column, click the **order number**.
  6. On the Order page, notice that the order status is Processing, and the order line status is Awaiting Shipping, which indicates that Order Management released the sales order to order fulfillment.

The messaging framework creates an acknowledgment message, and then publishes it to the trading partner.

## Process the Sales Order in Shipping

Process the sales order in shipping:

1. In the Navigator, click **Shipments**.
2. On the Overview page, search for the sales order that you noted in the Source Order attribute earlier in this topic, such as **Demo\_Order\_0819\_001**.
3. On the Edit Shipment Line page, click **Cancel**.
4. On the Manage Shipment Lines page, click **Actions**, click **Pick Release**, and then click **Save and Close**.
5. On the Overview page, search **Demo\_Order\_0819\_001**.
6. On the Edit Shipment Line page, notice that the Line Status is Staged, and then click the **link** next to Shipment.
7. On the Edit Shipment page, click **Ship Confirm**, and then click **Save and Close**.

Shipping creates an event. Collaboration Messaging subscribes to this event.

8. On the Manage Shipments page, in the row for your shipment, notice that the ASN Status is Sent.

This status indicates that Shipping sent the ASN status to Collaboration Messaging.

## Examine the Results in Collaboration Messaging

Examine the results in Collaboration Messaging:

1. In the Navigator, click **Collaboration Messaging**.
2. Click **Tasks**, and then click **Manage Collaboration Messaging History**.
3. In the Search area, set the following attribute, and then click **Search**.

Attribute	Value
Document	PROCESS_PO_IN

4. In the Messages area, in the External Message ID column, click the **row** that references the message that you simulated earlier in this topic, such as 08192016\_001.
5. Click **Actions**, and then click one of the following choices.

Menu Item	Description
View Source Document	View the input XML that Collaboration Messaging received from the trading partner.
View Output Document	View the output XML that Collaboration Messaging converted from the input XML, and then sent to Order Management.

6. Click **Done**.
7. Click **Tasks**, and then click **Manage Collaboration Messaging History**.
8. In the Search area, set the following attribute, and then click **Search**.

Attribute	Value
Document	ACKNOWLEDGE_PO_OUT

9. Examine the results of the acknowledgment.
10. In the Search area, set the following attribute, and then click **Search**.

Attribute	Value
Document	PROCESS_SHIPMENT_OUT

11. Examine the results of the shipment.

### Related Topics

- [Customer Data Management Offering: Overview](#)
- [Using Files to Import Orders into Order Management: Procedure](#)

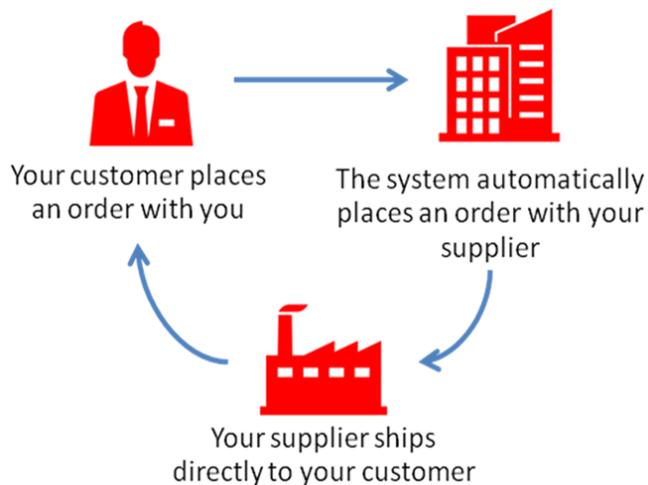
## Implementing Drop Ship

## Using Drop Ship with Order Management: Overview

Drop ship is a supply chain management technique where the seller relies on a supplier or contract manufacturer to build, store, and ship each sales order to the customer. The retailer does not maintain an inventory of the goods that each sales order references. You can use Order Management Cloud to automate this process.

Assume your customer places an order with you that requires a drop ship. Order Management sends a purchase request to Procurement, which places a purchase order with your supplier, and then your supplier ships directly to your customer. You provide a purchase order for the item and instructions that describe how to ship directly to the customer. The supplier or contract manufacturer ships the item, and your company earns a profit.

The following diagram illustrates the drop ship flow:



Drop ship allows you to do the following:

- Forecast and plan for future demand.
- Allow your customer to place an order with you, and for you to promise a ship date.
- Automatically place an order with your supplier.
- Allow your supplier to ship directly to your customer.
- Receive notification from your supplier when you when shipment has occurred.
- Manage order changes.

### **Sending Shipping Instructions and Packing Instructions in Drop Shipments**

If your fulfillment uses the predefined drop ship flow, then note the following points:

- Order Management sends shipping instructions and packing instructions in one or more attachments on the order line because Oracle Fusion Purchasing only accepts these instructions in an attachment.
- Oracle Fusion Purchasing only processes attachments that include an attachment category of MISC on the fulfillment line. It ignores an attachment that contains any other value.
- A user who uses the Create Order page in the Order Management work area can set the Category field in the Attachments dialog to any value that this field displays when adding an attachment to an order line. However, Order

Management sends a fixed value of **misc** (Miscellaneous) to Oracle Fusion Purchasing regardless of the value of the Category field.

- The user can enter text in the Shipping Instructions and Packing Instructions fields in the Shipping Details on the Create Order page. However, Order Management does not send this text to Oracle Fusion Purchasing because Oracle Fusion Purchasing only accepts them in an attachment.
- You must set the attachment category to **misc** on the order line of each source order that you import.

For details about attachment categories and how to set them, see the document that describes attachments in Article ID 1536633.1 on My Oracle Support at <https://support.oracle.com>.

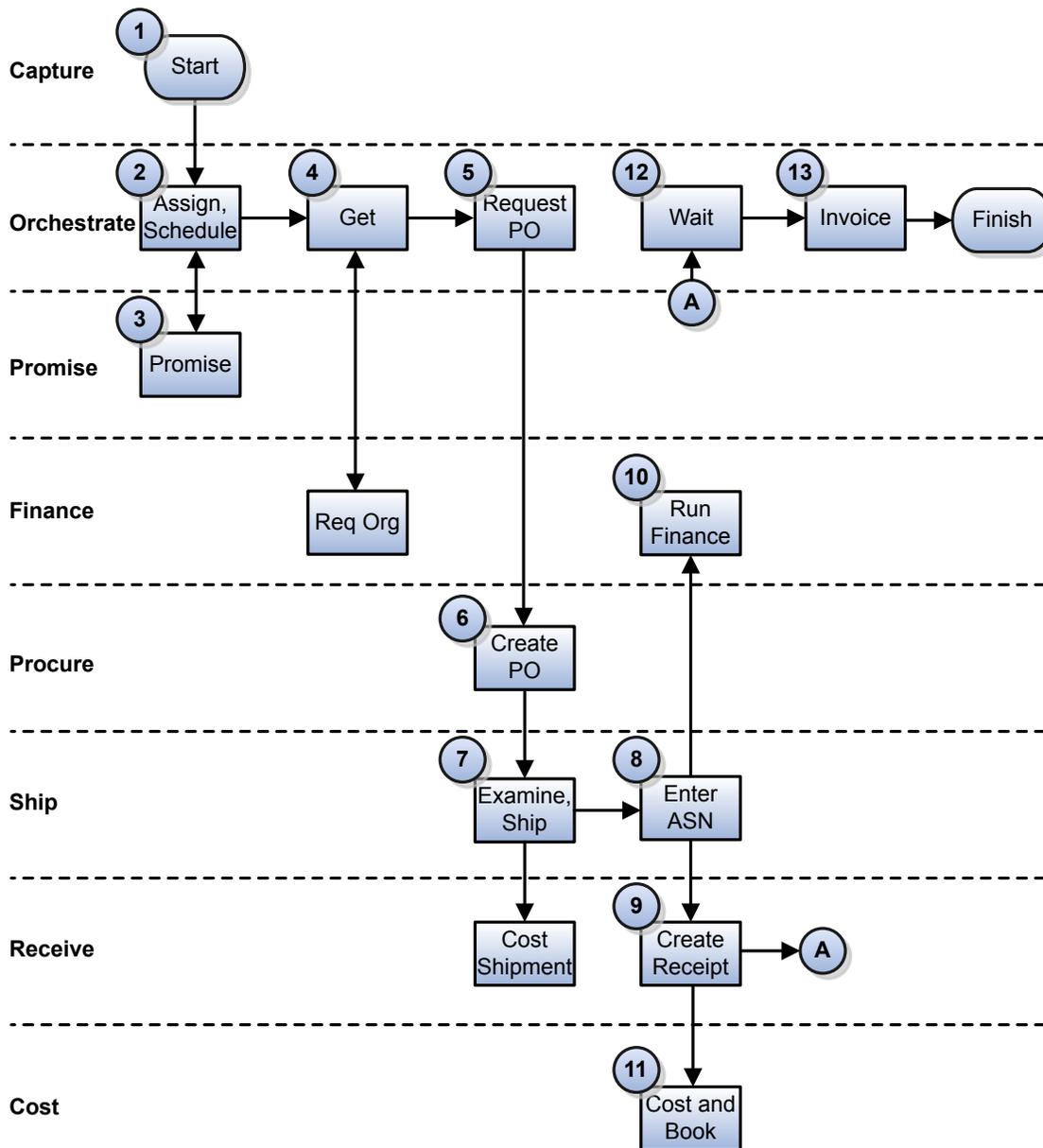
## Using Drop Ship with Order Management: How it Works

You can use Order Management Cloud to manage the flow of information through the systems and business functions that it uses to run the drop ship flow. This topic describes the systems and functions involved.

The following diagram illustrates how Order Management runs a drop ship flow. Each horizontal row in this diagram represents a system or business function that this flow uses. The drop ship flow is a variation of the order-to-cash flow. For details, see *Order-to-Cash: How It Works*.

You can modify how Order Management performs some of these steps. For example, you can use Manage Sourcing Rules to modify how Global Order Promising considers sourcing rules, you can use Manage Drop Ship Financial Flows to modify

how Supply Chain Financial Orchestration runs the financial flow, and so on. For details, see Implementing Drop Ship in Order Management: Procedure.



### Explanation of Callouts

Order Management does the following work when it runs a drop ship flow:

1. An order capture system captures a source order. Although Order Management comes predefined to capture sales orders directly in Order Management, you can also capture source orders in an external order capture system. For details, see Order-to-Cash with Order Capture Systems: How it Works.

2. The Order Orchestration component in Order Management assigns an orchestration process to the sales order, uses the Schedule Order task to start the scheduling work for this sales order, and then sends a scheduling request to Global Order Promising Cloud.
3. Global Order Promising considers sourcing rules, supplier calendar, capacity, and so on to identify the supplier and supplier site that can fulfill the order in the most efficient way. If the sales order specifies a supplier, then Global Order Promising uses this supplier.
4. Order Orchestration gets the requisition organization from Supply Chain Financial Orchestration so that it can create a purchase request. You can use the Manage Drop Ship Financial Flows task in Supply Chain Financial Orchestration to configure a relationship between the selling business unit and the requisition organization.
5. Order Management sends a purchase order request to Oracle Procurement Cloud.
6. Procurement creates a purchase requisition and a purchase order, and then sends the purchase order to the supplier. If a blanket purchase agreement exists, then Procurement might source the requisition from this agreement.
7. The supplier examines the purchase order and uses your shipping fulfillment system to ship the goods to the customer. The enterprise supplier also communicates with the Receiving component of Oracle Inventory Management Cloud to cost the shipment.
8. The supplier uses a supplier portal to enter an advance shipment notice (ASN). Order Management sends this notice to interested parties and Financial Orchestration. For details about the supplier portal, see [Supplier Portal Overview: Explained](#).
9. The Receiving component of Oracle Inventory Management Cloud creates a logical receipt in the receiving organization. It does not create a physical receipt because Order Management ships goods directly to the customer.
10. Financial Orchestration runs a financial flow that comes predefined to handle drop ships, and that specifies how to handle the flow that runs from the supplier to the customer.
11. Oracle Cost Management Cloud performs receipt accounting and cost accounting.
12. Order Management waits to receive the advance shipment notice from receiving that indicates that the goods have been shipped, and that the customer acknowledged receipt.
13. Order Management communicates with Oracle Receivables to create an invoice and process payments.

## Implementing Drop Ship in Order Management: Procedure

This topic describes how to set up Order Management Cloud so that it supports the drop ship flow.

### Summary of the Work

To implement drop ship in Order Management, do the following work:

1. Set up Oracle Applications.
2. Enable the features that drop ship requires.
3. Manage items.
4. Manage sourcing rules.
5. Manage financial flows for drop ship.
6. Manage order management parameters.
7. Manage suppliers and supplier sites.
8. Manage agreements.
9. Manage orchestration process definitions.
10. Test your work.

All of these steps are required except for Manage Agreements and Manage Orchestration Process Definitions, which are optional.

## Set Up Oracle Applications

You must set up various Oracle applications so that they support drop ship. For example, the Order Management offering references the Global Order Promising application to collect supply data from multiple sources. To start your implementation work for drop ship, it is recommended that you perform the following tasks:

1. Perform the Define Blanket Agreement and ASL task in Procurement. Requires a user login that can set up Procurement.
2. Perform the Create Drop-Ship Validation Org task in Inventory Management. Requires a user login that can set up Inventory Management.
3. Perform the following tasks in Global Order Promising. Requires a user login that can set up Global Order Promising:
  - o **Manage Data Collections.**
  - o **Define Item Processing Lead Times.**
  - o **Manage Sourcing Rules and Manage ATP Rules.** Set up sourcing rules and ATP rules so that they support the drop ship flow. For details, see ATP Rules, Allocation Rules, and Sourcing Rules: How They Work Together.
4. Perform the Manage Trade Operations task in the Manufacturing and Supply Chain Materials Management offering.
5. Perform the following tasks:
  - o Define Shipping Network
  - o Define Transit Lead Times
  - o Manage Assignment Sets
  - o Maintain Supply Network Model

For details about how to perform some of these tasks, see the guide titled Oracle SCM Cloud, Implementing Common Features for Oracle SCM Cloud..

## Enable the Features That Drop Ship Requires

Enable the features that drop ship requires:

1. Implement the Order Management offering.

For details, see the Implementing Order Management Cloud chapter in the guide titled Oracle SCM Cloud, Implementing Order Management.

2. Log into Order Management.
3. In the Navigator, click **Setup and Maintenance**.
4. On the Setup and Maintenance page, click **Order Management**.
5. Click **Actions**, and then click **Change Configuration**.
6. On the Configure Order Management page, in the Order Management row, click **Features**.
7. On the Features page, add a check mark to **Drop Ship**, and then click **Done**.

This feature allows you to access the setup tasks that you must perform in Oracle Supply Chain Financial Orchestration so that you can implement the drop ship flow.

8. Click **Actions**, and then click **Go to Offerings**.
9. On the Setup and Maintenance page, click **Procurement**.
10. Click **Actions**, and then click **Change Configuration**.
11. On the Configure Procurement page, in the Procurement row, click **Features**.
12. On the Features page, add a check mark to **Customer Sales Order Fulfillment**, and then click **Done**.

This feature allows Oracle Procurement to accept purchase requests for sales orders that a drop ship supplier fulfills, and to display some of the fields that you use to set up drop ship.

13. Click **Actions**, and then click **Go to Offerings**.

## Manage Items

Allow items to participate in drop shipments:

1. In the Navigator, click **Product Information Management**.
2. On the Overview page, click **Tasks**, and then click **Manage Items**.
3. On the Manage Items page, search for an item that your supplier must ship directly to your customer, such as Sentinel Deluxe Desktop.
4. In the Search Results list, click the **link** in the Item column for the item that you must manage.

For example, in the Sentinel Deluxe Desktop row, click **AS18947**.

5. On the Edit Item page, click **Specifications**, and then click **Purchasing**.
6. In the Item Organization: Purchasing area, set the following value.

Field	Description
Purchasable	Set to Yes. Allows this item to participate in a drop shipment.

7. If a blanket purchase agreement does not exist for this item, then, in the Pricing area, enter a number in the **List Price** field.
8. Click **Save**.
9. Repeat steps 3 through 8 for each item that you must make purchasable.

You must specify each item that your supplier ships directly to your customers. You specify this item as purchasable in the organization that is responsible for purchasing each item. The Product Information Management work area comes predefined to set the Purchasable field to Yes, so you must modify it only if you have set it No.

For details about how to make a large number of items purchasable, see the guide titled Oracle SCM Cloud, Implementing Product Management.

## Manage Sourcing Rules

You can specify a sourcing rule that already includes information for the supplier and the supplier site. Global Order Promising Cloud evaluates this sourcing rule and considers the supplier calendar, supplier capacity, and supplier lead times when it promises the order. For details, see ATP Rules, Allocation Rules, and Sourcing Rules: How They Work Together.

Manage the sourcing rule:

1. In the Navigator, click **Order Promising**.
2. On the Overview page, click **Tasks**, and then click **Manage Sourcing Rules**.
3. On the Manage Sourcing Rules page, search for the sourcing rule that you must modify, such as DOO-DS-Rule.
4. In the Search Results list, click the **sourcing rule** that you must edit, click **Actions**, and then click **Edit**.
5. On the Edit Sourcing Rule page, verify the following value.

Field	Description
Organization Assignment Type	Make sure the type is Global. If it is not Global, then you cannot use this sourcing rule.

6. In the Effective Start Date area, to add a sourcing rule for your drop ship flow, click **Actions**, click **Add**, and then set the following values.

Field	Description
Organization	Leave empty. The Organization is typically the warehouse that stores inventory for a flow that does not include a drop ship.
Type	Choose Buy From. Global Order Promising interprets Buy From to indicate that this value must come from a drop ship supplier.
Supplier	Choose the supplier who will drop ship the item.
Supplier Site	Choose the site that the supplier uses to store the item that this supplier will drop ship.

7. Click **Save and Close**.

8. Click **View Assignment Sets**.

9. In the Assignment Sets dialog, choose an **assignment set**, such as AYY-OP-ASET, and then click **Done**.

10. On the Edit Assignment Set page, in the Sourcing Assignments area, click **Actions**, click **Add Row**, and then assign an item to this sourcing rule. Use the following values.

Field	Description
Assignment Level	Set to Item. This setting assigns the sourcing rule to the item.
Item	Choose an item that the supplier that you specified in step 6 supplies. For example, if your supplier drop ships video cards, then choose CM22111.
Sourcing Type	Set to Sourcing Rule. This value associates the sourcing rule with the assignment set so that Order Management Cloud can use the rule to assign the item.
Sourcing Rule or Bill of Distribution	Set to Drop Ship.

11. Repeat step 10 for each item that the supplier you specified in step 6 supplies through a drop ship.

Note that you can also specify the following items for the supplier.

Field	Description
Supplier Calendar	Specify the working days for the supplier. Note the following: <ul style="list-style-type: none"> <li>You can modify the calendar for each supplier site.</li> <li>The calendar that you specify can be different from the calendar that the Supplier Site uses.</li> <li>Global Order Promising considers only working days when it calculates and incorporates lead times.</li> </ul>
Supplier Capacity	Specify the supplier capacity according to item, supplier, and supplier site. Order Promising measures the supplier capacity that exists on the arrival date.

Field	Description
Supplier Lead Times	You specify and collect the lead times for item processing on the item master in Oracle Fusion Product Model. You can specify a lead time for each supplier in Oracle Fusion Global Order Promising.

## Manage Financial Flows for Drop Ship

Specify the receiving trade organization for each selling business unit that your enterprise contains. For details about how the settings that you make in this topic affects the financial flow, see [Managing Drop Ship Financial Flows: Explained](#):

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, click **Orders**, search for, and then open Manage Drop Ship Financial Flows.
4. On the Manage Drop Ship Financial Flows page, in the Search Results area, click **Actions**, and then click **Create**.
5. Set the Name and Priority, set a value for the following field, and then click **Save**.

Field	Description
Supplier Ownership Change Event	<p>Choose one of the following values. Start the ownership change when the financial flow for this drop ship order receives:</p> <ul style="list-style-type: none"> <li>○ <b>ASN From Supplier</b>. Receives an advance shipment notice (ASN) from the supplier.</li> <li>○ <b>AP Invoice Match</b>. Receives an invoice from the supplier.</li> </ul> <p>For details about how to set this field, see <a href="#">Specifying Events That Indicate an Ownership Change: Explained</a>.</p>

6. In the Selling BU to Receiving BU Relationships area, click **Actions**, click **Add Row**, and then specify the relationship.  
For details about how to specify this relationship, see [Specifying Business Units for Financial Flows That Drop Ship Uses: Explained](#).
7. Repeat steps 3 through 4 for each selling business unit that your enterprise contains.
8. Click **Save and Close**.

## Manage Order Management Parameters

Specify an Order Management user who the buyer can contact to help resolve a problem that might occur with an order line during fulfillment that involves a drop ship supplier:

1. On the Setup page, search for, and then open Manage Order Management Parameters.
2. On the Manage Order Management Parameters page, click **Preparer for Procurement**.
3. In the Preparer for Procurement list, add one or more preparer, and then click **Save and Close**.

You must specify at least one value so that Order Management Cloud can successfully send a purchase request to Oracle Procurement Cloud. The default value applies across all business units. You can add a separate preparer for each business unit. For details, see [Managing Order Management Parameters: Procedure](#).

## Manage Suppliers and Supplier Sites

Specify the client business unit that Procurement will use to requisition and process invoices for the supplier site. For details, see [Supplier Sites and Supplier Site Assignments: Explained](#):

1. Log in with a user role that allows you to edit suppliers.

2. In the Navigator, click **Suppliers**.
3. On the Overview page, click **Tasks**, and then click **Manage Suppliers**.
4. On the Manage Suppliers page, search for the supplier that you must manage, such as GVR\_DS\_SUPPLIER1.
5. In the Search Results area, click the **supplier** you must manage, such as GVR\_DS\_SUPPLIER1.
6. On the Edit Supplier page, click **Sites**.
7. In the Site column, click a **site**, such as GVR\_SUP1\_SITE1.

The supplier site that you select must reference a procurement business unit. This business unit authorizes procurement services for the site.

8. On the Edit Site page, make sure the Purchasing option contains a check mark.  
This option specifies that this supplier site fulfills a purchasing role for the client business unit.
9. Click **Site Assignments**, and then set the following value. If necessary, click **Actions**, and then click **Create** to add a client business unit.

Field	Description
Client BU	Choose the client business unit that will requisition and process invoices for the supplier site.

10. Repeat step 9 for each requisition business unit that resides at the site that might request to purchase a drop ship item.
11. Click **Save and Close**.
12. Repeat steps 4 through 10 for each supplier that you must manage.

## Manage Agreements

Oracle Fusion Purchasing allows your buyer to create a blanket purchase agreement for the items that it will drop ship from a supplier. You must define an agreement for each supplier and supplier site, and associate one or more items with this agreement. You do this work when you set up purchasing. This topic describes how to modify this setup so that it supports drop ship. Purchasing uses the prices that the agreement specifies to set default values in the purchase documents. For details, see Blanket Purchase Agreement Lines: Points to Consider.

You manage agreements differently for a drop ship that includes a configured item. For details, see Setting Up Drop Ship for Configured Products: Procedure.

Manage agreements:

1. Log in with a user role that allows you to create an agreement.
2. In the Navigator, click **Purchasing**.
3. Click **Tasks**, and then click **Manage Agreements**.
4. On the Manage Agreements page, choose a value in the Procurement BU field, such as Vision Operations, and then click **Search**.
5. In the Search Results list, in a row that includes an agreement with a supplier who will drop ship your item, click the **link** in the Agreement column.
6. On the Agreement page, click **Actions**, and then click **Edit**.
7. On the Edit Document page, click **Controls**, and then set the following values.

Field	Description
Automatically Generate Orders	Set to one of the following values: <ul style="list-style-type: none"> <li>o <b>Contains a check mark.</b> Purchasing will automatically convert each requisition that it sources from this blanket purchase agreement. It will convert each requisition to a purchase order.</li> </ul>

Field	Description
	<ul style="list-style-type: none"> <li>○ <b>Does not contain a check mark.</b> Your buyer must do this conversion manually in the Purchasing work area.</li> </ul> <p>Purchasing examines eligibility according to the requisition to purchase order, even if Automatically Generate Orders does not contain a check mark. For example, it makes sure each of the following items is valid:</p> <ul style="list-style-type: none"> <li>○ Buyer</li> <li>○ Supplier</li> <li>○ Supplier site</li> <li>○ Source agreement</li> <li>○ Item is purchasable in the inventory organization of the procurement business unit</li> </ul>
Automatically Submit for Approval	It is recommended that you make sure this option contains a check mark. If it does not, then Order Management will create the order with an incomplete status, and the buyer must manually submit it for approval.
Use Customer Sales Order	Set to one of the following values: <ul style="list-style-type: none"> <li>○ <b>Contains a check mark.</b> Purchasing will group requisition lines that reference the same sales order number. It will group these lines on a single purchase order.</li> <li>○ <b>Does not contain a check mark.</b> Purchasing will not do this grouping.</li> </ul>

8. Click **Save**, and then click **Save and Close**.
9. Repeat steps 5 through 7 for each supplier who participates in your drop ship flow.

### Manage Orchestration Process Definitions

Order Management comes predefined with the following orchestration process:

- DOO\_OrderFulfillmentGenericProcess

It contains branches that run under the following conditions:

- If the shipment is a drop shipment, then run the Create Shipment Request branch.
- If the item in the inventory organization is enabled for back-to-back shipping, then run the back-to-back branch.
- If the first two conditions are false, then run the Create Reservation branch.

This branch gets the shipment from on-hand quantity in your warehouse. You can use this orchestration process as the default process assignment in your Assign and Launch rule. You can also define your own, custom orchestration process to meet your business requirements. For details, see *Defining Orchestration Processes: Explained*.

Examine the predefined orchestration process:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, click **Orders**, search for, and then open Manage Orchestration Process Definitions.
4. On the Manage Orchestration Process Definitions page, search for DOO\_OrderFulfillmentGenericProcess.
5. Examine the steps and branches.
6. If necessary, make a copy of this process, and then modify it so that it meets your business requirements, or create a new orchestration process and use DOO\_OrderFulfillmentGenericProcess as a starting point.

### Test Your Work

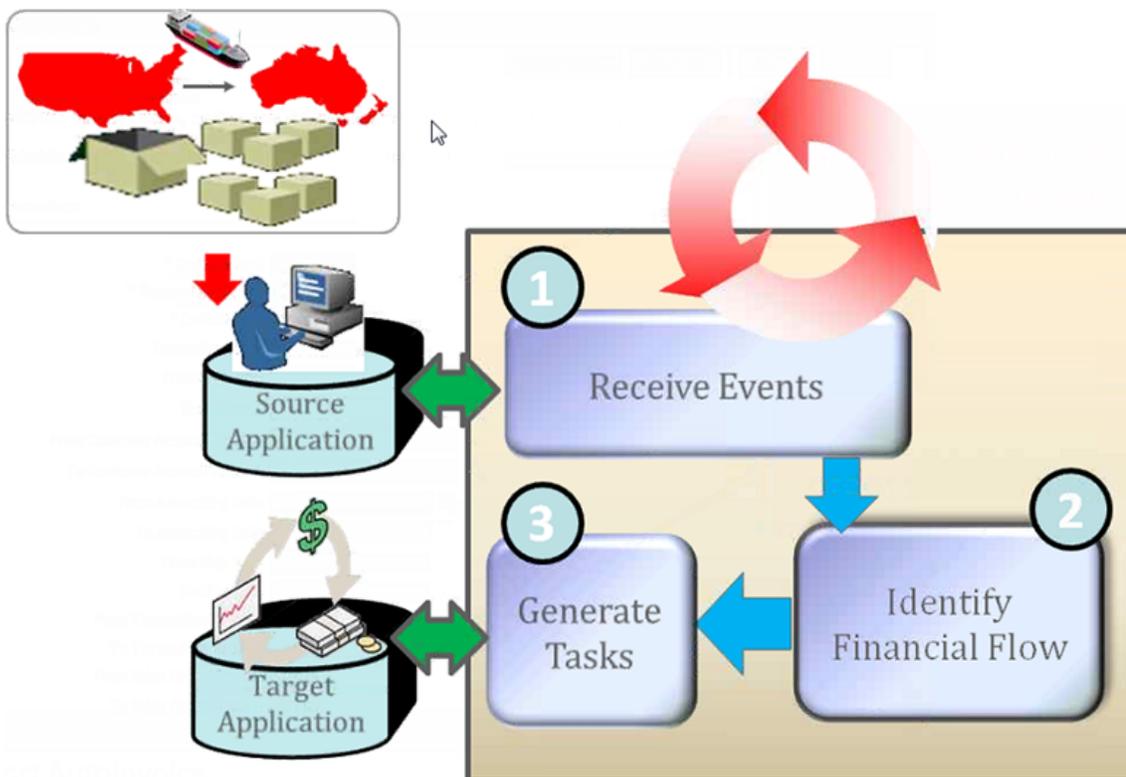
Create a sales order that uses drop ship. For details, see Using Drop Shipments: Procedure.

## Managing Drop Ship Financial Flows: Explained

Financial Orchestration Cloud controls the change in ownership of an item that it processes during a drop ship flow. For example, to transfer ownership from the selling business unit to the requisition business unit. The financial flow allows you to create cost accounting distributions that track the costs and ownership liability each time a transfer occurs between parties, including the supplier, one or more internal organizations, and the customer. It also creates an intercompany invoice for each internal ownership transfer, when necessary.

You can specify multiple requisition business units to manage and own multiple transactions that request the item. You can also specify the selling business unit that resides in the selling legal entity that sells the item.

The following diagram provides a high-level illustration of the process that Financial Orchestration performs during a drop ship flow.



### Explanation of Callouts

Financial Orchestration performs the following work during a drop ship flow:

1. Receives events. Financial Orchestration captures the physical supply chain event each time one occurs in the drop shipment flow. For example, when the supplier sends the Advance Shipment Notice to indicate that shipment of goods occurred.

2. Identifies the financial flow to run. Uses the following information to identify the financial flow to use:
  - o Purchase order for the drop ship
  - o Sales order information that it gets from source documents
  - o Internal selling and buying business units
  - o Financial orchestration qualifiers
  - o Priority of the financial orchestration flow
3. Creates tasks.

The flows that Financial Orchestration runs vary depending on the number of internal organizations that are involved:

- **One organization.** The selling business unit and the requisition business unit are the same unit, and Financial Orchestration uses only one financial flow for the drop ship.
- **Multiple organizations.** Financial Orchestration might run through multiple business units that involve procurement financial flows and shipment financial flows.

For details about the settings that you can make that affect this flow, see the Manage Drop Ship Financial Flows subtopic in *Implementing Drop Ship in Order Management: Procedure*. For details about financial orchestration, see *Financial Orchestration Flows: Explained*.

## Specifying Business Units for Financial Flows That Include Drop Shipments: Explained

The following diagram illustrates the relationship between the selling business unit, requisition business unit, suppliers, and customers in a financial flow that includes a drop shipment.



The Manage Drop Ship Financial Flows page sets the Receiving BU and the Receiving Legal Entity according to the value that you set in the Receiving Trade Organization field. For example, assume you set Receiving Trade Organization to Vision Operations, and that you set up the application to create a requisition in the Vision Operations inventory organization for any drop ship sales order that the Vision Operations business unit creates. In this situation, each of the following fields will contain a value of Vision Operations:

- Selling BU
- Selling Legal Entity
- Receiving Trade Organization

- Receiving BU
- Receiving Legal Entity
- Requisition BU

Order Management Cloud gets the value for the requisition organization field and the Requisition BU field from the purchase requisition that it creates for each drop ship order that involves an external supplier, and that requires a requisition organization.

## Specifying the Receiving Trade Organization

Choose the organization that does the following work for the drop ship flow when you specify the Receiving Trade Organization field on the Create Drop Ship Financial Flow page:

- Places the requisition for the goods
- Receives the drop ship purchase order
- Performs the receipt accounting and shipment accounting for the drop shipment

## Setting the Selling Business Unit and the Receiving Business Unit to Different Values

Financial Orchestration Cloud allows you to set up and run the financial flow according to ownership transfers that occur between the parties that are involved in the drop ship flow. You can configure the relationship that exists between the selling business unit and the receiving business unit. Financial Orchestration uses this relationship when it creates a purchase requisition for a drop shipment.

If you set the value for the Selling BU field to a value that is different from the value that the Receiving BU field contains, then Financial Orchestration determines whether or not it must perform more financial and accounting transactions according to the procurement and shipping flows that the application defines.

## Defining Different Requisition Trade Organizations

If you set the value for the Selling BU field to a value that is different from the value that the Receiving BU field contains, then a shipment financial flow must exist between the receiving business unit and the selling business unit. Financial Orchestration determines whether or not this flow exists when you create this relationship. If it does not, then Financial Orchestration displays an error message.

If a procurement financial flow exists in Financial Orchestration, and if Financial Orchestration can use it for the purchase order that the drop ship references, then Financial Orchestration uses this procurement financial flow when it does financial orchestration for the drop ship. For example, assume you must create a requisition that references China for each item that resides in the Metal category, and create a requisition that references the United States for each item that resides in the Plastic category. In this situation, you can create drop ship financial flows that use the following configuration.

Drop Ship Financial Flow	Inventory Organization	Financial Orchestration Qualifier
Flow 1	Resides in China as the receiving trade organization.	Uses Metal as the item category.
Flow 2	Resides in the United States as the receiving trade organization.	Uses Plastic as the item category.

For details about how to prioritize a drop ship financial flow, see Oracle Supply Chain Management Cloud, Implementing Order Management, on My Oracle Support at [https:// support.oracle.com](https://support.oracle.com).

## Specifying Events That Indicate an Ownership Change: Explained

To specify the event that indicates the ownership change, you can specify to use the receipt of the advanced shipment notice (ASN) from the supplier, or to use the arrival and entry of the accounts payable invoice. You specify these values in the Supplier Ownership Change Event field on the Manage Drop Ship Financial Flows page.

### Using ASN From Supplier

If the supplier site sends an advanced shipment notice (ASN), then it is recommended that you set the Supplier Ownership Change Event field to ASN From Supplier. The arrival of the advanced shipment notice starts the downstream processes that cost and bill the shipment, including creating the drop ship receipts, starting the billing processes, and starting the financial orchestration process for the drop ship order. Order Management Cloud loads the advanced shipment notice into the Receiving component, which creates a type of drop ship flow known as **receipt and delivery** that records an event. Receiving sends this event to Financial Orchestration and Order Management. Financial Orchestration uses the status information and details that this event contains to start financial orchestration processes. These processes perform trade transactions in costing, process intercompany receivable invoices, process intercompany payable invoices, and so on. Order Management uses this information to start billing processes and order orchestration processes.

The supplier can use one of the following features on the supplier portal to create the advanced shipment notice:

- Create ASN
- Create ASBN
- Upload ASN
- Upload ASBN

If the supplier uses informal communication, such as e-mail, to report the shipment of a purchase order that references a drop shipment, then the warehouse manager can also use these features on the Receipts page of the Warehouse Operations work area to manually enter the advanced shipment notice. You can also upload advanced shipment notices electronically through XML or EDI. For details, see [Supplier Portal Overview: Explained](#), and [Creating ASNs and ASBNs: Overview](#).

### Using AP Invoice Match

If the supplier site does not send an advanced shipment notice, then it is recommended that you set the Supplier Ownership Change Event field to AP Invoice Match. Financial Orchestration receives the events that are related to the accounts payable invoice from Accounts Payable, and then uses one of these events to start the financial orchestration processes and to inform the receiving process. The arrival and entry of the accounts payable invoice starts the downstream processes that cost and bill the shipment. The receiving process works the same way that it does for ASN From Supplier, except the process never creates an advanced shipment notice. Order Management creates the receipt and delivery drop ship when it creates the accounts payable invoice for the purchase order that the drop ship references.

The following scheduled processes handle processing:

- **Transfer Invoice Details to Supply Chain Financial Flow Orchestration.** Sends details about validated invoices, canceled invoices, and corrected invoices to Financial Orchestration.
- **Transfer Ownership Chain Events to Receiving.** Sends details about the AP Invoice Match from Financial Orchestration to the receiving process.

## Setting Up Drop Ship for Configured Items: Procedure

To allow your users to drop ship a configured item, you must define an agreement for the configuration model that specifies the configured item and for the optional components that might be part of the configure-to-order shipment.

A hybrid configuration model is a type of configuration model that includes an assemble-to-order configuration model as the child and a purchase-to-order configuration model as the parent. Order Management Cloud uses one blanket purchase agreement to source the assemble-to-order components, and a different blanket purchase agreement to source the purchase-to-order components. So, Order Management issues separate purchase orders to the supplier.

Set up drop ship for configured items:

1. Log in with a user role that allows you to edit values in the Procurement work area.
2. In the Navigator, click **Purchasing**.
3. Click **Tasks**, and then click **Manage Agreements**.
4. Click **Actions**, and then click **Create**.
5. In the Create Agreement dialog, set the following values, and then click **Create**.

Field	Description
Style	<p>Select the following value:</p> <ul style="list-style-type: none"> <li><input type="radio"/> Configure to Order Blanket Purchase Agreement</li> </ul> <p>You must use this value for an assemble-to-order configuration model. For details, see Supplier Agreement Creation: Points to Consider.</p>
Supplier	Select the supplier who will drop ship the configured item.

6. On the Edit Document page, in the Lines area, add at least one line, and then click **Submit**.

For details about how to import multiple agreements, see Understanding File-Based Data Import and Export on Oracle Help Center at [http://docs.oracle.com/cloud/latest/salescs\\_gs/docs.htm](http://docs.oracle.com/cloud/latest/salescs_gs/docs.htm).

# 3 Integrating Order Management Cloud

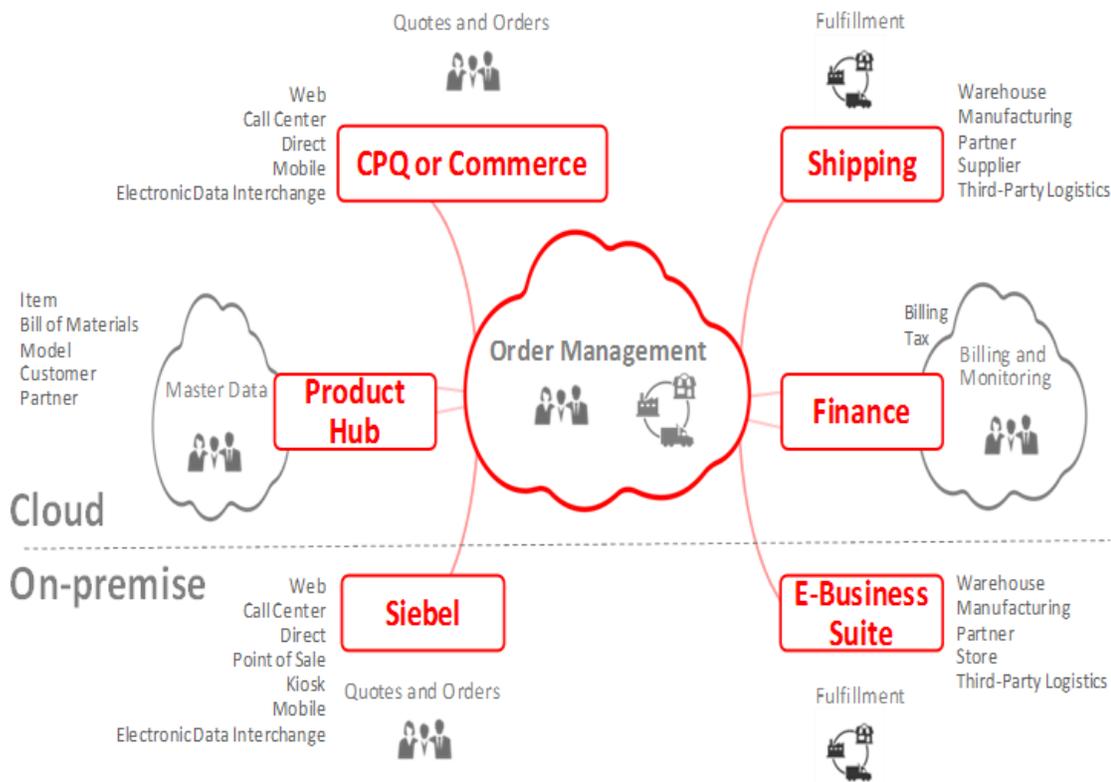
## Integrating Order Management with External Systems: Overview

Order Management Cloud comes predefined with integrations to several other Oracle cloud services so that you can use a set of advanced fulfillment processes that require minimal configuration work to get them up and running in your environment. For example, Order Management and Oracle Fusion Inventory Management work together to fulfill orders that your users can enter directly in Order Management or that you import from some other source.

Order Management also provides a set of web services that you can use to integrate with other Oracle and third-party cloud and on-premise applications that your supply chain requires to complete the order-to-cash process:

- **Integrate cloud processes.** Integrate cloud processes, including order-to-cash, drop-ship, back-to-back, configure-to-order, and internal orders. The integration includes predefined processes and simplified set up.
- **Integrate applications.** Order Management comes integrated with the following Oracle cloud services:
  - Oracle Inventory Cloud
  - Oracle Cost Management Cloud
  - Oracle Manufacturing Cloud
  - Oracle Procurement Cloud
  - Oracle Product Hub Cloud
  - Oracle Financials Cloud
  - Oracle Configure, Price, and Quote Cloud.
- **Integrate web services.** Use web services to do batch and real-time order import, send a request to a fulfillment system, receive a status from a fulfillment system, and integrate to other cloud and on-premise systems. Inventory, shipping, receiving, finance, and order management are each an example of a fulfillment system.

The following diagram illustrates some of the integrations that you can set up with Order Management.



For details, see the following topics:

- Creating User Credential Keys: Procedure
- Connecting Order Management to Source Systems: Procedure
- Connecting Order Management to Fulfillment Systems: Procedure
- Troubleshooting Connection Problems with Source Systems: Procedure
- Routing Requests to Fulfillment Systems: Explained
- Routing Requests to Fulfillment Systems: Procedure
- Routing Requests to Fulfillment Systems When Cross-Reference Data Does Not Exist: Procedure
- Integrating Order Management to Communicate Attachments: Explained
- Collecting Orchestration Data: Explained
- Mapping Names and Codes from Order Capture Systems: Explained
- Order Import Template: Explained

For details, see the following documents on My Oracle Support at <https://support.oracle.com>:

- Integrating Order Management Cloud with Oracle eBusiness Suite (Article ID 2132744.1)
- Non-Fusion Fulfillment System Connectors in DOO: Oracle Fusion Distributed Order Orchestration White Papers (Article ID 1536633.1).
- Using Web Services with Oracle Fusion Order Management Cloud (Article ID 2051640.1)

# Integrating Order Management with Upstream Source Systems

## Creating User Credential Keys: Procedure

You must create a user credential key to integrate Order Management Cloud with an external service.

The external interface layer is an intermediary layer that uses open access protocols, such as HTTP, so extra security setup is required. You must make sure that the user credential is valid in the external system and in the security certificate so that the integration can encrypt and decrypt messages.

Create a user credential key:

1. Use Oracle Wallet Manager to add a user credential key to a credential map. For more information, see Oracle Fusion Middleware Security Guide 11g Release 1 (11.1.1). You must use the administration privilege and administrator role.
2. In Oracle Wallet Manager, in the Select Map list, select `oracle.wsm.security`.
3. Enter the user credential key, user name, and password from the service that you are integrating with Order Management.
4. Log into Order Management Cloud.
5. In the Navigator, click **Setup and Maintenance**.
6. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
7. On the Setup page, click **Orders**, search for, and then open Manage External Interface Web Service Details.
8. On the Manage Connector Details page, click **Actions**, click **Add Row**, and then set the values that you set in steps 1 through 3.

## Connecting Order Management to Source Systems: Procedure

If your deployment must integrate with an external system, then you can register a connector that allows Order Management Cloud to communicate with it. You must create, deploy, and register this connector. This topic describes how to register the connector and connect Order Management to an external source system, such as Oracle Configure, Price, and Quote Cloud. To do this, you add a connector that uses a web service that communicates order details with the source system.

The external interface layer is an intermediary layer that uses open access protocols, mainly HTTP, so extra security setup is required. You must make sure that the user credential is valid in the source system and in the security certificate so that the integration can encrypt and decrypt messages. For details about how to call a web service, see the topic that describes DOO Connector: Invoking DOO Web Services from Non-Fusion Order Capture System in Article ID 1536633.1 on My Oracle Support at <https://support.oracle.com>. For details about Oracle Configure, Price and Quote Cloud, see <https://cloud.oracle.com/cpq>.

### Summary of the Work

To connect Order Management to a source system, do the following work:

1. Create a user credential key. For details, see Creating User Credential Keys: Procedure.
2. Define the source system.
3. Add the connector.

This topic includes example values. You might use different values, depending on your business requirements.

## Define the Source System

Define the source system:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, click **Customers**, search for, and then open Manage Trading Community Source Systems.
4. On the Manage Trading Community Source Systems page, click **Actions**, and then click **Create**.
5. On the Create Source System page, set the following values.

Field	Description
Code	Enter any text that Order Management can use as an abbreviation for this external system. Order Management uses this code to identify this external system throughout the user interface, such as in lists and logs. For example, assume you work for a company named Vision Corporation, and that your deployment must integrate with a legacy order capture system named Vision Capture. You can enter VCAP.
Name	Enter text that describes the source system, such as Vision Capture.
Type	Choose Spoke.
Enable for Order Orchestration and Planning	Contains a check mark.

6. Click **Save and Close**, and then click **Done**.
7. On the Search page, search for Manage Upstream and Fulfillment Source Systems, and then open it.
8. On the Manage Orchestration Source Systems page, click **Actions**, and then click **Create**.
9. In the Create Source System dialog, set the following values.

Field	Description
Code	Choose the code that you created in step 4, such as <b>VCAP</b> .
Time Zone	Choose the time zone where the server is located.
Version	Choose <b>Other</b> .
Order Orchestration Type	Specify how your deployment will use this external system. For example, choose <b>Order capture</b> .
Collections Allowed	Contains a check mark.
Enable Data Cross-Reference	Contains a check mark.

10. Click **Save and Close**, and then click **Done**.

## Add the Connector

Add the connector:

1. On the Search page, search for Manage External Interface Web Service Details, and then open it.
2. On the Manage Connector Details page, click **Actions**, and then click **Add Row**.
3. In the new row, set the following values, and then click **Save and Close**.

Field	Description
Target System	Choose the code that you created in step 4 on the Create Source System page, such as <b>VCAP</b> .
Connector Name	Enter text that describes the connector. For example, enter Connector_to_VCAP
Connector URL	Enter the URL that locates the connector service that resides on the source system. In this example, enter the URL that locates the VCAP system.
User Name and Password	Enter the values that the Status Update service requires. For example, the user that you specify must be a valid user, and this user must use the privileges that allow this user to run the Status Update service. Order Management uses the credentials that you provide so that it can communicate with the order capture system when it creates an order, and when it provides a status update.

4. **Optional.** Allow more than one source system instance to communicate with Order Management:
  - o Use Trading Community Architecture to add a value to the Target System drop-down list.
  - o Repeat step 3, except set the Target System to the value that you added in Trading Community Architecture.

For example, assume you work for a telecommunications company. You add one connection to a system named PER\_ORA\_BM\_CPQ for personal lines, and then add another connection to a system named BUS\_ORA\_BM\_CPQ for business lines. Note that CPQ is an acronym for Configure, Price and Quote. Order Management will deliver status notifications and billing notifications to any system that contains the string ORA\_BM\_CPQ. You can add a prefix, a suffix, a prefix and a suffix, or no prefix or suffix to this string. For example, you can use ABC\_ORA\_BM\_CPQ\_XYZ.

5. Verify that Order Management is connected to the source system, and that it is communicating sales order data:
  - o Use a page in the Order Management work area to verify that it updated an order status. For example, verify that it modified the status from Scheduled to Shipped.
  - o Log into your source order system, and then verify that it displays the updated status value of the order that you examined in Order Management. For example, if Order Management modified the status from Scheduled to Shipped on the fulfillment line, then verify that your source system also displays Shipped for this value.

If Order Management cannot connect to your source system, then it might display an error message that indicates that it cannot connect. If this situation occurs, then see [Troubleshooting Connection Problems With Source Systems: Procedure](#).

## Troubleshooting Connection Problems with Source Systems: Procedure

You can troubleshoot an error message that indicates that Order Management Cloud cannot connect to a source system, such as Oracle Configure, Price, and Quote Cloud.

Troubleshoot a connection problem with a source system:

1. In the Navigator, click **Order Management**.
2. On the Overview page, click **Tasks**, and then click **Manage Order Orchestration Messages**.
3. On the Manage Order Orchestration Messages page, in the **Order Orchestration Function** field, select **Send Event Notification**, and then click **Search**.
4. Examine the search results:
  - o Make sure the Search Results includes an entry that includes the same URL that you specified in the Connector URL field of the Manage Connector Details page. For details about this field, see Connecting Order Management to Source Systems: Procedure.
  - o Make sure this URL correctly identifies the connector service that resides on the source system. To do this, log into your source system, and then examine the connector services that are running.
  - o Wait a minute for the log to refresh, requery the Order Orchestration Messages page, and then examine the search results again to determine whether or not the connection successfully restarted. A network error or some other problem might cause the connection to momentarily fail. If the connection restarted successfully, then the list will include information about the events that are associated with the connector URL.

## Integrating Order Management with Downstream Fulfillment Systems

### Connecting Order Management to Fulfillment Systems: Procedure

You can use a web service to allow Order Management Cloud to communicate with a fulfillment system through the external interface layer. You can use a predefined web service or create a new one.

You must create and register a connector for each fulfillment system. For details about how Order Management routes requests to a fulfillment system, see Architectural Layers in Order Management: How They Work, and the document that describes Non-Fusion Fulfillment System Connectors in Article ID 1536633.1 on My Oracle Support at <https://support.oracle.com>.

Connect Order Management to fulfillment systems:

1. Use an integrated development environment of your choice to create a transformation style sheet that can handle the web service payload.
2. Create and deploy the connector:
  - o Create an XSLT transformation file that transforms the Common Order Service Data Object (SDO) to the payload of the web service where Order Management must connect. You can use an integrated development environment of your choice or Integration Cloud Services from Oracle to create this file.

- o Create an XSLT transformation file that transforms the reply from the web service to an Order Management message.
- o Reference the transformation style sheet from the connector.

Deploy the connector. Make a copy of the connector template, and replace the XSLT transformation files with the files that you created for the connector.

3. Create a user credential key, and add it to the identity store on the server that will call the web service. For details, see *Creating User Credential Keys: Procedure*.
4. Get the user credentials that the service provider requires when calling their web service. You can typically get this information from the service provider.
5. Send a request to the IT administrator to add the user credentials that you identified in step 4. Request to add them to the server that will call the web service. Use the CSF-KEY (Credential Store Framework) reference.
6. In the Navigator, click **Setup and Maintenance**.
7. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
8. On the Setup page, click **Orders**, search for, and then open *Manage External Interface Web Service Details*.
9. On the *Manage Connector Details* page, click **Actions**, click **Add Row**, and then register the service that Order Management must interact with. The external system hosts this service. Set the following values:

Field	Description
Target System	Choose a system.
Connector Name	Enter the name of the web service.
Connector URL	Enter the URL that locates the web service that resides on the fulfillment system.
User Name and Password	<p>Enter the user name and password that the service requires.</p> <p>As an option, you can use the CSF-KEY that references the user credential that Order Management must use to interact with the external web service. There is no requirement to match the actual name of the connector, so you can provide a short name. You can then use this short name in a routing rule. The CSF-KEY applies to all services that the external system provides.</p>
Keystore Recipient Alias	<p>For each server that runs an external web service that Order Management calls, it is recommended that you configure this server to advertise the security certificate in the WSDL. The Oracle WebLogic Server advertises the security certificates, by default. If your servers support this advertisement; then enable it.</p> <p>If you configure the server to advertise the security certificate, then it is recommended that you use the keystore recipient alias. You must do the following work:</p> <ul style="list-style-type: none"> <li>o Ask the service provider for the security certificate.</li> <li>o Make sure an IT administrator imports the target server security certificate into the calling server and provides the keystore recipient alias.</li> <li>o Add this alias to the external service entry that you created when you specified the user credential.</li> <li>o Add this alias to the Keystore Recipient Alias field on the <i>Manage Connector Details</i> page. This key applies to all services that the target system provides.</li> </ul> <p>If these options do not work, then configure the servers to use the Oracle security certificate, and then import this certificate into your servers. No setup is required on the calling server for the security certificate.</p>

10. Repeat step 8 until you register all services that Order Management must interact with.
11. Create a routing rule that selects the web services. For details see Routing Requests to Fulfillment Systems: Explained.

*Related Topics*

- [Order-to-Cash with Order Capture Systems: How it Works](#)
- [Architectural Layers in Order Management: How They Work](#)

## Routing Requests to Fulfillment Systems: Explained

An external interface routing rule specifies how to route a fulfillment request to a fulfillment system. You can specify each rule so that it selects the fulfillment system connector according to order, fulfillment line, or orchestration process attribute. At runtime, the rule calls the connector service that translates the payload into the structure that the fulfillment system uses.

The following examples describe ways that you might use an external interface routing rule.

Example	Description
Route orders according to the value of an orchestration process attribute.	Route orders that are ready to be shipped to a specific shipping fulfillment system according to the task type. To meet this requirement, you can write the following external interface routing rule: <ul style="list-style-type: none"> <li>• If the type code is Shipment, then route the order to the ABCShippingSystem connector.</li> </ul>
Route orders according to the value of a customer attribute.	Assume your company uses two invoicing systems, and that system ABC sends invoices to a customer named Widget Company. To meet this requirement, you can write the following external interface routing rule: <ul style="list-style-type: none"> <li>• If the product type is Goods, and if the task type is Invoice, then route the request to the ABCInvoicingSystem connector.</li> </ul>

*Related Topics*

- [Architectural Layers in Order Management: How They Work](#)

## Routing Requests to Fulfillment Systems: Procedure

This topic describes how to create an external interface routing rule that routes a request to a fulfillment system. For details about how to use a simplified rule builder, see Using Visual Information Builder to Create Rules: Explained.

Assume you must create a routing rule that implements the following condition:

- If the request contains the 2TX server model, and if the task type code of this request is Invoice, then route the service request to the Big Server fulfillment system.

### Summary of the Work

Create an external interface routing rule:

1. Create the If statement.
2. Create the Then statement.

This topic includes example values. You might use different values, depending on your business requirements.

## Create the If Statement

In this section, you create the following If statement:

- If the request contains the 2TX server model, and if the task type code of this request is Invoice

For a demonstration, see *Creating Business Rules: Demo*.

Create the If statement:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, click **Orders**, search for, and then open *Manage External Interface Routing Rules*.
4. On the *Manage External Interface Routing Rules* page, click **Add Rule**, click **Next**, and then replace Rule 1 with **Route Invoice Request Rule 1**.
5. Next to *Route Invoice Request Rule 1*, click **Expand**, and then click **Show Advanced Settings**.
6. Enter a description. For this example, use the following value.

Field	Value
Description	Rule that routes invoice requests for item 2TX to the Big Server fulfillment system.

7. In the If area, click **Left Value**.
8. In the Condition Browser, expand **DOOExternalInterfaceLayer**, expand **FulfillLineTLVO**, click **InventoryItemId**, and then click **OK**.
9. Make sure the operator equals **is**, and then, in the Right Value field, enter **300000003357446L**. L indicates a long value. Inventory Item ID 300000003357446 identifies the 2TX server model.
10. Click the **down arrow** to the right of the Right Value field, and then click **Simple Test**.
11. In the new row, click **Left Value**.  
In the Condition Browser, expand **DOOExternalInterfaceLayer**, expand **HeaderTLVO**, click **TaskName**, and then click **OK**.
12. Make sure the operator equals **is**, and then, in the Right Value field, enter **"Invoice"**. You must include the double quotation marks.

## Create the Then Statement

In this section, you create the following Then statement:

- Route the service request to the Big Server fulfillment system

Create the Then statement:

1. In the Then area, click **Add Action**, and then click **Call**.
2. In the Select a Target field, enter **print**.
3. Click **Edit Properties**.
4. In the Arguments dialog, enter the following value, and then click **OK**. You must include the double quotation marks.

Name	Value
Message	"DOO_ TransportationPlanning: " +headerSourceOrderNumber

5. Specify the connector:

- o In the Then area, click **Add Action**, and then click **Assert New**.
- o Click **Select a Target**, and then click **DOOExternalInterfaceLayer.Result**.
- o Click **Edit Properties**.
- o In the Edit Properties dialog, enter the following values, and then click **OK**. You must include the double quotation marks.

Name	Value
ResultObj	"Connector01"
In this example, assume you already used the Manage Connector Details page to add a custom connector named Connector01, and referenced it with the Big Server fulfillment system.	
ResultObjKey	"SERVICE_NAME"

6. Specify the interface:

- o In the Then area, click **Add Action**, and then click **Assert New**.
- o Click **Select a Target**, and then click **DOOExternalInterfaceLayer.Result**.
- o Click **Edit Properties**.
- o In the Edit Properties dialog, enter the following values, and then click **OK**. You must include the double quotation marks.

Name	Value
ResultObj	"SDO"
ResultObjKey	"InterfaceType"

7. Specify the service:

- o In the Then area, click **Add Action**, and then click **Assert New**.
- o Click **Select a Target**, and then click **DOOExternalInterfaceLayer.Result**.
- o Click **Edit Properties**.
- o In the Edit Properties dialog, enter the following values, and then click **OK**. You must include the double quotation marks.

Name	Value
ResultObj	"DOO_ TransportationPlanning"
ResultObjKey	"SERVICE_NAME"

8. Click **Save and Close**.

### Related Topics

- Using Visual Information Builder to Create Rules: Explained
- Using Visual Information Builder to Manage External Interface Routing Rules: Procedure
- Creating Business Rules: Demo

## Routing Requests to Fulfillment Systems When Cross-Reference Data Does Not Exist: Procedure

This topic describes how to create an external interface routing rule that routes a request to a fulfillment system when a cross-reference for this system does not exist or when Order Management Cloud cannot find this data.

Assume you must create a routing rule that implements the following condition:

- If the task type code on the fulfillment line is Shipment, then route the service request to Shipping Fulfillment System A.

If Order Management cannot find customer cross-references for Shipping Fulfillment System A in an Oracle Application, then you must set the value of parameter RESOLVE\_XREF\_FOR\_CUSTOMERINFO to Yes.

### Summary of the Work

To create an external interface routing rule when cross-reference data does not exist, do the following work:

1. Create the If statement.
2. Create the Then statement.

This topic includes example values. You might use different values, depending on your business requirements.

### Create the If Statement

In this section, you create the following If statement:

- If the task type code on the fulfillment line is Shipment

For a demonstration, see Creating Business Rules: Demo.

Create the If statement:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, click **Orders**, search for, and then open Manage External Interface Routing Rules.
4. On the Manage External Interface Routing Rules page, click **Add Rule**, click **Next**, and then replace Rule 1 with **Shipping Fulfillment System A**. You can enter any value.

To simplify managing your configuration, it is recommended that you enter the name that you used when you added the connector to this fulfillment system on the Manage Connector Details page.

5. Next to Shipping Fulfillment System A, click **Expand**, and then click **Show Advanced Settings**.
6. Add a check mark to Advanced Mode and to Tree Mode.
7. In the Root list, select **DOOExternalInterfaceLayer.HeaderTLVO**.
8. In the If area, in the field to the left of Is A, enter **header**.
9. In the If area, in the field to the right of Is A, select **DOOExternalInterfaceLayer.HeaderTLVO**.

10. In the If area, in the Left Value field, enter **"Shipment"**. You must include the double quotation marks.
11. Click **Is**, and then click **Equals Ignore Case**.
12. Click **Right Value**.
13. In the Condition Browser, expand **header**, click **TaskTypeCode**, and then click **OK**.

## Create the Then Statement

In this section, you create the following Then statement:

- Route the service request to Shipping Fulfillment System A.

Create the Then statement:

1. In the Then area, click **Add Action**, and then click **Assert New**.
2. Click **Select a Target**, and then click **DOOExternalInterfaceLayer.Result**.
3. Click **Edit Properties**.
4. In the Edit Properties dialog, in the ResultObj row, enter the following values, and then click **OK**.

Name	Value
ResultObj	"Shipping Fulfillment System A" You must include the double quotation marks.
ResultObjKey	"SERVICE_NAME" You must include the double quotation marks.

5. Click **Add Action**, and then click **Assert New**.
6. Click **Select a Target**, and then click **DOOExternalInterfaceLayer.Result**.
7. Click **Edit Properties**.
8. In the Edit Properties dialog, in the ResultObj row, enter the following values, and then click **OK**.

Name	Value
ResultObj	"Yes" You must include the double quotation marks.
ResultObjKey	"RESOLVE_XREF_FOR_CUSTOMERINFO" You must include the double quotation marks.  The value that you set for the RESOLVE_XREF_FOR_CUSTOMERINFO parameter instructs the cross-reference service to do one of the following: <ul style="list-style-type: none"> <li>○ Yes. Return the Oracle Application keys.</li> <li>○ No. Return the fulfillment system keys.</li> </ul> If Order Management cannot find customer cross-references for the fulfillment system in an Oracle Application, then you must set the value of this parameter to Yes.

9. Click **Save and Close**.

### *Related Topics*

- [Cross-Referencing in Order Management: Explained](#)
- [Creating Business Rules: Demo](#)

## Using Integration Cloud Service to Integrate Order Management with Other Systems

### Using Integration Cloud Service to Integrate Order Management with Other Systems: Overview

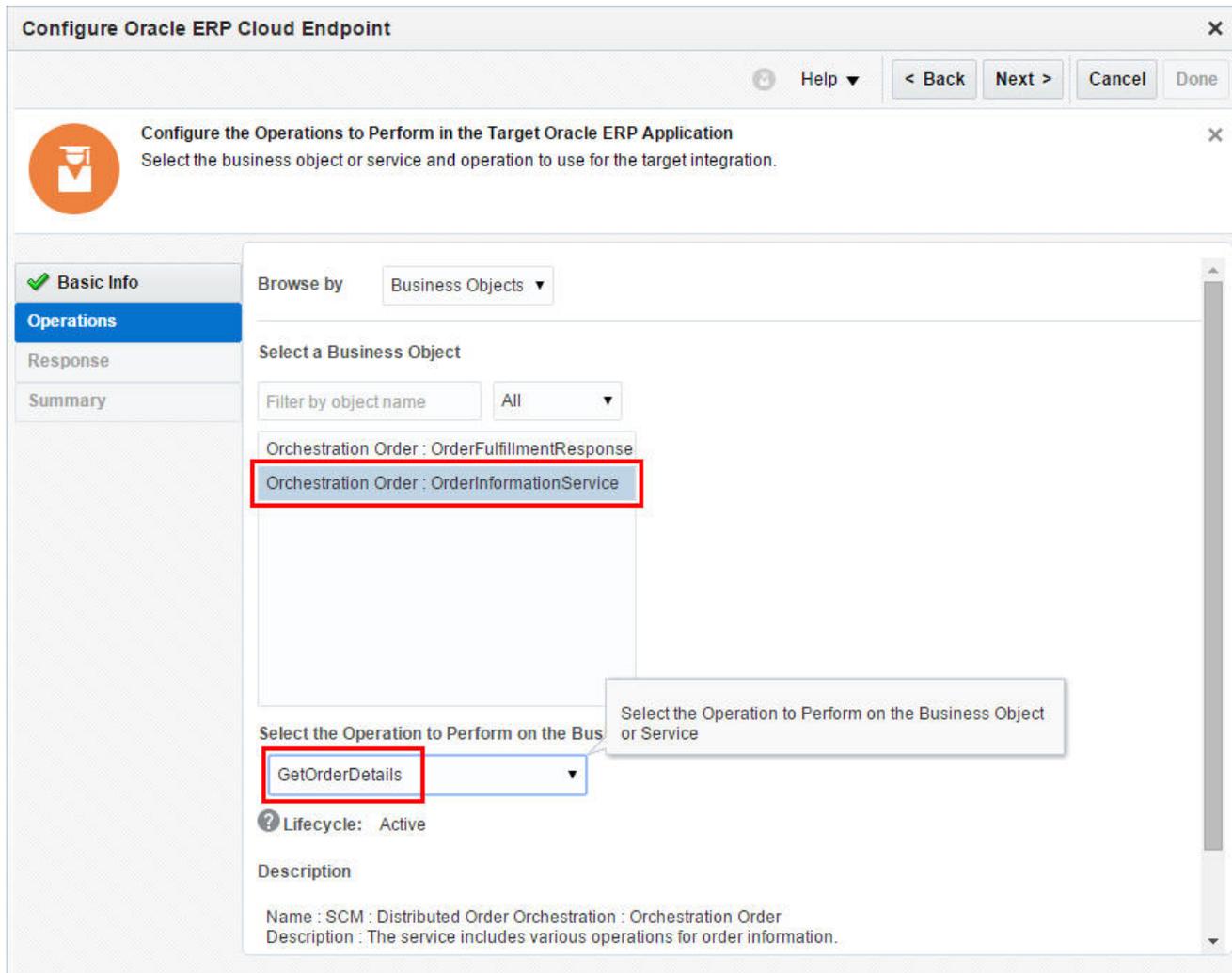
You can use Oracle Integration Cloud Service to integrate Order Management Cloud with some other Oracle or non-Oracle application. Oracle provides Integration Cloud Service as an integration platform in the cloud to help you maximize the value of your investments in SaaS (Software as a Service) and on-premise applications.

You can leverage the convenience and ease of use that Integration Cloud Service provides to create integrated business processes that operate across multiple application systems in the cloud or on premise, without having to rely on dedicated information technology resources. You can use Integration Cloud Service to integrate Oracle ERP with your commercial applications and software solutions.

You typically require specialized information technology staff to create an integration between applications. Integration Cloud Service simplifies this work so someone who does not have this level of technical expertise can create the flow. A business analyst can create the flow and map the attributes from the Order Information Service to the web service on the channel that your order fulfillment system uses. The analyst can use a WYSIWYG (what you see is what you get) interface to create the mapping.

## Specifying the Data to Communicate

The following diagram illustrates how you can configure the Oracle ERP Cloud Endpoint in Integration Cloud Service to administer application products, including published web services and business events for products that reside in the Supply Chain Management domain.



Order Management publishes the Orchestration Order business object and the service OrderInformationService to the catalog in the Order ERP cloud. You can choose this business object and service, and then choose one of the following operations:

- **GetOrderDetails.** Get details about the progress of a sales order from Order Management.
- **ChangeOrderAllowed.** Determine whether Order Management allows changes on the sales order or an order line.
- **ChangeOrderAttributeAllowed.** Determine whether Order Management allows changes on a sales order attribute.

You can also use the service OrderImportService to send an order revision to Order Management.

For details about how to specify the data to communicate, see the Creating Connections chapter and Creating Integrations chapter of the guide Using Oracle Integration Cloud Service on Oracle Help Center at [https://docs.oracle.com/cloud/latest/intcs\\_gs/ICSUG](https://docs.oracle.com/cloud/latest/intcs_gs/ICSUG).

## Mapping Source to Target

Integration Cloud Service provides a drag and drop interface that you can use to map the source payload to the target payload. The following diagram includes part of the mapping for the response payload of the GetOrderDetails operation of the Order Information Service, to a target web service.

The screenshot shows the Oracle Integration Cloud Service Mapper interface. The title bar reads "ORACLE Integration Cloud Service" and "Map: GetOrderDetailsResponse to invokeOrderInformationServiceResponse". The interface is divided into two main panes: "Source" and "Target".

**Source Pane:** Lists data elements from the "GetOrderDetailsResponse" source. Elements include "Order", "SourceTransactionIdentifier", "SourceTransactionSystem", "RevisionSourceSystem", "SourceTransactionNumber", "BuyingPartyIdentifier", "BuyingPartyName", "BuyingPartyNumber", "BuyingPartyOrigSystemReference", "BuyingPartyContactIdentifier", "BuyingPartyContactName", "BuyingPartyContactNumber", "BuyingPartyContactOrigSystemReference", "PreferredSoldToContactPointIdentifier", "PreferredSoldToContactPointOrigSystemReference", "CustomerPONumber", "TransactionalCurrencyCode", "TransactionalCurrencyName", "TransactionOn", "RequestingBusinessUnitIdentifier", "RequestingBusinessUnitName", "TransactionTypeCode", "CurrencyConversionType", "CurrencyConversionRate", "CurrencyConversionDate", and "TransactionDocumentTypeCode".

**Target Pane:** Lists data elements from the "invokeOrderInformationServiceResponse" target. Elements include "result", "HeaderIdentifier", "OrderNumber", "SalesOrderId", "SalesOrderNumber", "OrderedDate", "TransactionCurrencyCode", "ComplianceInformation", "ComplianceReason", "ComplianceValue", "SalesOrderSystem", "StatusCode", "ICSGetOrderDetailsResponseLine", "OpenFlag", "SalesOrderId", "SalesOrderNumber", "SalesFulfillLineIdentifier", "SalesFulfillLineNumber", "Status Code", "OrderedQty", "OrderedUom", "PackingInstruction", "ShippingInstruction", "ShippingCost", "SalesOrderLineId", and "SalesOrderLineNumber".

**Mapping Table:** A table in the center shows the mapping between source and target elements. A red box highlights the following mappings:

Source Element	Target Element
HeaderIdentifier	HeaderIdentifier
OrderNumber	OrderNumber
SalesOrderId	SourceTransactionIdentifier
SalesOrderNumber	SourceTransactionNumber

Green checkmarks are visible next to the source elements in the Source pane and the target elements in the Target pane that have been mapped.

If you map a data element on the source to an element on the target, then this interface adds a green check mark to the source and target. Note the following points:

- The target also displays the name of the data element at the source.
- The mapping applies to one target web service, so you must perform this mapping for each target system.
- If your deployment includes more than one channel system that must call GetOrderDetails, or if it must perform other operations under the Order Information Service, then you must create an integration in Integration Cloud Service, and then map the source payload to the target payload for each channel. However, you only require one connection between the ERP Service Catalog and Integration Cloud Service because it can support an integration under each web service that you publish under the catalog with a WSDL (Web Services Description Language).

For details about how to create a mapping, see the Mapping Data chapter of the guide Oracle Cloud, Using the Oracle Mapper on Oracle Help Center at [https://docs.oracle.com/cloud/latest/intcs\\_gs/OCMAP](https://docs.oracle.com/cloud/latest/intcs_gs/OCMAP).

## Exposing Events in Oracle ERP Cloud Adapter

Oracle ERP Cloud adapter exposes events so that you can achieve the following results:

- Send a notification to a subscriber when a condition occurs.

- Use the setup data that already exists for conditions.
- Use one integration for each subscriber system. This subscriber can use a single web service that listens for the triggering conditions through the Order Status Updated event.
- Allow the subscriber to request data enrichment in the integration.

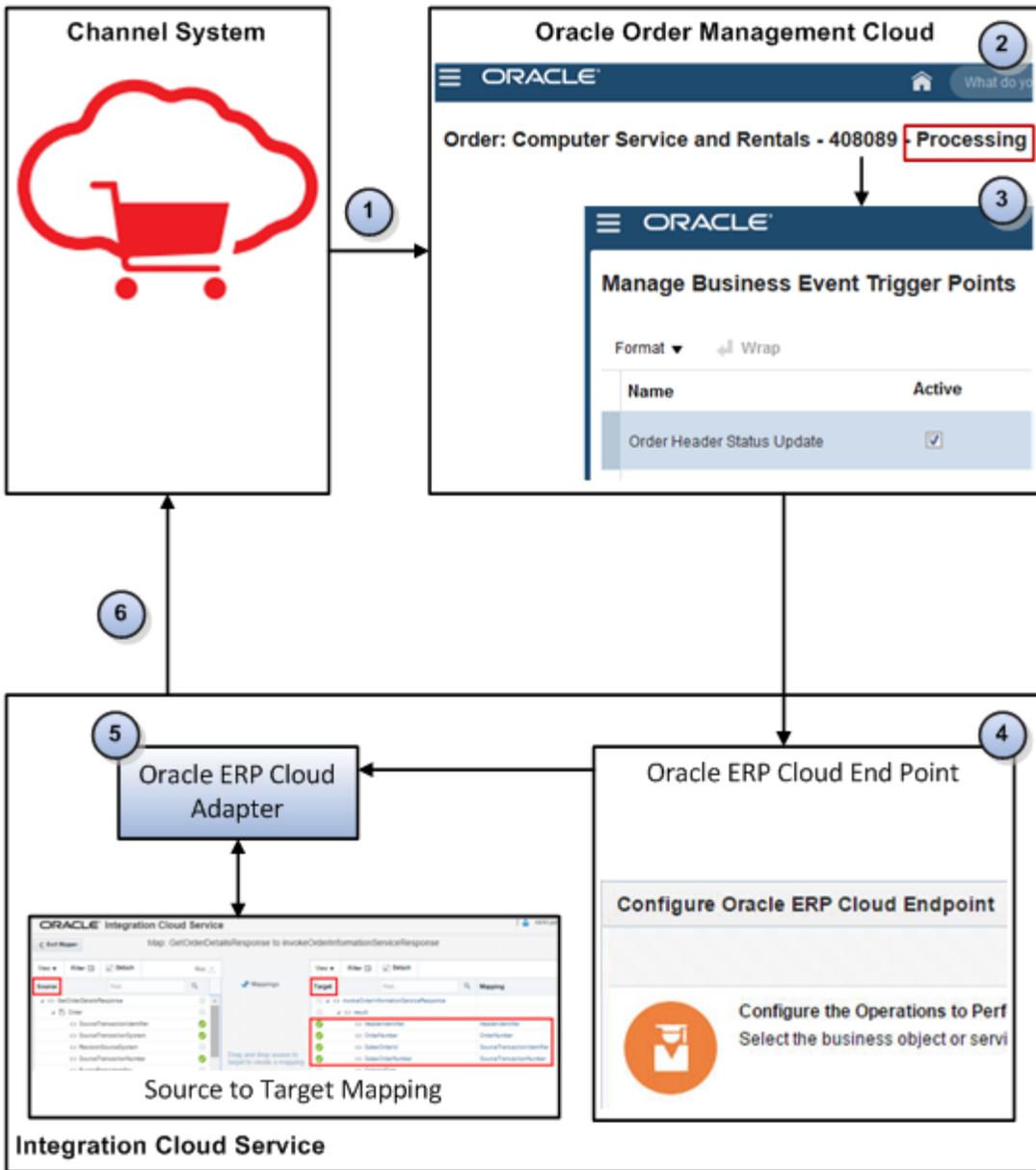
If you create an integration that subscribes to the Order Status Updated business event, then Order Management can send a notification each time one of the following conditions occurs:

- Update order header status.
- Update fulfillment line status.
- Close fulfillment line.

## Using Integration Cloud Service to Integrate Order Management with Other Systems: How It Works

An Order Management Cloud implementation that includes Oracle Integration Cloud Service uses the setup data for the business event trigger points so that it can send a notification to a subscriber when the order status changes. The subscriber can also request to get more details about the sales order.

The following diagram illustrates an example flow that uses Integration Cloud Service to integrate Order Management with a channel system. The diagram includes a channel system that resides in the cloud. However, Integration Cloud Service can integrate Order Management with a cloud or noncloud channel system.



### Explanation of Callouts

Assume Integration Cloud Service uses the following example flow to integrate Order Management with your subscribers. Assume you configure this integration to broadcast the Order Status Updated Event to subscribers when Order Management updates the order status, such as from Processing to Closed:

1. A channel system receives the shipment for a sales order, and then sends a notification to Order Management regarding the successful delivery.

2. Order Management changes the order status from Processing to Closed.
3. Order Management raises a business event. It examines the configuration on the Manage Business Event Trigger Points page and finds that you configured the Order Header Status Update event so that it is Active, and is also configured to raise an event when the order header status updates to Closed.
4. Assume you configured the Oracle ERP Cloud Adapter to do the following:
  - o Use the OrderInformationService to perform the GetOrderDetails operation.
  - o Mapped order attributes in the GetOrderDetailsResponse source payload to the InvokeOrderInformationServiceResponse target payload.
  - o Listen for the Order Status Updated events that are occurring on `https://my_server.com:9999/fndAppCoreServices/ServiceCatalogService?wsdl`.

The Oracle ERP Cloud Adapter recognizes the event, and then uses the configuration that you specified for the Oracle ERP Cloud Endpoint to determine the service and operation to use.

5. The Oracle ERP Cloud Adapter reads the source to target mapping that you created to determine how to map the source payload to the target payload.
6. Integration Cloud Service sends a notification to the subscriber at the channel system.

## Using Integration Cloud Service with Order Management to Communicate Events: Procedure

To use Oracle Integration Cloud Service with Order Management Cloud to communicate business events, you manage trigger points, create a connection, create an integration, track business events, and then test your work.

### Summary of the Work

Use Integration Cloud Service with Order Management to communicate events:

1. Manage business event trigger points.
2. Create a connection between Integration Cloud Service and Order Management.
3. Create an integration that monitors business events.
4. Track business events.
5. Test your work.

This topic includes example values. You might use different values, depending on your business requirements.

### Manage Business Event Trigger Points

Manage business event trigger points:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, click **Orders**, search for, and then open Manage Business Event Trigger Points.
4. Configure the trigger points so that Order Management raises a business event and sends a notification to each subscriber for each of the trigger points that your integration requires.

For example, send a notification to each subscriber when the status of the sales order header changes to Closed:

- o On the Manage Business Event Trigger Points page, click the **Order Header Status Update** row, and then make sure the Active option in this row contains a check mark.
- o In the Details area, in the Closed row, add a check mark to the **Raise Event** option.

- o Click **Save**.

### Manage Business Event Trigger Points to Send Status Updates on Fulfillment Lines

Configuring the trigger points so that Order Management sends a notification to each subscriber when the status updates on a fulfillment line requires that you modify the orchestration process definition.

Manage business event trigger points to send status updates on fulfillment lines:

1. On the Manage Business Event Trigger Points page, click the **Fulfillment Line Status Update** row, and then make sure the **Active** option in this row contains a check mark.
2. Click **Save and Close**.
3. On the Setup page, search for, and then open Manage Orchestration Process Definitions.
4. On the Manage Orchestration Process Definitions page, search for the orchestration process that your deployment uses.

For example, search for OrderFulfillmentGenericProcess.

Each orchestration process controls the status value for each fulfillment line, so you must modify the orchestration process that controls the status value. In this example, you modify the orchestration process that controls the shipping status value.

5. In the Search Results, click the **row** that contains OrderFulfillmentGenericProcess, click **Actions**, and then click **Edit**.
6. In the Process Details area, click **Status Conditions**, click **Fulfillment Line Status Values**, and then click **Edit Status Rule Set**.
7. On the Edit Status Rule Set page, add a check mark to the **Notify External System** option for each Status Value where your deployment must send a notification.

For example, to send a notification when the fulfillment line status changes to Shipped, add a check mark to the **Notify External System** option in the Shipped row.

8. Repeat step 7 for other status value, as necessary.  
Order management will send a notification when each fulfillment line that references this orchestration process definition reaches the status that you specify in steps 7 and 8.
9. Repeat steps 4 through 8 for each orchestration process definition in your deployment that updates status values, as necessary.

## Create a Connection Between Integration Cloud Service and Order Management

Create a connection between Integration Cloud Service and Order Management:

1. Log a service request so the Cloud Operations team can register the CSF-KEY (Credential Store Framework) that allows you to log into Integration Cloud Service. You must request a separate key for each order administrator who must access Integration Cloud Service. For details, see Requesting CSF Keys for Oracle Integration Cloud Service: Procedure.

2. Get access to the SOA Infra Operations Duty role.

The Order Administrator job role and the Supply Chain Application Administrator job role each come predefined so that they include SOA Infra Operations Duty.

Note the following points:

- o If you use a different role, such as Order Manager, then use the Security Console to get access to the SOA Infra Operations Duty role.
- o When you create a connection in Integration Cloud Service that you plan to use to publish the Order Status Updated event to Integration Cloud Service, you must specify the login credentials of an Oracle Cloud

Application user so that you can access the Event Catalog URL. You must assign the SOA Operator Role role to this Oracle Cloud Application user.

- o Assign the Order Manager job role so that this same user can call the web services that Order Management uses. Integration Cloud Service requires a single set of Oracle Cloud Application login credentials to define the integrations at design time, and to call the web services at runtime. If you use this connection to publish a business event to Integration Cloud Service, and to call Order Management web services, then the Oracle Cloud Application user must use the Order Manager job role and the SOA Infra Operations Duty role.

For details about job roles, see Assigning Job Roles That Allow Order Management to Communicate with External Applications: Explained.

3. Log into Integration Cloud Service.
4. On the Welcome page, click **Create Connections**.
5. On the Connections page, click **New Connections**.
6. In the Create Connection - Select Adapter dialog, click **Oracle ERP Cloud**.
7. In the New Connection - Information dialog, set the following values, and then click **Create**.

Attribute	Description
Connection Name	Enter any text that helps describe the connection. For example, enter Connection_ for_ Order_ Status_ Update.
Identifier	Enter any text that helps describe the connection. For example, enter CONNECTION_ FOR_ ORDER_ STATUS_ UPDATE.
Connection Role	Select Trigger and Invoke.
Description	Enter any text that helps describe the connection. For example, enter Connection for the order status update.

8. On the page that displays, click **Configure Connectivity**.
9. In the dialog that displays, set the following values, and then click **OK**.

Attribute	Description
ERP Services Catalog WSDL URL	Enter the URL that locates the WSDL. Use the following format:  https: //server: port/ fndAppCoreServices/ ServiceCatalogServicewsdl  For example:  https://my_ server. com: 9999/ fndAppCoreServices/ ServiceCatalogServicewsdl
ERP Events Catalog URL (optional)	Enter the URL that locates the events catalog. Use the following format:  http: //server: port/soa-infra  For example: http: //myserver com: 7818/ soa-infra

These URLs allow Integration Cloud Service to connect to Oracle ERP so that Integration Cloud Service can get details about the services and events that are available in Oracle ERP Cloud. It is recommended that you contact your system administrator to determine the URLs that you must use.

10. Click **Configure Security**.

11. In the Credentials dialog, enter the user name and password that you use to access Order Management as an administrator, and then click **OK**.
12. At the top of the page, click **Test**, and then wait for the indicator that displays immediately to the right of Test to change to a green color, and to display 100%.
13. Click **Save**, and then click **Exit Connection**.

## Create an Integration That Monitors Business Events

In this section, you create an integration that resembles the following setup:



Create an integration that monitors business events:

1. On the Connections page, Click **Integrations**.
2. On the Integrations page, Click **New Integration**.
3. In the Create Integration - Select dialog, click **Basic Publish to ICS**.

This procedure provides a partial integration example. In an actual integration, it is more likely you will choose Map My Data. To create a full, end-to-end configuration, see the Integration Cloud Service documentation.

4. In the Create New Integration dialog, set the following values, and then click **Create**.

Attribute	Description
What do you want to call your integration?	Enter any text that helps describe the connection. For example, enter Connection_for_Order_Status_Update.
Identifier	Enter any text that helps describe the connection. For example, enter CONNECTION_FOR_ORDER_STATUS_UPDATE.
Version	Accept the value that displays.
What does your integration do?	Enter any text that helps describe the connection. For example, enter Connection for the order status update.
Which package does your integration belong to?	Leave empty.

5. On the page that displays, you will identify the source of the connection that provides the details. For this example, in the search window, enter **Connection\_for\_Order\_Status\_Update**, which is the integration that you created earlier, and then click **ENTER**.
6. Drag, and then drop **Connection\_for\_Order\_Status\_Update** from the search results onto the **Drag and Drop a Trigger** area.

7. In the Oracle ERP Cloud Endpoint Configuration Wizard, you specify the events and scenarios that reside in the ERP Cloud connection. Set the following values, and then click **Next**.

Attribute	Description
What do you want to call your endpoint?	Enter any text that helps describe the endpoint. For example, enter PublishOrderStatusUpdated.
What does this endpoint do?	Enter any text that helps describe the connection. For example, enter Publishes the OrderStatusUpdated event.

8. Choose the **With Business Events** option.

Integration Cloud Service uses the URLs that you defined earlier to get the event catalog, and then display it in the Business Event for Subscription list. This list will help you to choose from the events that are available in Oracle ERP Cloud.

9. In the Business Event for Subscription list, enter `order`, click **Order Status Updated**, and then click **Next**.
10. On the page named Configure the Response to Send to the Oracle ERP Cloud Application, note that Integration Cloud Service is only listening for events in this configuration, and then sends them to subscribers without replying to Order Management. So, click **None**, and then click **Next**.
11. On the Summary page, click **Done**.

## Track Business Events

In this section, you set up Integration Cloud Service to track business events so that you test, and then monitor your configuration. It is recommended that you track at least one business event so that you can monitor your deployment during normal operations.

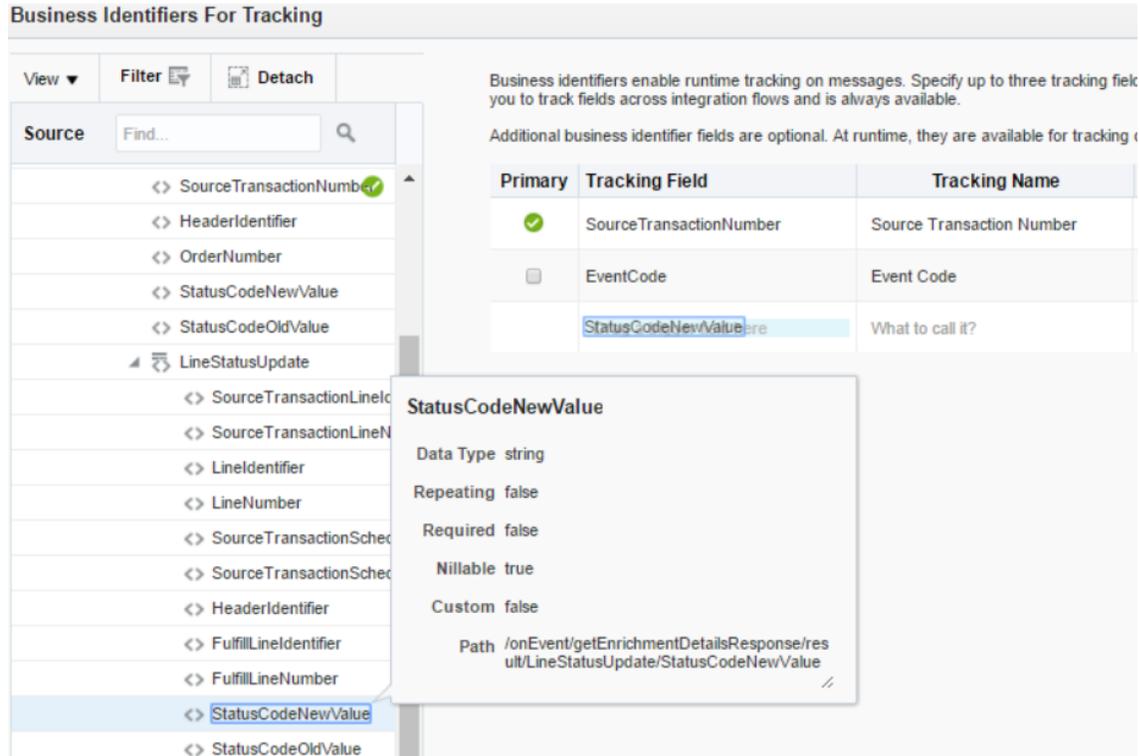
Track business events:

1. On the page that displays, click **Tracking**.

You can use Tracking for testing purposes. It allows you to view the business events that your integration raises in the monitoring system that Integration Cloud Service provides.

2. For this example, on the Business Identifiers for Tracking page, in the Source tree, under the result branch, click **Load More**.

You will choose the following business identifiers to track:



3. Drag, and then drop SourceTransactionNumber from the hierarchy tree to the first row of the **Tracking Field** column.
4. Drag, and then drop **Event** from the hierarchy tree to the second row of the **Tracking Field** column.
5. In the result branch, expand **LineStatusUpdate**, and then click **Load More**.
6. Drag, and then drop **StatusCodeNewValue** from the hierarchy tree to the third row of the **Tracking Field** column.
7. Click **Done**.
8. On the page that displays, click **Save**, and then click **Exit Integration**.
9. On the Integrations page, in the Connection\_for\_Order\_Status\_Update row, click **Pending Activation**.
10. In the Confirmation dialog, add a check mark to the **Enable Tracing** option, and then click **Yes**.

## Test Your Work

Test your work:

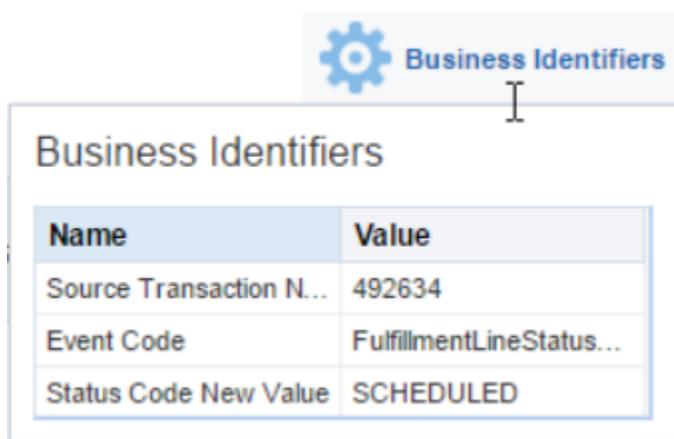
1. Log into Order Management, create a sales order, add one order line, use the Shipment Details tab to set the Requested Date to a time that occurs in the future, and then click **Submit**.
2. In the Confirmation dialog, make a note of the order number that displays, such as 492634, and then click **OK**.  
The shipment step of an orchestration process runs, sets the Status to scheduled, and then displays this status on the order line in the Order Lines tab. This shipment step raises the business event that you specified earlier. Note that you must have already configured this orchestration process and finished other setup work in Order Management.
3. Navigate back to Integration Cloud Service, and then click **Monitoring**.
4. On the Integrations Dashboard, click **Tracking**, and then click anywhere in the Messages area.

5. In the search window, enter the order number that you noted earlier, such as **492634**, and then click **ENTER**.
6. In the search results, notice that Order Management raised multiple events for the sales order. Click the **event** at the top of the list, which is the most recent event.
7. On the page that displays, click **Business Identifiers**, and then verify that the tracking information you set up for tracking earlier displays.

In this example, you set up tracking for the following items:

- o **SourceTransactionNumber.** Identifies the order number.
- o **Event.** Identifies the business event that gets triggered, which is FulfillmentLineStatusUpdate.
- o **StatusCodeNewValue.** Identifies the new status value of the fulfillment line, which is Scheduled.

For example:



Name	Value
Source Transaction N...	492634
Event Code	FulfillmentLineStatus...
Status Code New Value	SCHEDULED

#### Related Topics

- [Security Console: Overview](#)

## Filtering Details from Integration Cloud Service: Procedure

The setup that you perform on the Manage Business Event Trigger Points page is independent of the systems that monitor these business event notifications. If you must control the event notification that your channel or fulfillment system receives at a finer level, then you can use the filter capability in Oracle Integration Cloud Service to filter this data.

Filter details from the Integration Cloud Service:

1. Use the Manage Business Event Trigger Points page to set up the trigger points that enable the Order Status Updated event.
2. Create the integration in Integration Cloud Service.

For details, see the following resources on the Oracle Help Center:

- o [Integrating Order Management with Integration Cloud Service to Communicate Events: Procedure](#)

- o Tutorials for Oracle Integration Cloud Service at [http://docs.oracle.com/cloud/latest/intcs\\_gs/intcs\\_tutorials.htm](http://docs.oracle.com/cloud/latest/intcs_gs/intcs_tutorials.htm)
- o Videos for Oracle Integration Cloud Service at [http://docs.oracle.com/cloud/latest/intcs\\_gs/intcs\\_videos.htm](http://docs.oracle.com/cloud/latest/intcs_gs/intcs_videos.htm)
- o Developing Integration Flows with Integration Cloud Service at [http://www.oracle.com/webfolder/technetwork/tutorials/obe/cloud/ics/DevelopingIntegrationFlowswithIntegrationCloudServicesOBE/ICS\\_OBE.html](http://www.oracle.com/webfolder/technetwork/tutorials/obe/cloud/ics/DevelopingIntegrationFlowswithIntegrationCloudServicesOBE/ICS_OBE.html)

**3.** Log into Integration Cloud Service and add the filters:

- o Open the mapping that maps the integration to Order Management Cloud.
- o To allow events that only originate for sales orders from Order Management, create a filter on the Source Transaction System attribute.
- o To ignore events for a trigger that you define on the Manage Business Event Trigger Points page, create a filter on the Event Code attribute for each trigger that you must filter.

For details, see the following topics in the Using Oracle Integration Cloud Service section on the Oracle Help Center:

- o Configuring Oracle ERP Cloud Trigger Request Properties at [http://docs.oracle.com/cloud/latest/intcs\\_gs/ICSUG/GUID-4CE0BA8B-D4CD-46BA-BF80-9F8732469B48.htm#ICSUG-GUID-4CE0BA8B-D4CD-46BA-BF80-9F8732469B48](http://docs.oracle.com/cloud/latest/intcs_gs/ICSUG/GUID-4CE0BA8B-D4CD-46BA-BF80-9F8732469B48.htm#ICSUG-GUID-4CE0BA8B-D4CD-46BA-BF80-9F8732469B48)
- o Configuring Oracle HCM Cloud Trigger Request Properties at [http://docs.oracle.com/cloud/latest/intcs\\_gs/ICSUG/GUID-4CE0BA8B-D4CD-46BA-BF80-9F8732469B48.htm#ICSUG-GUID-4CE0BA8B-D4CD-46BA-BF80-9F8732469B48](http://docs.oracle.com/cloud/latest/intcs_gs/ICSUG/GUID-4CE0BA8B-D4CD-46BA-BF80-9F8732469B48.htm#ICSUG-GUID-4CE0BA8B-D4CD-46BA-BF80-9F8732469B48)
- o Creating Routing Expression Logic in Both Expression Mode and Condition Mode in Creating Integrations at [http://docs.oracle.com/cloud/latest/intcs\\_gs/ICSUG/GUID-F66ABE41-0A69-4A73-BC3D-0CE36FA78C4D.htm#ICSUG-GUID-9A4F5DE9-C549-4E01-A6F4-39EFE9729CCD](http://docs.oracle.com/cloud/latest/intcs_gs/ICSUG/GUID-F66ABE41-0A69-4A73-BC3D-0CE36FA78C4D.htm#ICSUG-GUID-9A4F5DE9-C549-4E01-A6F4-39EFE9729CCD). You can use different attributes, such as Country, SourceTransactionSystem, or EventCode, to create an expression.

## Requesting CSF Keys for Integration Cloud Service: Procedure

The CSF-KEY (Credential Store Framework) key allows you to log into Oracle Integration Cloud Service. You must log a service request so the Cloud Operations team can register the CSF-KEY.

Request the CSF-KEY for Oracle Integration Cloud Service:

- 1.** Create a service request that requests Oracle to create the CSF-KEY. Include the following details in your request.

Service Request Section	Description
Subject line	Include the following text:
Body	Provide the identity domain that your integration can use for Order Management Cloud and for Oracle Integration Cloud Service subscriptions. For example, icssvc. identity. domainidm2152.  Request a separate CSF-KEY for each order administrator who must access Integration Cloud Service.

- 2.** Wait for Oracle to reply to your service request. When Oracle does reply, provide them with the user name and password that you use to access Integration Cloud Service. Oracle will then create the CSF-KEY.

## More Information About Integrating Order Management

### Assigning Job Roles That Allow Order Management to Communicate with External Applications: Explained

A duty role allows Order Management Cloud to access the web services that it uses to communicate with an external system according to the user who is currently logged in. You must make sure that any custom job roles you create reference the duty roles that Order Management requires.

For example, if a user releases a sales order for shipping, then the job role that this user uses must reference the duty roles that allow Order Management to call the shipping system and to receive information about the shipment from this shipping system. In this example, you must assign a job role that references the following duty roles:

- Shipment Request Processing Web Service Duty
- Orchestration Order Shipping Web Service Duty

#### Job Roles

The following table describes the job roles that come predefined with Order Management and the duty roles that each job role includes.

Job Role	Duty Role
Order Entry Specialist	This job role includes the following duty roles: <ul style="list-style-type: none"> <li>• FSCM Load Interface Administration Duty</li> <li>• Item Inquiry Duty</li> </ul>
Order Manager	This job role includes the following duty roles: <ul style="list-style-type: none"> <li>• Orchestration Order Monitoring Duty</li> <li>• Orchestration Order Management Duty</li> <li>• Orchestration Order Scheduling Duty</li> </ul>
Order Administrator	Each of these job roles include the following duty roles: <ul style="list-style-type: none"> <li>• Orchestration Order Administration Duty</li> <li>• Orchestration Infrastructure Administration Duty</li> </ul>
Supply Chain Application Administrator	

#### Duty Roles

The following table describes the duty roles that allow Order Management to communicate with an external system. The Description column describes the communication that the web service provides.

Duty Role	Description
Order Orchestration Decomposition Web Service Duty	Communicate with source systems so that Order Management can separate source orders during transformation.
Orchestration Order Activity Management Web Service Duty	Communicate with a fulfillment system that can process an activity.
Orchestration Order Billing Web Service Duty	Communicate with a system that processes the billing for each sales order.
Orchestration Order External Integration Web Service Duty	Allow an external system to call Order Management.
Orchestration Order Fulfillment Web Service Duty	Allow a fulfillment system to send status updates to Order Management through the fulfillment task layer.
Orchestration Order Inquiry Web Service Duty	Extract order details from a sales order.
Orchestration Order Receiving Web Service Duty	Receive a return for a sales order.
Orchestration Order Shipping Web Service Duty	Process the shipment of a sales order.
Orchestration Order Template Web Service Duty	Allow a fulfillment system to send status updates for fulfillment tasks through the template task layer.

For details about job roles and duties, see the Oracle Fusion Applications Order Orchestration Security Reference Manual on Oracle Applications Help at <https://fusionhelp.oracle.com>.

## Integrating Order Management to Communicate Attachments: Explained

Order Management Cloud allows users to add an attachment to a sales order, such as a document that includes requirements for manufacturing, a memo that includes guidelines for negotiating a price, or a URL to a page that includes item installation instructions. You must integrate Order Management so that it can add these attachments.

Note the following:

- Users can view attachments in the Order Management work area and send them to a fulfillment system.
- Order Management can receive an attachment from a source system and include it as part of a sales order.
- Order Management cannot receive an attachment from a fulfillment system.
- Order Management send an attachment to an order capture system.

To allow Order Management to receive and send sales order attachments, a Supply Chain Administrator or Controller must do the following work:

Functionality	Description
Allow Order Management to receive sales order attachments from an order capture system.	Collect the attachment category during orchestration data collection.
Allow Order Management to send sales order attachments to a fulfillment system.	Call the AttachmentsAM public service. Use this service to select and send attachments according to the type of the fulfillment request and the attachment category.

For more information, see the document that describes attachments in Article ID 1536633.1 on My Oracle Support at <https://support.oracle.com>.

## Collecting Orchestration Data: Explained

Order Management Cloud cross-references attributes and data that reside in an order capture system and each fulfillment system so that it can use a single representation across these different systems. The order capture systems and fulfillment systems use these values to communicate with Order Management. You must use the Oracle Fusion Customer and Product Models to set up customer data, product master data, and cross-references.

Make sure you collect the following reference entities for validation and cross-referencing:

- Activity types
- Currencies, conversions, and types
- Demand class
- Document category
- FOB points
- Freight term
- Invoicing and accounting rules
- Payment methods
- Payment terms
- Receipt methods
- Return reason
- Resources
- Sales credit types
- Shipment priority
- Shipping carriers
- Shipping class of service
- Shipping mode of transport
- Tax classification code
- Tax exemption reason
- Units of measure
- Warehouse

These attributes are not related to the customer or the item.

Oracle provides the following data collection tools:

- **Continuous collection.** Collects data incrementally. It provides fast collection for each entity that Order Management must source.
- **Targeted collection.** Refreshes data for a selected business object. You can run a scheduled process that performs targeted collection as needed or on a schedule.

## Mapping Names and Codes from Order Capture Systems: Explained

If your deployment includes an external order capture system, then you must map names and codes from this system into Order Management Cloud.

An orchestration reference object is an object that resides in the set of objects that an orchestration process processes so that it can determine the meaning of a name or the description of a code, such as a payment term name, freight code, or transport code. An order capture system typically sends sales order data that contains names or codes to an orchestration process, and this orchestration process must display a meaning for the name or a description for the code. You must collect the data that determines these meanings and descriptions in the Order Orchestration and Planning Data repository.

For example, assume an external order capture system sends sales order data that includes a payment term of 2/10, Net 30 to an orchestration process, and the Order Orchestration and Planning Data repository contains a record that includes a payment term of 2/10, Net 30. In this situation, the orchestration process uses the matching codes to identify the following payment term description:

**2% discount earned if paid within 10 days**

To get the complete list of orchestration reference objects, you can examine the collected data for them, and view the list of values for the Lookup Type field.



# 4 Importing and Transforming Source Orders

## Importing Orders

### Order Import Template: Explained

You can use the Order Import Template to help reduce errors and simplify order import. This template contains a structure that the Oracle database requires. It includes a tab for each database table, and it displays these tabs in a specific sequence. It includes columns on each tab that represent the table columns that Oracle requires, and it specifies the data type that Oracle requires for each database column.

To get a copy of the Order Import Template, and to get details about the tables that the template references, see the Import Sales Orders topic in the File-Based Data Import for Oracle Supply Chain Management Cloud guide on Oracle Help Center at [http://docs-uat.us.oracle.com/cloud/latest/scmcs\\_gs/OEFSC/Import\\_Sales\\_Orders\\_433316\\_fbd\\_16.htm#433316](http://docs-uat.us.oracle.com/cloud/latest/scmcs_gs/OEFSC/Import_Sales_Orders_433316_fbd_16.htm#433316).

For details about how to use the Order Import Template, see Using Files to Import Orders into Order Management: Procedure.

You can also use a web service to create an integration that imports source orders. For details, see Article ID 2051640.1 (Using Web Services with Oracle Fusion Order Management Cloud) on My Oracle Support at <https://support.oracle.com>.

### Guidelines for Using the Order Import Template

It is recommended that you use the following guidelines when you use the Order Import Template:

- The Order Import Template uses a separate spreadsheet in an Excel workbook to represent each interface table. A row near the top of each spreadsheet contains column headers. Each column header represents the name of an interface table column. The spreadsheet displays these columns in a specific sequence. You must not modify this sequence. If you do, then the import will fail.
- The Order Import Template comes predefined with example data already populated, and some column headers include descriptive text. Use this example data and the descriptive text to help you determine the type of data that you must include. For example, click the DOO\_ORDER\_LINES\_ALL\_INT tab, and notice that row five in the Source Transaction System column includes a value of LEG, which is a text abbreviation for the term legacy. Position your mouse over the Source Transaction System column header, and then read the descriptive text that indicates this field must contain only VARCHAR2(30) type data. The value LEG meets this requirement.
- The data that you enter into the spreadsheet must use a format that the template can accept and process according to the format that the database column uses. For example, data that you enter in the Source Transaction Identifier column of the DOO\_ORDER\_LINES\_ALL\_INT tab must use the VARCHAR2(50) format. For more information about these formats, see the topic about Oracle data types at [https://docs.oracle.com/cd/A57673\\_01/DOC/server/doc/SCN73/ch6.htm](https://docs.oracle.com/cd/A57673_01/DOC/server/doc/SCN73/ch6.htm).
- You must enter data for each required column. The template uses an asterisk ( \* ) to indicate each required column.
- Required columns are not always contiguous. It is recommended that you carefully examine the column headings in each spreadsheet to make sure you include data for each required column. Tip: to locate all the required headings on a spreadsheet, press **CTRL + F**, enter ~\* (a tilde and an asterisk), and then click **Find All**.
- The template uses a blue background color for column headings, by default. It uses other colors to group some columns. For example, the DOO\_ORDER\_HEADERS\_ALL\_INT sheet uses a tan color (red 253, green 233, and blue 217) to group the Buying Party Identifier, Buying Party Name, and Buying Party Number columns. You must

enter a value in at least one column of this color group. Note that these columns might not display contiguously. It is recommended that you carefully examine the color group on each spreadsheet to make sure you include at least one value for the color group.

- Some color groups are conditional. For example, you must enter a value for at least one column in the color group that represents the Requested Fulfillment Organization Identifier, Requested Fulfillment Organization Code, and Requested Fulfillment Organization Name columns on the DOO\_ORDER\_LINES\_ALL\_INT spreadsheet only for each row in this spreadsheet that references a return order. The spreadsheet includes an instructional note immediately above each conditional color group.
- Some sheets include more than one color group. You must enter at least one value for each color group.
- You must use the YYYY/MM/DD/HH/MI/SS format for each date field.
- You must use a comma (,) for the thousands separator and a period (.) for the decimal separator in the Amount column. For example, use 1,500.25. Do not use 1500,25.
- If a column must use a whole number, then each control file validates that the data in this column includes only whole numbers. If it does not, then the import fails.
- Use the descriptive text in the spreadsheet for assistance with columns that require an internal identifier.
- You can hide columns that you do not require, but you must not delete them. If you delete a column, then the import will fail.
- At a minimum, it is recommended that you include information in each of the following tables. If you do not include this information, then your sales orders will not be complete:
  - DOO\_ORDER\_HEADERS\_ALL\_INT
  - DOO\_ORDER\_LINES\_ALL\_INT
  - DOO\_ORDER\_TXN\_ATTRIBUTES\_INT

### Importing New Data

If you use a file to import a customer, contact, or address that does not already exist in Trading Community Architecture, then you must manually assign the following privileges to the role that the user who runs the scheduled process named Load Interface File for Import uses.

Privilege	Description
HZ_ENTER_TRADING_COMMUNITY_PERSON_PRIV	Enter Trading Community Person
HZ_UPDATE_TRADING_COMMUNITY_PERSON_PRIV	Update Trading Community Person
HZ_ENTER_TRADING_COMMUNITY_ORGANIZATION_INFORMATION_PRIV	Enter Trading Community Organization Information
HZ_UPDATE_TRADING_COMMUNITY_ORGANIZATION_PRIV	Update Trading Community Organization

For details about how to add a privilege, see [Data Security: Explained](#).

## Tables That the Order Import Template Contains

The Order Import Template includes a tab for each of the following interface tables.

Table Name	Information That This Table Must Contain
DOO_ORDER_HEADERS_ALL_INT	Order header data.
DOO_ORDER_HDRS_ALL_EFF_B_INT	Flexfield information for the order header.
DOO_ORDER_LINES_ALL_INT	Order line data.
DOO_ORDER_LINES_ALL_EFF_B_INT	Flexfield information for the order line.
DOO_ORDER_TXN_ATTRIBUTES_INT	Attributes that might be associated with an order line.
DOO_ORDER_PAYMENTS_INT	Payment details for the order line.
DOO_ORDER_LOT_SERIAL_NUMS_INT	Lot serial information for an order line.
DOO_ORDER_SALES_CREDITS_INT	Sales credits for the sales representative that the source order references.
DOO_ORDER_DOC_REFERENCES_INT	References to documents that Order Management imports with the source order.

### Related Topics

- [Data Security: Explained](#)

## Using Files to Import Orders into Order Management: Procedure

This topic describes how to use files to import source orders from a source system into Order Management Cloud.

### Summary of the Work

To use files to import source orders into Order Management, do the following work:

1. Convert your source data.
2. Create the import file.
3. Upload your source data.
4. Import your source data.
5. Delete imported orders from interface tables.

Order import generally uses this same flow that you use when you integrate Order Management with a source system, except you import orders from a spreadsheet. For details, see *Order-to-Cash with Order Capture Systems: How it Works*.

## Convert Your Source Data

You must convert your source data into a CSV (comma separate values) file that uses the same structure that the Order Import Template uses. Using this template helps to make sure that your converted data uses the same structure that the Oracle database requires. This topic describes one recommended way to convert your source data. The details of your conversion might require a different way. If you cannot use the Order Import Template for some reason, and if you are not familiar with doing data conversion, then it is recommended that you consult Oracle resources about how to use an open interface when importing data. For details about other ways to import your source data into Order Management, such as through web services, see *Using Public Services to Integrate with Order Management: Explained*.

Convert your source data into a CSV file:

1. Download the Order Import Template from the Import Sales Orders topic in the File-Based Data Import for Oracle Supply Chain Management Cloud guide on Oracle Help Center at [http://docs-uat.us.oracle.com/cloud/latest/scmcs\\_gs/OEFSC/Import\\_Sales\\_Orders\\_433316\\_fbdi\\_16.htm#433316](http://docs-uat.us.oracle.com/cloud/latest/scmcs_gs/OEFSC/Import_Sales_Orders_433316_fbdi_16.htm#433316).
2. Use a spreadsheet editor that can read a CSV file, such as Excel, to open the Order Import Template, and then familiarize yourself with the structure that it uses.

For details about this structure and guidelines for using the template, see *Order Import Template: Explained*.

3. Use a data manipulation tool to structure your source data so that it mirrors the structure that the Order Import Template contains, and then save this data to a CSV file.

You can use SQL (Structured Query Language), ODI (Oracle Data Integrator), or some other tool to convert your source data into a CSV file.

### Guidelines for Converting Your Source Data into a CSV File

You must create a CSV file that includes data for interface tables and columns when you do the conversion. Do the following work to make sure the structure that this CSV file contains mirrors the structure that the Order Import Template contains:

- Include the same table names and column names.
- Sequence the tables in the same sequential order. The tab sequence that this template uses determines this sequential order.
- Include the same columns in each table, and arrange these columns in the same sequential order inside each table.
- Use the same data type for each column.
- Include data for all required columns. The Order Import Template uses an asterisk ( \* ) to indicate required columns. For example, the Source Transaction Schedule Identifier column on the DOO\_ORDER\_LINES\_ALL\_INT tab is required.
- If your data includes an internal identifier, then make sure you map it correctly. The Import Template includes multiple columns that specify information for internal identifiers. It uses an \_ID suffix to identify these columns, such as SOURCE\_TRANSACTION\_ID or BUYING\_PARTY\_ID. For each internal identifier column, you must map your source data value to an Order Management value. If your source data includes an identifier column that does not require setup in the Product Information Management work area, Trading Community Architecture, or data collection, then you can use the implementation pages in your implementation project to get the identifier and other attributes that you must map for the internal identifier.

To view an example of this CSV structure, do the following work:

1. Open the Order Import Template in Microsoft Excel.
2. Click the **DOO\_ORDER\_HEADERS\_ALL\_INT** tab.
3. Choose **File, Save As**, and then click **Excel Workbook**.
4. Save the file as a .csv file, and then use a text editor to examine the output.

## Create the Import File

Create the import file:

1. Prepare the Order Import Template.
  - o Use a spreadsheet editor that can read a CSV file, such as Excel, to open the Order Import Template. For details, see Order Import Template: Explained.
  - o Delete the example data from the Order Import Template. This template comes with example data that helps you determine the type of data that you must include. For example, row four of the DOO\_ORDER\_HEADERS\_ALL\_INT tab includes example data. Make sure you delete all example data from all tables in the spreadsheet, even from tables that you do not require.
2. Copy and paste your source data into each of the tables in the Order Import Template:
  - o Use a spreadsheet editor to open the CSV file that contains your source data.
  - o Copy your order line data to the clipboard.
  - o In the Order Import Template, click the **DOO\_ORDER\_LINES\_ALL\_INT** tab.
  - o Click cell **A5**, and then paste your data.
  - o Examine the results to make sure you correctly pasted the data. For example, make sure the Product Description column contains VARCHAR data, and that the Ordered Quantity column contains NUMBER data.
  - o Continue copying data for each table until you finish copying all your source data into the Order Import Template.

It is recommended that you save your work after each copy.

3. Create the import file:
  - o Click the **Create CSV** tab, and then click **Generate CSV File**.  
If the Generate CSV File button is not active, then click **Developer** in the menu bar, and then click **Macros**. In the Macro dialog, choose **GenCSV**, and then click **Run**.
  - o Wait for the macro in Excel to finish running.  
When the macro finishes running, Excel displays a dialog that allows you to save a .zip file.
  - o In the save dialog, choose a location to save your .zip file, and then click **Save**.  
The macro creates a .zip file that includes a separate file for each table that the template contains.
  - o Optional. In subsequent save dialogs, save each individual CSV file.  
If you must import data for an individual table, then save the data as an individual CSV file; otherwise, click **Cancel**.

## Upload Your Source Data

Upload your source data:

1. Log into Order Management. You must use a login that includes order manager privileges.
2. In the Navigator, under Tools, click **Scheduled Processes**.
3. On the Scheduled Process page, click **Schedule New Process**.
4. In the Schedule New Process dialog, set the following field, and then click **OK**.

Field	Value
Name	Load Interface File for Import

Field	Value
-------	-------

5. In the Process Details dialog, set the following parameters.

Parameter	Description
Import Process	Choose Import Sales Orders.
Data File	<p>Do the following work:</p> <ol style="list-style-type: none"> <li>Click the <b>down arrow</b> in the Data File field.</li> <li>Scroll down, and then click <b>Upload a New File</b>.</li> <li>In the Upload File dialog, click <b>Choose File</b>.</li> <li>In your Windows Explorer window, locate and choose the .zip file that you created when you used the Order Import Template, and then click <b>Open</b>.</li> <li>In the Upload file dialog, click <b>OK</b>.</li> <li>In the Process Details dialog, make sure the Data File field displays the name of the .zip before you continue.</li> </ol>

6. Click **Submit**.

7. In the Confirmation dialog, note the value of the Process ID, click **OK**, and then click **Close**.

8. Click **Actions**, and then click **Refresh**.

9. Use the Process ID that you noted earlier to locate your scheduled process, and then make sure the Status field for this process displays Succeeded.

The Succeeded status indicates that the scheduled process successfully uploaded all of your source data. If the upload fails on any row, then the status displays Error. If the Search Results list does not display your process, then click **Refresh** until it does.

10. Correct errors, if necessary:

- o If the scheduled process ends in an error, then click the **Error** status in the Search Results list for your scheduled process, and then examine the log and output files to get details about the data that caused the error.
- o Use Excel to open the Order Import Template that contains your source data, and then correct the source data that is causing the error.
- o In the Order Import Template, click **Generate CSV File**, and then run the scheduled process again.

Repeat this step until the scheduled process successfully uploads all of your source data.

## Import Your Source Data

Import your source data:

1. Run the Import Sales Orders scheduled process. In the Process Details dialog, set one of the following parameters. To avoid a conflict in the data to import, it is recommended that you specify only one parameter.

Parameter	Description
Batch Name	Enter a value.
Source System	To import all records from the source system, enter the value that you set in the Source Transaction System column of the DOO_ORDER_LINES_ALL_INT tab of the Order Import Template. For example, enter LEG.

Parameter	Description
Source Order Number	To import the record for only one source order, enter the value that you set in the Source Transaction Identifier column of the DOO_ ORDER_ LINES_ ALL_ INT tab of the Order Import Template. For example, enter 12345.

2. Make sure the Status for the Import Sales Orders scheduled process that you submitted is Succeeded. The Succeeded status indicates that the scheduled process successfully imported all of your source data. If the import fails on any row, then the status displays Error.
3. Verify your import:
  - o In the Navigator, click **Order Management**.
  - o On the Order Management page, click **Tasks**, and then click **Manage Order Orchestration Messages**.
  - o Query for records that the import includes.
  - o Examine the errors in the spreadsheet, and then fix the source data in the Order Import Template that is causing them.
  - o Submit the modified data and make sure the scheduled process successfully imports all rows.
  - o Navigate to the Order Management work area, and then query for one or more of the orders that you imported.
  - o Make sure the work area displays the sales order, and that the order data is identical to the source data.

## Delete Imported Orders From Interface Tables

Order Management uses interface tables when you import data from a source system so that it can handle import errors and to retain a data backup during import. As an option, you can remove this data to save storage space.

Delete imported orders from interface tables:

1. In the Navigator, under Tools, click **Scheduled Processes**.
2. On the Scheduled Process page, click **Schedule New Process**.
3. In the Schedule New Process dialog, set the following field, and then click **OK**.

Field	Value
Name	Delete Orders From Interface Tables

4. In the Process Details dialog, set values, as necessary, and then click **Submit**.

### Related Topics

- [Order-to-Cash with Order Capture Systems: How it Works](#)

## How can I fix an order import that cannot find a matching configuration node?

You might encounter an error that is similar to the following when you import source orders:

**Cannot find a matching configuration node for item 12345**

This error might occur even if the node exists in a reference model that includes an instance type of Optional Single Instance. The Order Import process does a search that validates, and then creates the configuration structure that it includes in the sales order even when a node, such as Option Classes, is missing in the order import data that defines the configuration structure.

However, if a reference model defined as an Optional Single Instance resides in the root model, or if it is part of an option class, and if the order import data does not include the absolute path to the node, then the search will not find the underlying node.

To avoid this problem, you can modify the instance type for the reference model from Optional Single Instance to Required Single Instance. You do this modification in the Product Information Management work area. It will not affect the functional behavior. You must also set the system parameter Use Configurator for Order Import Validation to Yes.

For example, assume the following model exists in Product Information Management, and that the order import data includes M1, M1.M2, M1.M2.SI2:

```
M1
|_M2 (Optional Single Instance)
  |_OC1
  |_SI2
```

In this situation, the Order Import search will not find the complete structure, it will create an error during order import, and it will add an entry in the Order Import log that is similar to the following:

```
The order import process failed for source order source_order_identifier for the following reason: Cannot find a matching configuration node for item item_number on order line number order_line_number.
```

where:

- source\_order\_identifier identifies the source order
- item\_number identifies the item
- order\_line\_number identifies the order line

This entry indicates that the search could not find a matching component in the model reference node named Optional Single Instance.

For another example, assume the following model exists in Product Information Management, and that the Order Import data includes M1, M1.M2, M1.M2.SI1:

```
M1
|_OC1
  |_M2 (Optional Single Instance)
    |_SI1
```

In this situation, the Order Import search will fail in the same way that it failed in the first example.

## Exporting and Importing Setup Data with Inventory Organizations: Explained

If you use an inventory organization when you set up Order Management Cloud, then you must import the setup data of the offerings that include your inventory organizations, and then import the setup data from Order Management to your implementation instance. You do this work when you deploy your setup data from one application instance to another application instance.

You can use one of the following options when you export or import your setup data:

- **Create one configuration package.** This package contains the offerings that include the inventory organizations, and that also contains the Order Management offering.
- **Create multiple configuration packages.** Use a separate configuration package for the offering that includes the inventory organizations, and use another package for the Order Management offering.

## Using Web Services to Import Source Orders: Explained

An order capture service can receive a request for information about a fulfillment line from an order capture system that you have integrated with Order Management Cloud, and then transform it into a structure that Order Management can use.

You can use the Order Import web service to capture orders from a source system. For details, see Article ID 2051640.1 (Using Web Services with Oracle Fusion Order Management Cloud) on My Oracle Support at <https://support.oracle.com>.

You can also use the following order capture services:

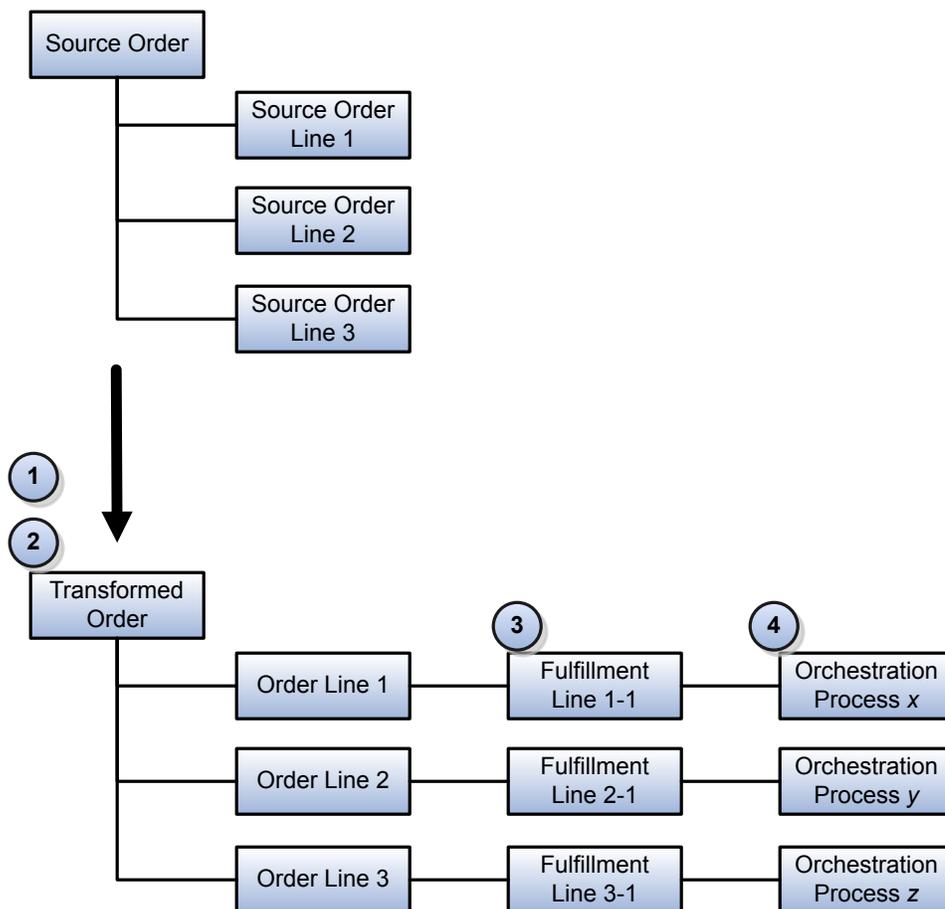
Service	Description
Get Order Details	<p>This service does the following work:</p> <ul style="list-style-type: none"> <li>• Communicates status information or order details to the order capture system that requests it.</li> <li>• Provides information about the entire sales order or a set of order lines.</li> <li>• Provides details about a sales order that originates in an integrated source system.</li> </ul> <p>You can specify a source system so that the query limits results to only this system. If you do not specify a source system, then the service might return order numbers from multiple source systems.</p>
Get Order Shipping Details	<p>This service does the following work:</p> <ul style="list-style-type: none"> <li>• Communicates the following shipping details to the order capture system that requests it: <ul style="list-style-type: none"> <li>○ Current schedules</li> <li>○ Status of each schedule</li> <li>○ Schedule details, such as warehouse and shipping method</li> </ul> </li> <li>• Organizes the reply according to shipments instead of fulfillment lines.</li> <li>• Includes information about fulfillment lines in the reply. Multiple fulfillment lines in Order Management might reference a single order line in the source order.</li> <li>• Returns details about the order lines of a sales order, or for only a subset of these lines.</li> </ul>
Apply a Hold	Routes a request for a hold from an order capture system to the Hold task layer. Processes multiple holds in a request, each hold for a single sales order, or for multiple order lines in this sales order.
Release a Hold	Routes a request to release an existing hold from an order capture system to the Hold task layer.
Check Availability	Sends a request to Order Management to determine the quantity that is available on a date at a source system. Order Management sends a reply to the order capture system that includes this information.

# Transforming Source Orders to Sales Orders

## Transforming Source Orders to Sales Orders: How It Works

Order Management Cloud transforms each source order that you create in Order Management or that it receives from a source system so that it can optimize order fulfillment.

The following diagram illustrates how Order Management transforms a source order.



### Explanation of Callouts

In this example, Order Management does the following work:

1. Receives a source order that you submit in Order Management or that it gets from a source system. The source order in this example includes three order lines.
2. Separates the source order into logical pieces that it can fulfill, such as fulfillment lines.
3. Assigns each fulfillment line to a new orchestration process that it creates to orchestrate fulfillment for this fulfillment line. For example, it might assign fulfillment line 1-1 to orchestration process 300100090333189, and fulfillment line 1-2 to orchestration process 300100090333210.

4. Starts each orchestration process that performs the tasks that fulfill each fulfillment line. To examine these orchestration processes, you can use the Switch to Fulfillment View action on the Order page of the Order Management work area.

Order Management typically finishes transformation within a few seconds after you click Submit.

Order Management sets the completion date of the final step in each orchestration process to the requested date of the source order. Note that the Last Fulfillment Completion Step specifies the final step of the orchestration process. This final step is not necessarily the last step that Order Management completes in chronological order.

## Parts of Sales Orders You Can Use After Order Management Transforms Source Orders

You can use the following parts of a sales order after Order Management transforms a source order. You can use them to monitor and manage order fulfillment.

Part of Sales Order	Description
Order	<p>Includes order lines and fulfillment lines. Note the following:</p> <ul style="list-style-type: none"> <li>• The relationship that exists between a sales order and a fulfillment line is similar to the relationship that exists between a source order and a source order line.</li> <li>• A sales order that resides in Order Management might not exactly mirror the structure of a source order that resides in an external source system.</li> </ul>
Order line	<p>Typically references one fulfillment line. Note the following:</p> <ul style="list-style-type: none"> <li>• If Order Management splits a fulfillment line, then it maps two or more fulfillment lines to the same order line.</li> <li>• The relationship that exists between a sales order and the order lines preserves some of the original structure of the source order, making it easier to keep track of the original intent of the source order, even if Order Management splits the fulfillment lines that it associates with the sales order.</li> </ul>
Fulfillment line	<p>Part of a sales order that you can modify in the Order Management work area. For example, you can do the following modifications:</p> <ul style="list-style-type: none"> <li>• Schedule or unschedule a fulfillment line.</li> <li>• Reserve items for a fulfillment line.</li> <li>• Substitute an item on a fulfillment line.</li> <li>• Change the warehouse, shipping method, or demand class for a fulfillment line.</li> </ul>
Orchestration process	<p>An order administrator can set up an orchestration process to determine the tasks that this process must perform. You cannot modify an orchestration process in the Order Management work area, but you can monitor an instance of it while it manages each fulfillment line. You can monitor each fulfillment line as it progresses through the orchestration process.</p>
Return fulfillment line	<p>Represents a sales order line that Order Management uses to return items. It is similar to a fulfillment line, but it includes a set of attributes that are specific to returning items.</p>

### Related Topics

- [Transforming Order Lines to Fulfillment Lines: How it Works](#)
- [Creating Sales Orders: Procedure](#)
- [Managing Sales Orders: Overview](#)

- Processing Change Orders: How it Works

## Transformation Rules: Explained

Order Management Cloud allows you to write rules that populate order attributes before it transforms a source order, rules that transform a source order, and rules that populate order attributes after it transforms a source order.

### Pretransformation Rules

A pretransformation rule populates an order attribute before Order Management transforms the source order. You can use the predefined attribute value that the product transformation rules already contain. Order Management defaults the master inventory organization to the sales order so that you can use it in a transformation rule. Consider the following examples.

Example Usage	Description
Populate an attribute on a fulfillment line.	<p>Assume your company receives orders for widgets that include an attribute named Request Date, and that this attribute must display on all fulfillment lines for widgets. You can write a pretransformation rule that implements the following logic:</p> <ul style="list-style-type: none"> <li>• If the item is a widget, then populate the Request Date attribute on the fulfillment line.</li> </ul>
Populate an attribute so you can use it in a transformation rule.	<p>Assume your company receives orders for widgets, and that you plan to write a transformation rule that converts the widget size from centimeters to inches. You must first populate the Size attribute in the fulfillment line. You can write a pretransformation rule that implements the following logic:</p> <ul style="list-style-type: none"> <li>• If the item is a widget, then populate the Size attribute in the fulfillment line.</li> </ul>

### Transformation Rules

Order Management uses item relationships, item structures, transactional attributes, and business rules to transform a sales representation of the item that a source order contains to a fulfillment representation of the item that a sales order in Order Management contains. You can use Oracle Fusion Product Model to define item relationships, item structures, and transactional attributes. You can use the Manage Product Transformation Rules page in the Setup and Maintenance work area to write the transformation rules that this topic describes.

#### Product-to-Product Transformation

A product-to-product transformation transforms a single item to one or more items according to item structure, item relationship, and the transformation rule. Assume your company receives orders for a laptop that comes with multiple accessories, such as a docking station, mouse, and so on. You can write a transformation rule that transforms an item to another item, and that creates individual fulfillment lines. In the following example, a product-to-product transformation rule

transforms a source order for a laptop and accessory package into a sales order that includes individual fulfillment lines for the laptop and for each item that the accessory package contains.

**Source Order**

Line	Quantity	Product	Price
1	1	Laptop	700 USD
1	1	Accessory Package	225 USD

**Transformation**



**Sales Order**

Product	Transformed Order Line	Fulfillment Line	Quantity
Laptop	1	1-1	1
Accessory Package	2	2-1	1
Docking Station	2.1	2.1-1	1
Mouse	2.2	2.1-2	1
AC Adapter	2.3	2.1-3	1
Keyboard	2.4	2.1-4	1

Order Management does the following work to transform the source order:

- Transforms two source order lines into six fulfillment lines.
- Uses the price from the source order to populate the price in the sales order.
- Transforms the line for the laptop in the source order to fulfillment line 1 in the sales order.
- Transforms the Accessory Package in the source order to multiple fulfillment lines in the sales order. Each line represents part of the content of the Accessory Package, such as docking station, mouse, AC adapter, and so on.

**Product-to-Attribute Transformation**

A product-to-attribute transformation transforms the item that a source order references according to the attributes of another item. Assume your company receives orders for a Window item. You can write a transformation rule that uses the Width attribute, Height attribute, Pane attribute, and Glass attribute of the source order to get the item number for an item that uses the same dimensions, and that uses double-pane, tempered glass. The transformed order includes the item number for this combination of transactional attributes.

**Source Order**

Line	Quantity	Product	Transactional Attributes	Price
1	1	Window	Width: 48 inches Height: 96 inches Pane: Double-pane Glass: Tempered	\$550

**Transformation**

Pane	Tempered Glass	Annealed Glass
Double	Product: DP4896T	Product: DP4896A
Triple	Product: TP4896T	Product: TP4896A

**Sales Order**

Line	Quantity	Product	Transactional Attributes
1	1	DP4896T	Width: 121.92 centimeters Height: 243.84 centimeters Pane: Double-pane Glass: Tempered

**Attribute-to-Product Transformation**

An attribute-to-product transformation uses transactional attributes to transform an attribute to an item number. It can add the transformation to an existing item or replace the item that the source order references. Assume your company receives orders for an MP3 player that includes a Color attribute and a Size attribute, and that you must use a combination of these attributes to reference an item number. In the following example, a transformation rule transforms the Color and Size attributes of the Mini Plus item to the VIS481 item. Order Management displays VIS481 on the order line.

**Source Order**

Line	Quantity	Product	Transactional Attributes	Price
1	1	Mini Plus	Color: Silver Size: 8 MB Engraving: For Maria	\$100

**Transformation**

Color	4 MB Size	8 MB Size
Silver	Product: VIS451A	Product: VIS481A
Black	Product: VIS452A	Product: VIS482A
Pink	Product: VIS453A	Product: VIS483A
Blue	Product: VIS454A	Product: VIS484A

**Sales Order**

Line	Quantity	Product
1	1	VIS481A

**Attribute-to-Attribute Transformation**

An attribute-to-attribute transformation transform the value of a transactional attribute on an order line in a source order to a different transactional attribute on an order line in a sales order. Assume your company resides in Europe, it receives orders from an office in the United States that measures the item size in inches, and that you require Order Management to display the item size in centimeters. In the following example, a transformation rule transforms the width and height from inches on the source order to centimeters on the sales order.

**Source Order**

Line	Quantity	Product	Transactional Attributes	Price
1	1	Window	Width: 48 inches Height: 96 inches Pane: Double-pane Glass: Tempered	\$550

**Transformation**

**Sales Order**

Line	Quantity	Product	Transactional Attributes
1	1	DP4896T	Width: 121.92 centimeters Height: 243.84 centimeters Pane: Double-pane Glass: Tempered

**Context-to-Product Transformation**

A context-to-product transformation uses the context of the source order to determine the item that the sales order must reference. Assume your company receives orders for laptop computers that you must ship to different geographical regions. Each region requires a different electrical adapter, such as 110 volts or 240 volts. In the following example, the transformation rule uses the geographical region where you ship the item to determine the adapter to include with the sales order, transforms the source order to a sales order that includes two order lines, and then adds the adapter to one of these lines.

**Source Order**

Line	Quantity	Product	Transactional Attributes	Price
1	1	Vision Pro Laptop	550 Vision Way, Boston, MA 02108	\$1500

**Transformation**

Europe or Asia	North America
Product: 240VAC adaptor	Product: 120VAC adaptor

**Sales Order**

Line	Quantity	Product	Ship To	Price
1	1	Vision Pro Laptop	550 Vision Way, Boston, MA 02108	\$1500
2	1	Product: 120VAC adaptor	550 Vision Way, Boston, MA 02108	0

**Context-to-Attribute Transformation**

A context-to-attribute transformation transforms the source order context to a transactional attribute. Assume your company manufactures laptop computers. Assume it ships some of them to domestic locations that reside in the USA, ships some of them to international locations that reside in other countries, and that the destination location requires different packaging. In the following example, the value of the transactional attribute in the source order includes a domestic address, so the context is domestic, and the transformation rule sets the Domestic Packaging transactional attribute to Yes.

**Source Order**

Line	Quantity	Product	Transactional Attributes	Price
1	1	Vision Pro Laptop	550 Vision Way, Boston, MA 02108	\$1500



Ship International	Ship to USA
Domestic Packaging: No	Domestic Packaging: Yes



**Sales Order**

Line	Quantity	Product	Transactional Attributes	Ship To
1	1	Vision Pro Laptop	Domestic Packaging: Yes	550 Vision Way, Boston, MA 02108

## Posttransformation Rules

A posttransformation rule populates an order attribute after Order Management transforms the item. Consider the following examples.

Example Usage	Description
Populate order lines so they reference different warehouses.	<p>Assume your company receives orders for laptop computers, and that you created a transformation rule that transforms the source order into a sales order that includes the following lines:</p> <ul style="list-style-type: none"><li>• Order line 1 for the laptop computer</li><li>• Order line 2 for an AC adapter</li></ul> <p>You can write a posttransformation rule that populates order line 2 so that it references a warehouse that is different from the warehouse that supplies the laptop computer.</p>
Populate an order with a new attribute.	<p>Assume your company receives orders that use a MM/DD/YYYY format for the requested date. Your staff finds it useful to also know the day of the week because delivery costs more on Saturday and Sunday. You write a posttransformation rule that populates the day of the week in the new sales order.</p>

## Setting Up Transformation: Procedure

You must set up transformation to make sure that Order Management Cloud transforms source orders correctly.

Set up transformation:

1. Use the Oracle Product Model to do the following work:
  - Use the product master to define the product that the transformation rule references.
  - If you plan to use a transformation rule that references the product structure, then define product structures.
  - If you plan to use a transformation rule that references transactional item attributes, then define transactional item attributes.
  - Define the relationship between sales product and fulfillment product.

You must create item substitution rules when you define items. Order Management does not validate item substitutions. For example, assume a fulfillment line includes a standard item, the Order Manager examines availability for this item, and then order promising displays multiple options. If you do not set up each item substitution rule correctly, then the Order Manager might select an option that uses an incorrect substitution, such as a kit.

2. Use the Manage Product Transformation Rules page to create transformation rules. For details, see [Creating Transformation Rules: Procedure](#).

## Transforming Source Orders from Source Systems: How It Works

Order Management Cloud can support one or more source systems. The source orders that your source system captures probably includes information and uses a structure that is different from the information and structure that Order Management uses. If you use multiple source systems, then the information that each system captures on each source order might be different in each system. To manage and fulfill these source orders, Order Management transforms them into a consistent structure.

Order Management uses the following sequence to transform each source order that it receives from a source system:

1. Order Management receives a source order from a source system.
2. The connector service structurally transforms the source order to a sales order that Order Management can use.

3. The connector service calls the Receive and Transform service.
4. The Receive and Transform service (`SalesOrderOrchestrationEBS`), looks up the cross-reference values to determine whether or not it must transform the source order values. It looks up these values in the following sources:
  - o Oracle Product Model
  - o Oracle Trading Community Model, which is a customer data hub
  - o Order Orchestration and Planning Data Repository

Cross-referencing is required for static data, such as country code and currency codes, and for dynamic data, such as customers and items. For details, see [Cross-Referencing: Explained](#).

5. If the order capture system and Order Management use different domain values, then the connector service transforms the structure and values from the domain that the order capture system uses to the domain that Order Management uses.
6. The connector service calls a process that separates the source order into parts.
7. Another service calls the requested operation for the source order, such as create, update, delete, or cancel.
8. Order Management transforms the item according to the following types of transformation rules:
  - o Pretransformation rules
  - o Transformation rules
  - o Posttransformation rules
  - o Process assignment rules

You can create custom transformation rules to specify how to do transformation and to make sure that the source order displays correctly in Order Management. For details, see [Transformation Rules: Explained](#).

9. The Assign and Launch service assigns an orchestration process to each line item according to the rules.

Order Management uses the following attributes to calculate the planned dates for each step and task, starting with the first step that it performs in chronological order:

- Default Lead Time
- Lead Time UOM
- Lead-Time Expression

Order Management can use an order capture service to communicate updates to the order capture system. To receive these updates, the order capture system must subscribe to the events. If Order Management receives an update from the source system, then it replans the orchestration process. It does this replanning every time it receives an update.

Order Management does not perform the same amount of transformation for a sales order that a user creates in the Order Management work area because these orders already include the data and use the structure that Order Management requires.

## Creating Transformation Rules: Procedure

This topic describes how to define a transformation rule that includes a bucket set in a decision table. For details about how to use a simplified rule builder, see [Using Visual Information Builder to Create Rules: Explained](#).

Assume you work for a company that uses priority shipping for each source order that includes the Green Server 3500R. You create a bucket set that contains the values you will select when you create your rule, and then create a transformation rule that represents this offer.

## Summary of the Work

To create transformation rules, do the following work:

1. Create the bucket set list.
2. Create the decision table and add a condition.
3. Add a rule to the decision table.
4. Add the action to take if the condition is true.

This topic includes example values. You might use different values, depending on your business requirements.

## Create the Bucket Set List

Create the bucket set list:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, click **Orders**, search for, and then open Manage Product Transformation Rules.
4. On the Manage Product Transformation Rules page, click **Bucket sets**, click **Add Bucket set**, and then click **List of Values**.
5. In the Bucket sets list, click **Bucket Set 1**, and then click **Edit Bucket set**.  
A bucket set specifies the options that you can choose when you create a transformation rule. It is recommended that you create smaller bucket sets so that you can reuse them.
6. In the Bucket set Editor dialog, set the following values.

Field	Value
Name	Server IDs
Description	List of server item IDs and aliases to select in product transformation rules.
Data Type	String

7. Click **Add Bucket**, and then set the following values in the Bucket 1 row of the Bucket Values list.

Field	Value
Value	300000047394016L This number identifies the item ID that the database contains. The letter L indicates a long value.
Alias	"AS85005" You must include the double quotation marks.
Description	Green Server 3500R

8. Click **Add Bucket**, and then set the following values in the Bucket 2 row of the Bucket Values list.

Field	Value
Value	300000047393961L

Field	Value
Alias	"AS85004" You must include the double quotation marks.
Description	Green Server 3000

9. Click **OK**, and then click **Save**.

## Create the Decision Table and Add a Condition

In this section and the following sections, you will create a decision table, condition, and action that resembles the following configuration:

### Manage Product Transformation Rules

The screenshot shows the 'Manage Product Transformation Rules' interface. On the left, a navigation pane lists 'Globals', 'Bucketsets', 'Links', 'Decision Functions', and 'Rulesets', with 'TransformationCustomRS' selected. The main workspace shows a decision table for the rule 'AddExpeditedShippingforProductAS85005'. The table is titled 'R1' and contains the following data:

Conditions		OrderTransformationRules
		"AS85005"
		?
Conflict Resolution		
Actions		
modify Order		—
ShipClass		"Expedited"

Create the decision table and add a condition:

1. On the Manage Product Transformation Rules page, click **TransformationCustomRS**.
2. In the View list, click **IF/THEN Rules**, click **Add**, and then click **Add Decision Table**.
3. Replace Decision Table 1 with **AddExpeditedShippingforProductAS85005**, and then click **Save**.
4. Click **Collapse Pane**.
5. Immediately above the decision table, click **Add**, and then click **Add Condition**.
6. In the Condition Browser dialog, expand **OrderTransformationRules**, expand **FulfillLineVO**, expand **InventoryItemId**, click **toString**, and then click **OK**.

## Add a Rule to the Decision Table

Add a rule to the decision table:

1. In the decision table, click the **cell** that displays the `OrderTransformationRules.FulfillLineVO.InventoryItemId.toString()` condition.
2. Click **Local List of Values**, and then click **Server IDs**, which is the bucket set that you created.
3. In the `OrderTransformationRules.FulfillLineVO.InventoryItemId.toString()` row, double-click the **cell** in the R1 column, single-click the **cell** in the R1 column, and then add a check mark to **AS85005**.

## Add the Action to Take If the Condition Is True

Add the action to take if the condition is true:

1. Immediately above the decision table, click **Add**, click **Add Action**, and then click **Modify**.
2. In the Action Editor dialog, in the Target list, click **OrderTransformationRules.FulfillLineVO**.
3. In the Arguments list, locate the `ShipClassOfService` argument, set the following value, and then click **OK**. It might be necessary to scroll or page down through the Arguments list.

Property	Description
Parameterized	Contains a check mark.

4. In the decision table, in the `ShipClassOfService` row, double-click the cell in the R1 column, and then enter **"Expedited"**. You must include the double quotation marks.
5. Click **Save**.

### Related Topics

- [Using Visual Information Builder to Create Rules: Explained](#)
- [Using Visual Information Builder to Manage Pretransformation Rules: Procedure](#)

## Creating Advanced Product Transformation Rules: Procedure

This topic describes how to create an advanced transformation rule that can compare two or more lines in a source order. As an example, it uses a rule that removes multiple requests that cancel each other.

This topic implements the following logic.

Statement	Description
First IF statement	If the change in fulfillment line 1 is Add.
Second IF statement	If the change in fulfillment line 2 is Delete.
Third IF statement	If the inventory item in fulfillment line 1 is the same as the inventory item in fulfillment line 2.
Fourth IF statement	If the fulfillment line ID of fulfillment line 1 is different from the fulfillment ID of fulfillment line 2.
THEN statement	Delete fulfillment line 1 and fulfillment line 2.

Your finished work must resemble the following transformation rule.

View IF/THEN Rules + - 1-1 of 1

+ - ↑ ↓ ✂

Consolidate Add and Delete

Description: Rule that removes requests that cancel each other.

Effective Date: Always

Priority: Medium  Active  Advanced Mode  Tree Mode

IF

FulfillLine is a OrderTransformationRules.FulfillLineVO and

FulfillLine.DeltaType is "Add" and

FulfillLine2 is a OrderTransformationRules.FulfillLineVO and

FulfillLine2.DeltaType is "Delete" and

FulfillLine.InventoryItemId is FulfillLine2.InventoryItemId and

FulfillLine.InventoryItemId isn't FulfillLine2.InventoryItemId

THEN

assert new OrderTransformationRules.DeleteOrderLine ( fulfillmentLineId:FulfillLineId, viewRowImpl:FulfillLineId )

retract FulfillLine2

For a demonstration, see [Creating Business Rules: Demo](#).

## Summary of the Work

To create an advanced transformation rule, do the following work:

1. Create the rule.
2. Create the first IF statement.
3. Create the second IF statement.
4. Create the third IF statement.

5. Create the fourth IF statement.
6. Create the THEN statement.

This topic includes example values. You might use different values, depending on your business requirements.

## Create the Rule

Create the rule:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, click **Orders**, search for, and then open Manage Product Transformation Rules.
4. On the Manage Product Transformation Rules page, in the View list, click **IF/THEN Rules**.
5. Click **Expand**, and then click **Show Advanced Settings**.
6. Add a name and description. You can use any value. For this example, use the following values.

Field	Value
Rule 1	Consolidate Add and Delete Actions
Description	Rule that removes requests that cancel each other.

7. Click **Save**.

## Create the First IF Statement

Create the following IF statement:

- If the change in fulfillment line 1 is Add

Create the first IF statement:

1. Add a **check mark** to Advanced Mode.
2. In the field to the left of Is A, enter **FulfillLine**.
3. In the field to the right of Is A, click the **down arrow**, and then click **OrderTransformationRules.FulfillLineVO**.
4. Click **Add Test**, and then click **Simple Test**.
5. Click **Left Value**.
6. In the Condition Browser, expand **FulfillLine**, click **DeltaType**, and then click **OK**.
7. In the Right Value field, enter **"Add"**. You must include the double quotation marks.
8. Click **Save**.

## Create the Second IF Statement

Create the following IF statement:

- If the change in fulfillment line 2 is Delete

Create the second IF statement:

1. Click **Add Pattern**.
2. In the window below And, enter **FulfillLine2**.
3. In the field to the right of Is A, click the **down arrow**, and then click **OrderTransformationRules.FulfillLineVO**.
4. Click **Add Test**, click **Simple Test**, and then click **Left Value**.
5. In the Condition Browser, expand **FulfillLine2**, click **DeltaType**, and then click **OK**.
6. In the Right Value field, enter **"Delete"**. You must include the double quotation marks.

Click **Save**.

## Create the Third IF Statement

Create the following IF statement:

- If the inventory item in fulfillment line 1 is the same as the inventory item in fulfillment line 2

Create the third IF statement:

1. Click **Add Pattern**.
  2. In the window below And, enter **FulfillLine**.
  3. In the field to the right of Is A, click the **down arrow**, and then click **OrderTransformationRules.FulfillLineVO**.
  4. Click **Add Test**, click **Simple Test**, and then click **Left Value**.
  5. In the Condition Browser, expand **FulfillLine**, click **InventoryItemId**, and then click **OK**.
  6. Click **Right Value**.
  7. In the Condition Browser, expand **FulfillLine2**, click **InventoryItemId**, and then click **OK**.
- Click **Save**.

## Create the Fourth IF Statement

Create the following IF statement:

- If the fulfillment line ID of fulfillment line 1 is different from the fulfillment ID of fulfillment line 2

Create the fourth IF statement:

1. Click **Add Pattern**.
  2. In the window below And, enter **FulfillLine**.
  3. In the field to the right of Is A, click the **down arrow**, and then click **OrderTransformationRules.FulfillLineVO**.
  4. Click **Add Test**, click **Simple Test**, and then click **Left Value**.
  5. In the Condition Browser, expand **FulfillLine**, click **InventoryItemId**, and then click **OK**.
  6. Click **Is**, and then click **Isn't**.
  7. Click **Right Value**.
  8. In the Condition Browser, expand **FulfillLine2**, click **InventoryItemId**, and then click **OK**.
- Click **Save**.

## Create the Then Statement

Create the following action:

- Delete fulfillment line 1 and fulfillment line 2

Create the action:

1. In the Then area, click **Add Action**, and then click **Assert New**.
2. Click **Select a Target**, and then click **OrderTransformationRules.DeleteOrderLine**.
3. Click **Edit Properties**.
4. In the Properties dialog, enter the following values, and then click **OK**

Name	Value
fulfillmentLineId	fulfillmentLineId
viewRowImpl	fulfillmentLineId

Name	Value
------	-------

5. Click **Add Action**, and then click **Retract**.
6. Click **Select a Target**, and then click **FulfillLine2**.  
Click **Save**.

*Related Topics*

- [Creating Business Rules: Demo](#)

## Cross-Referencing in Order Management: Explained

You must create and maintain cross-references that relate business data between order capture systems, order fulfillment systems, and Order Management Cloud.

You can use the following types of cross-references.

Type of Cross-Reference	Description
Customer Cross-Reference	<p>The following work occurs:</p> <ol style="list-style-type: none"> <li>1. The source capture system sends a source order to Order Management that includes customer data.</li> <li>2. If the customer already exists in the Oracle customer master, then Order Management uses a customer cross-reference to get the master customer record and the customer ID of the order fulfillment system.</li> <li>3. Order Management sends the sales order, including the customer ID and other attributes that it gets from the Oracle customer master, to the order fulfillment system.</li> </ol> <p>Note the following:</p> <ul style="list-style-type: none"> <li>• The Oracle Trading Community Model maintains the customer cross-references. You can use an external customer hub with Order Management, but you must maintain cross-references in this model.</li> <li>• You can create customer cross-references in the Oracle customer model when you create the customer information when you configure your implementation. This configuration allows Order Management to resolve the Oracle customer values.</li> </ul>
Item Cross-Reference	<p>An item cross-reference creates the following relationships:</p> <ul style="list-style-type: none"> <li>• <b>Source system.</b>A relationship between the source item and the Oracle item when your deployment uses a product hub.</li> <li>• <b>Named item.</b>A relationship between the source item and the item when your deployment brings items from disparate systems into the Oracle Product Model, which is a repository that stores master product information.</li> </ul>
Other Cross-Reference	<p>The Order Orchestration and Product Data Repository stores the cross-references for all attributes that are not customer attributes or item attributes. You can use a domain value map for attributes that reside in the Order Orchestration and Planning Data Repository. The collections process uses these maps to map values from one domain to another domain, which is useful when different domains represent the same data in different ways.</p>



# 5 Controlling Application Behavior

## Managing Order Management Parameters: Procedure

You can use the Manage Order Management Parameters page to manage parameters that affect most or all of Order Management Cloud.

Manage Order Management parameters:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, click **Orders**, search for, and then open Manage Order Management Parameters.
4. Set the following parameters.

System Parameter	Description
Allow Changes Through Configurator Validation	Allows the Configurator to choose items in a configuration and to modify configuration choices after the user adds a configured item to a sales order, and then save the sales order as a draft, but before the user submits the sales order.

For example, assume the following occurs:

1. A user adds a configured item to a sales order, and then saves this sales order as a draft.
2. A configuration manager modifies the configuration model and the configuration rules for this configured item in such a way that it affects the configuration that the user specified.
3. The user returns to the draft several days later, and then submits the sales order.

In this situation, if you set Allow Changes Through Configurator Validation to:

- **Yes.** The configurator uses the modified configuration model and configuration rules to modify the configuration that the user specified. It allows the submit to proceed. It does not report these modifications to the user. This setting is useful when your business process does not require the user to understand and agree to the modifications that the configurator makes.
- **No.** The configurator does not modify the configuration that the user specified. Instead, it does not allow the submit to proceed, and displays a validation error when the user attempts to submit or validate the sales order. This setting is useful when your business process requires that the user understand and agree to the modifications that the configurator makes.

Note the following:

- This parameter does not apply to a source order that includes a configured item. For details about importing source orders, see Order Import Template: Explained.
- This parameter affects all business units.

For details about the configurator, see the guide titled Oracle SCM Cloud, Configurator Modeling.

System Parameter	Description
Configuration Effective Date	<p>Specify the date that Order Management uses to determine the components that it displays for a configured item. The components that it displays might vary depending on the date that you specify. You can choose one of the following values:</p> <ul style="list-style-type: none"> <li>• <b>Ordered Date.</b> Use the components that the Configurator defined for the configured item as of the date that the Ordered Date field displays on the Create Order page.</li> <li>• <b>Configuration Date.</b> Use the components that the Configurator defined for the configured item when it configured the item.</li> <li>• <b>Current Date.</b> Use the components that the Configurator used to define the configured item as of today.</li> <li>• <b>Requested Date.</b> Use the components that the Configurator defined for the configured item as of the requested date.</li> </ul> <p>If you do not specify the Configuration Effective Date, then Order Management uses the Current Date.</p>
Customer Relationship Type	<p>Specify how Order Management displays the Ship-to Address and the Bill-to Location for each customer during order entry. Set the Business Unit field to one of the following values:</p> <ul style="list-style-type: none"> <li>• <b>A single business unit, such as Vision Services.</b> Allow the user to select only the bill-to account or the ship-to address that is associated with the sold-to customer. For example, assume you set the Business Unit to Vision Services and the Customer Relationship Type to Yes. If the user creates a sales order for XYZ Company, then Order Management allows the user to select only the bill-to accounts and the ship-to addresses that XYZ Company references.</li> <li>• <b>All Business Units.</b> Allow the user to select any bill-to account or ship-to address for the business unit, regardless of the sold-to customer. For example, assume you set the Business Unit to Vision Services and the Customer Relationship Type to Yes. If the user creates a sales order for XYZ Company, then Order Management displays the bill-to account that XYZ Company references, by default, and it also allows the user to search and select any bill-to account for this sales order.</li> </ul>
From Address for E-mail Messages	<p>Specify the From e-mail address that the Universal Messaging Service uses when it sends an e-mail from the order document or from an automatic notification. The From address comes predefined as no-reply@orderreport.com. You can change this value to use an address in the domain that your company uses. You must use a valid e-mail address format. For example, something@somethingelse.xxx. For details, see Administering E-mail Formats in Order Management: Explained</p>
Halt Configurator Validation on First Error	<p>Specify one of the following values:</p> <ul style="list-style-type: none"> <li>• <b>True.</b> Stop processing on the first error that the Configurator encounters during order entry.</li> <li>• <b>False.</b> Do not stop processing if the Configurator encounters an error during order entry. Instead, allow the Configurator to continue to run until it finishes processing. This setting allows the Configurator to identify and report all errors that the configuration contains.</li> </ul> <p>Halt Configurator Validation on First Error affects only order entry. It does not affect order import.</p>
Item Validation Organization	<p>Specify the item validation organization that Order Management uses to validate and display item for the sales orders that the user enters for a business unit. Order Management will display only the items that it associates with the item validation organization that you specify.</p> <p>For example, assume you create a record in the Item Validation Organization list, you set the Business Unit for this record to Vision Operations, and you set the Organization to Denver Manufacturing. In this situation, Order Management displays only the items that are associated with</p>

System Parameter	Description
	<p>Denver Manufacturing when the user creates a sales order for Vision Operations. It displays these items when the user searches for items on the Order Lines tab of the Create Order page.</p> <p>Note the following:</p> <ul style="list-style-type: none"> <li>• You must specify an item validation organization for each business unit. If you do not do this, then Order Management disables search for the item in each sales order that references a business unit that you do not specify. As an alternative, if you use the same master organization for every business unit, then you can specify this organization in the Item Validation Organization parameter, and then set the Business Unit for this parameter to All Business Units.</li> <li>• You can set the Item Validation Organization only for business units that Order Management associates with your login responsibility. If it associates your login responsibility with only one business unit, then it uses this business unit as the item validation organization.</li> <li>• The Product Information Management work area associates inventory organizations with business units. If you do not set the Item Validation Organization, and if Product Information Management associates multiple inventory organizations with the same business unit, then Order Management does one of the following: <ul style="list-style-type: none"> <li>◦ If Product Information Management associates the same item master organization with these inventory organizations, then Order Management sets the Item Validation Organization to this item master organization, and then informs the user of this setting. The user can accept this value and continue entering the sales order, or reject it and contact the Order Administrator with a request to use the Item Validation Organization system parameter to set up the inventory organization.</li> <li>◦ If Product Information Management associates different item master organizations with these inventory organizations, then Order Management informs the user that the Order Administrator must first set up the inventory organization before the user can enter the sales order.</li> </ul> </li> <li>• Order Management displays only inventory organizations in the Organization field that the Product Information Management associates with the business unit that you choose in the Business Unit field. You can use the Product Information Management to set up an inventory organization and create these associations. For more information, see Inventory Organizations: Explained.</li> </ul>
Notify Frequency	<p>Specify an integer that represents the number of hours to wait before consolidating, and then sending notifications. It comes predefined with a value of 1. You can set it to any value that is greater than or equal to 0 (zero). If you set it to 0, then Order Management will not consolidate notifications, and will send each notification when the event that it references occurs. For details, see Sending Notifications from Order Management Cloud to External Systems: Explained.</p>
Preparer for Procurement	<p>Specify an Order Management user who the buyer can contact to help resolve a problem that might occur with an order line during fulfillment that involves a drop ship supplier. The buyer is a procurement application user. Order Management sends this information to purchasing.</p> <p>If you do not specify the Preparer for Procurement parameter, then the procurement system will reply with an error that the preparer is missing.</p> <p>You can define a preparer for each selling business unit. You can also define a preparer as the default value for all business units. Order Management will use this default value only for business units where you do not define a preparer.</p> <p>For details about drop ship, see Drop Ship: Overview.</p>
Use Configurator for Order Import Validation	<p>Specify one of the following values:</p> <ul style="list-style-type: none"> <li>• <b>True.</b> The Configurator validates each source order that includes a configured item during order import.</li> </ul>

System Parameter	Description
	<ul style="list-style-type: none"> <li><b>False.</b> The Configurator does not validate each source order that includes a configured item during order import.</li> </ul>

*Related Topics*

- [Product Master Data Management Features: Overview](#)
- [Inventory Organizations: Explained](#)
- [Administering E-mail Formats in Order Management: Explained](#)
- [Sending Notifications from Order Management to External Systems: Explained](#)

## Managing Order Profiles: Procedure

Order Management Cloud uses predefined profile values for each item. Some of these values control behavior in the Order Management work area. Some values control how Order Management receives and transforms source orders into sales orders. Most profiles include predefined values, so you do not need to configure them, unless your organization requires different values.

To set these values, do the following work:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, click **Orders**, search for, and then open Manage Order Profiles.
4. On the Manage Order Profiles page, in the Profile Option area, click **Search**.
5. In the Search Results : Profile Options list, click the **profile** that you must edit.
6. In the Profile Values list, add or delete values for the following fields, as necessary.

Field	Description
Currency Conversion Type	Specifies the value to use when converting a currency in the Order Management work area. This value is a conversion type. You can use this setting for an implementation across a single site and for each user.
Display Currency	Specifies the currency to display in the Order Management work area. You can use this setting for an implementation across a single site and for each user.
Required Overview Status Filter	Specifies the default customer to use when filtering the summary of status data on the Overview page of the Order Management work area. It allows you to view summary data for only one customer at a time. It removes the All option. Order Management provides no value, by default. To improve performance, you can enter a customer identification number. You can use this setting only for an implementation across a single site.
Retain Sales Order Number	Specifies to use the source order number as the order number during transformation. The default value is N. You can use this setting for an implementation across a single site and for each user.
User Request Waiting Period in Seconds	Specifies the number of seconds to wait after an action finishes. This time allows asynchronous services to complete before displaying a confirmation message or a warning message in the Order

---

Field	Description
	Management work area. The default value is 5. You can use this setting only for an implementation across a single site.

---



# 6 Controlling Order Processing

## Controlling Statuses for Sales Orders

### Order Management Statuses: Explained

Order Management Cloud displays a separate status for the sales order, fulfillment line, fulfillment task, and orchestration process. Each status indicates the progress of a sales order from beginning to completion. The statuses of the fulfillment lines, the orchestration processes, and the fulfillment task determines the order status. The Order Management work area displays these status values.

Order Management sequentially evaluates each of the status conditions that a fulfillment line contains while it processes a sales order. The true condition that contains the highest sequence number determines the fulfillment line status. Order Management then uses the status of the fulfillment line that has progressed the furthest in the sales order life cycle to update the order status.

### Where Order Management Displays Status

Order Management displays status in the following locations.

Status	Where Displayed
Order Status	<p>At the top of the Order page in the order title. For example, the order status of the following sales order is Processing:</p> <p>Order: Computer Service and Rentals - 282079 - Processing</p>
Fulfillment Line Status	<p>In the Order Lines area of the Order page. For example, the fulfillment line status of the following order line is Awaiting Shipping:</p> <p>Item, AS54888 - Standard Desktop, Status, Awaiting Shipping</p> <p>You can also click <b>Actions</b>, and then click <b>Switch to Fulfillment View</b> to view the fulfillment line status.</p>
Orchestration Process Status	<p>In the Status field of the Orchestration Process page. For example, Awaiting Fulfillment.</p>
Fulfillment Task Status	<p>In the Status column of the Orchestration Plan tab of the Orchestration Process page. For example, Awaiting Fulfillment.</p>

### Order Status

The status that Order Management displays for the sales order depends on the status of the order line.

Order Status	Description
Draft	You saved the order to the database but have not submitted to order fulfillment. You can modify a Draft order.
Processing	You submitted the order to order fulfillment, but Order Management has not delivered all fulfillment lines to the customer. You cannot modify a Processing order, but you can revise it. For details, see <a href="#">Revising Sales Orders That You Already Submitted: Procedure</a> .
Open	Order Management has not completed any order lines.
Partial	Order Management has completed some, but not all, order lines.
Reference	An earlier version of a revised order. For example, if you revise an existing sales order, and then click Submit, then Order Management displays the earlier version of the sales order with a status of Reference.
Closed	Order Management completed all order lines. You cannot modify a Closed order, but you can create a return order to return items from it. For details, see <a href="#">Returning Sales Orders: Procedure</a> .
Canceled	Order Management canceled all order lines. It also removed the sales order from order fulfillment and did not reschedule it. You cannot submit a Canceled sales order to order fulfillment.  If Order Management canceled only some order lines, then it ignores the canceled order lines, and sets the status according to the order lines that are open.

## Order Line Status

The status that Order Management displays for the order line depends on the following statuses of the fulfillment line.

Order Line Status	Fulfillment Line Status
Open	Order Management has not completed any fulfillment lines.
Partial	Order Management has completed some, but not all, fulfillment lines.
Closed	Order Management completed all fulfillment lines.
Canceled	Order Management canceled all fulfillment lines.  If Order Management canceled only some fulfillment lines, then it ignores the canceled fulfillment lines, and sets the status according to the order lines that are open.

### Related Topics

- [Transforming Order Lines to Fulfillment Lines: How it Works](#)
- [Revising Sales Orders That You Already Submitted: Procedure](#)

## Orchestration Process Status: Explained

An orchestration process status indicates the status of an orchestration process throughout Order Management Cloud. The value Started is an example of this status. It indicates that Order Management has started the orchestration process.

Order Management completes the fulfillment tasks of an orchestration process sequentially according to each orchestration process step when it processes an order.

Order Management comes predefined to use a default set of statuses for fulfillment tasks. You can create custom statuses and sequences. For example, you can create a custom orchestration process that uses a set of statuses and rule logic for textbooks for a college, and another custom orchestration process that uses a different set of statuses and rule logic for textbooks for a primary school.

If you customize an orchestration process status, then note the following:

- You must specify the status that Order Management assigns to an orchestration process at each orchestration process step. For example, if a Schedule Carpet task includes a status of Unsourced, then you must specify the status that Order Management assigns to the orchestration process for this step.
- You must specify a status that indicates when a task is complete. You can select only a status that you define to indicate that a task is complete.
- If you customize the name of the default status, then Order Management displays this custom name throughout the Order Management work area.
- If you do not customize statuses for an orchestration process, then Order Management uses the predefined statuses, by default.

### Status Conditions for Orchestration Processes

You can create a status condition that determines the orchestration process status. For example, you can create the following condition:

- If the status of the Schedule task is Not Started, then set the orchestration process status to Unscheduled.

Order Management evaluates the status conditions that you create sequentially at run time. The condition that evaluates to true, and that includes the highest sequence number, determines the orchestration process status.

### How Order Management Assigns Statuses

Order Management uses the following sequence when it assigns statuses:

1. A fulfillment system sends a status update to Order Management.
2. The External Interface Layer translates the status into status values that Order Management uses.
3. The Task Layer determines the status message to send.
4. The Status Service does the following work:
  - Uses source and target mapping to set the task status.
  - Sets the orchestration process status according to the statuses of the tasks that this process contains.
  - Sets the fulfillment line status according to the task statuses that the fulfillment line references.

#### Related Topics

- [Defining Orchestration Processes: Procedure](#)

## Managing Status Values: Explained

You can use the following tabs on the Manage Status Values page to manage and customize how Order Management Cloud displays statuses.

Tab	Description
Status Codes	Specify the display name that Order Management displays in the Order Management work area.
Fulfillment Lines	Create the status values that users can select for a fulfillment line in an orchestration process.
Task Types	Assign a status code to a task type. If an external system provides the status, then assign this status to the fulfillment task that references this system. For example, if a shipping system provides the status, then assign the code to the Shipping task type.
Orchestration Process Classes	Assign the status code to fulfillment lines or orchestration processes where Order Management must use the status. For example, you can use an orchestration process class to control the status codes that an administrator can choose when creating a status condition for an orchestration process. You must use the Manage Status Values page to define the status values and to make them available when you create a status condition.

## Splitting Fulfillment Lines: Explained

Order Management Cloud can split a fulfillment line into multiple fulfillment lines to improve order fulfillment performance. You can control some of this behavior.

Note the following behavior.

Split	Description
Manual split	<p>Note the following:</p> <ul style="list-style-type: none"> <li>You can use Split Fulfillment Line only for a sales order that is on a manual schedule task. For details, see Splitting Fulfillment Lines: Procedure.</li> <li>To split a fulfillment line that is not on a manual schedule task, you can click <b>Unschedule</b> while the orchestration process is running, and then split the fulfillment line. Unschedule moves the orchestration process back to the schedule task, which now is a manual task. For details, see Fixing Fulfillment Exceptions and Improving Fulfillment Performance: Procedure.</li> <li>Order Management does not run change management for a manual split.</li> <li>Although you cannot use the Check Availability page to manually split a fulfillment line, the availability option that you select might implicitly split the fulfillment line.</li> </ul>
Automatic split	<p>Order Management might automatically split a fulfillment line in the following situations:</p> <ul style="list-style-type: none"> <li>If you select a fulfillment option on the Check Availability page, then this option might require Order Management to split a fulfillment line so that it can provide the availability results that you require.</li> <li>Some other fulfillment system might automatically split a fulfillment line.</li> </ul>

Split	Description
Splits that Order Management does not allow	<p>Order Management does not allow you to split fulfillment line in the following situations:</p> <ul style="list-style-type: none"> <li>• The fulfillment line does not include the Splits Allowed attribute.</li> <li>• The quantity of the fulfillment line is one or less.</li> <li>• The fulfillment line does not allow a split. If you use an override privilege, then you can override this restriction.</li> <li>• A task in the parallel branch of an orchestration process references the fulfillment line.</li> <li>• The fulfillment line resides in a shipment set. To split this line, you can remove it from the shipment set, and then split it.</li> </ul>

## How Order Management Determines Availability When it Splits Fulfillment Lines

If a fulfillment line allows a split or substitution, then Order Management examines possible substitutions when it determines availability.

Where Split Occurs	Description
Split a fulfillment line across multiple warehouses	<p>If the Requested Ship-from Warehouse attribute on the fulfillment line is:</p> <ul style="list-style-type: none"> <li>• <b>Empty.</b> Order Management considers multiple warehouses when it splits a fulfillment line.</li> <li>• <b>Not empty.</b> Order Management considers only the warehouse that the Requested Ship-from Warehouse references. It does not split the fulfillment line.</li> </ul> <p>Each fulfillment line that the split creates might specify a different value for the Expected Ship-from Warehouse attribute.</p> <p>Order Management uses business rules that your order administrator sets up to determine the warehouses it uses to supply the item.</p>
Split a fulfillment line across substitute items	<p>Order Management substitutes an item only if the Allow Substitutions attribute on the fulfillment line equals Yes. Each fulfillment line that the split creates might specify a different value for the Available Item attribute.</p> <p>For example, assume you create a fulfillment line that requests 100 units of an item. Assume supply for this item is 80 units and supply for the substitute item is 50 units. Check Availability will split the fulfillment line into the following fulfillment lines:</p> <ul style="list-style-type: none"> <li>• One fulfillment line that references 80 units of the item</li> <li>• Another fulfillment line that references 20 units of the substitute item</li> </ul>
Split a fulfillment line across dates	<p>If a fulfillment line requests inventory that is not sufficient on a given date, then Order Management splits this line across the following dates:</p> <ul style="list-style-type: none"> <li>• Creates one fulfillment line for the quantity that is available on the requested date.</li> <li>• Creates another fulfillment line for the remaining quantity that it will deliver on a later date.</li> </ul> <p>For example, assume you create a fulfillment line on November 15, 2016 that requests 100 units of an item, and that supply is available for this item on the following dates:</p> <ul style="list-style-type: none"> <li>• 70 units are available on November 15, 2016.</li> <li>• 40 units are available on November 30, 2016.</li> </ul>

Where Split Occurs	Description
	<p>Check Availability will split the fulfillment line into the following fulfillment lines:</p> <ul style="list-style-type: none"> <li>• One fulfillment line with on-time delivery of 70 units, with an expected delivery date of November 15, 2016.</li> <li>• Another fulfillment line with delayed delivery for 30 units, with an expected delivery date of November 30, 2016.</li> </ul>

## How Split Fulfillment Lines Affect Status

Order Management Cloud creates two or more instances of the same fulfillment task when it splits a fulfillment line. These tasks might include different statuses during processing.

Consider the following examples:

- Assume the status of the Schedule task for fulfillment line A1 is Not Scheduled, and the status of the Schedule task for fulfillment line A2 is Scheduled. In this example, Order Management examines the split priority of the task statuses, and then sets the orchestration process status to the task status that includes the highest split priority.
- Assume Order Management splits a fulfillment line that results in two instances of the Schedule task. One of these tasks includes a Complete status, and the other task includes a Pending status. Assume the value of the split priority for Pending is two, and the value of the split priority for Complete is three. In this example, Order Management sets the orchestration process status to Pending.

### Related Topics

- [Splitting Fulfillment Lines: Procedure](#)
- [Fixing Problems and Improving Fulfillment Performance: Procedure](#)
- [Transforming Order Lines to Fulfillment Lines: How it Works](#)
- [Shipping Order Lines in Shipment Sets: Procedure](#)

## Managing Task Status Conditions: Procedure

Order Management Cloud fulfills each fulfillment task one step at a time when it processes a sales order, and it uses a predefined set of sequential statuses to track the progress of each task. You can use the Manage Task Status Conditions page to manage the status conditions that determine the status of these tasks according to each task type.

Manage task status conditions:

1. On the Manage Task Status Conditions page, in the task list, click the task type that you must manage.

For example, to manage the status conditions when Order Management processes a return, click the row that contains Return in the Type column.

2. In the Status Conditions list, modify or add new conditions as necessary. Note the following:

Field	Description
Internal Status Value	Displays a status that Order Management receives from a fulfillment system through the task layer. For details, see Task Layer: Explained

Field	Description
Display Status Value	Specifies the value that Order Management displays. For example, assume the fulfillment system sends a status of Invoiced for a task, but your company uses Billed. You can set Display Status Value to Billed to display Billed throughout the Order Management work area.
Mark as Complete	If this option contains a check mark, then Order Management considers the task to be complete when it reaches the condition. For example, assume you click the Return row in the task list on the Manage Task Status Conditions page, and then, in the Status Conditions area, you add a check mark to Mark as Complete for the Canceled status condition and for the Delivered status condition. In this situation, Order Management considers the task to be complete when the task reaches the Canceled status or the Delivered status.

## Customizing Fulfillment Line Statuses: Explained

Order Management Cloud fulfills each fulfillment task when it processes each orchestration process step that is associated with a fulfillment line. It uses these statuses to represent the fulfillment line status throughout Order Management. You can specify the status that it assigns to this fulfillment line at each step when you define an orchestration process. For example, you can specify that if a Schedule Carpet task includes a status of Pending Scheduling, then set the fulfillment line status to Unscheduled.

You can specify different sets of statuses and rules for different items that a fulfillment line references. For example, you can specify one set of status conditions for a textbook, and another set of status conditions for a paperback book. If you do not create these conditions, then Order Management uses the status rule set that it assigns to the default category.

### Using Status Catalogs and Status Categories with Fulfillment Lines

Your organization might require each fulfillment line that an orchestration process references to use a different status. For example, an item that includes shippable fulfillment lines might require statuses that are different from the statuses that an item that includes nonshippable fulfillment lines uses. A status catalog allows you to group items that are similar so that they can achieve the same statuses at the same time.

You can select a status catalog when you define an orchestration process only if this catalog meets the following requirements:

- An item exists in only one category in the catalog.
- The category contains items or subcategories, but not items and subcategories.
- Order Management controls the catalog only for the master, and not for each organization.

You can use catalogs and categories in multiple orchestration process definitions. Use a category to make sure that Order Management applies the same set of status conditions to specific sets of fulfillment lines. It applies the same status conditions to all fulfillment lines that reference the item that resides in this category.

You can use the Oracle Product Model to define a custom status catalog.

### Using Status Rule Sets with Fulfillment Lines

A status rule set allows you to use a single set of rules with multiple fulfillment lines instead of specifying separate rules for each fulfillment line. For example, you can specify the following rules:

- If an item includes a status of Unsourced, then set the fulfillment line status to Unscheduled.

- If the Schedule Carpet fulfillment task reaches a status of Completed, then set the fulfillment line status to Scheduled.

You can use a single status rule set with multiple categories. If a parent category and a child category each reference a different status rule set, then Order Management uses the status rule set that the child references. This configuration allows you to define an All category to handle all items in one orchestration process. It also allows you to add a subcategory for a subset of items that must use a different status rule set.

**⚠ Caution:** If you use your implementation project to migrate an orchestration process instance from a development environment to a production environment, then do not modify the status rule set name in either environment. Modifying this name might prevent Order Management from updating references to other data in the orchestration process.

## Customizing Fulfillment Line Statuses: Procedure

You can use the Manage Status Values page to create custom status values that Order Management Cloud displays for a fulfillment line.

### Summary of the Work

To create custom status values that Order Management displays for a fulfillment line, do the following work:

1. Add the status codes.
2. Add status codes to fulfillment lines.

This topic includes example values. You might use different values, depending on your business requirements.

### Add the Status Codes

Add the status codes:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, click **Orders**, search for, and then open Manage Status Values.
4. On the Manage Status Values page, in the Status Codes list, click **Actions**, and then click **Create**.
5. In the Create Status Code dialog, enter the following values, and then click **Save and Create Another**.

Field	Value
Code	SCHED_GOODS
Name	Scheduled Goods

6. In the Create Status Code dialog, enter the following values, and then click **Save and Close**.

Field	Value
Code	SCHED_CLOTHING
Name	Scheduled Clothing

## Add Status Codes to Fulfillment Lines

Add the status codes that you created to the fulfillment lines:

1. On the Manage Status Values page, click **Fulfillment Lines**, click **Actions**, and then click **Select and Add**.
2. In the Select and Add dialog, in the Status Code field, enter `SCHED_GOODS`, and then click **Search**.
3. Click **SCHED\_GOODS**, and then click **OK**.
4. Click **Actions**, and then click **Select and Add**.
5. In the Select and Add dialog, in the Status Code field, enter `SCHED_CLOTHING`, and then click **Search**.
6. Click **SCHED\_CLOTHING**, and then click **OK**.
7. Click **Save and Close**.

You can now set the status value for a fulfillment line in an orchestration process to `SCHED_GOODS` or `SCHED_CLOTHING`.

## Creating Orchestration Process Classes: Procedure

You can use the orchestration process class as a way to group orchestration process statuses so that they are meaningful in an orchestration process. For example, the Ship Order Class includes statuses that are meaningful when shipping an order, such as Scheduled, Awaiting Shipping, and Shipped.

This topic includes example values. You might use different values, depending on your business requirements.

Create an orchestration process class:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, click **Orders**, search for, and then open Manage Status Values.
4. On the Manage Status Values page, click **Orchestration Process Classes**.
5. In the Orchestration Process Classes list, click **Actions**, click **Create**, and then set the following values.

Field	Value
Code	Standard_Class
Name	Standard_Class
Description	Class for standard orders

6. Click **Save**.
7. In the Status Values area, click **Actions**, click **Add Row**, and then add the following value.

Field	Value
Status Value	Scheduled

8. Repeat step 6 for each of the following values:
  - o Shipped
  - o Reserved
  - o Billed

- o Returned to Customer

**9. Click Save and Close.**

You can now set the Standard\_Class class in the Process Class field on the Edit Orchestration Process Definitions page. The class that you set determines the values that you can select when you set the Status Value of the Status Conditions tab on this page. For example, if you set the Process Class to Standard\_Class, then the Status Value field allows you to select the statuses that the Standard\_Class class references, such as Shipped.

*Related Topics*

- [Defining Orchestration Processes: Procedure](#)

## Adding Status Conditions to Orchestration Processes: Procedure

You can add status conditions that specify when to set the status of an orchestration process. Assume your company requires an orchestration process that can fulfill orders for company t-shirts, and that you must specify the statuses that this process uses throughout the order life cycle according to the status of the fulfillment task. You can create the condition that determines the orchestration process status.

In this topic, you create the following example status condition:

- If the status of the Schedule task is Scheduled, then set the orchestration process status to Scheduled.

### Summary of the Work

To add a status condition to an orchestration process, do the following work:

- Set the orchestration process class.
- Add the status condition.

You typically add more than one status condition to an orchestration process. For brevity, in this topic you add only one status condition.

This topic includes example values. You might use different values, depending on your business requirements.

### Set the Orchestration Process Class

Set the orchestration process class:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, click **Orders**, search for, and then open Manage Orchestration Process Definitions.
4. On the Manage Orchestration Process Definitions page, locate the CallCustomerWhenLargeInvoice orchestration process, click **Actions**, and then click **Edit**.  
For details about how to create this process, see Branching Orchestration Processes: Procedure.
5. On the Edit Orchestration Process Definitions page, set the following value.

Field	Value
Process Class	Standard_Class
	Standard_Class is a custom class. You must define it before you can select it. For details, see Creating Orchestration Process Classes: Procedure.

6. Click **Save**.

## Add the Status Condition

Add the status condition:

1. In the Process Details area, click **Status Conditions**.
2. In the Orchestration Process Status Values list, click **Actions**, and then click **Add Row**.
3. In the new row, set the following values.

Field	Value
Sequence	1
Status Value	Scheduled

4. Add the expression:
  - o In the new row, in the Expression column, click the icon.
  - o In the Expression Builder dialog, click **CallCustomerWhenLargeInvoice - Schedule**, and then click **Insert Into Expression**. Make sure you do not expand CallCustomerWhenLargeInvoice - Schedule.  
Notice that the dialog added a value of "**Schedule**" in the Expression window of the Expression Builder.
  - o In the Expression window, click anywhere after "**Schedule**", and then enter an equal sign (=).
  - o Expand **CallCustomerWhenLargeInvoice - Schedule**, click **SCHEDULED**, and then click **Insert Into Expression**.
  - o Notice that the Expression window contains "**Schedule**"=**SCHEDULED**", and then click **OK**.
5. Click **Save**.

### Related Topics

- [Adding Branches to Orchestration Processes: Procedure](#)

## Adding Status Conditions to Fulfillment Lines: Procedure

You can add status conditions to a fulfillment line that includes multiple items, and where each of these items requires a different status. If different fulfillment lines must use different statuses, then you must determine how you will use catalogs and categories to group these fulfillment lines. You do this work when you create an orchestration process.

In this topic, assume you must set up orchestration processes that can handle orders for different types of t-shirts. You could use the same orchestration process for multiple types of merchandise, but you prefer to define statuses for each type of clothing separately because each clothing type requires a different status. To do this, you select the status catalog, and then add the status conditions for a single category of items in an orchestration process.

### Summary of the Work

To add status conditions to a fulfillment line, do the following work:

- Set the status catalog.

- Add the status condition for each default.
- Add the status condition for an item.

You typically create multiple status conditions for a fulfillment line. For brevity, in this topic you create only one status condition for the default category and another status condition for an item.

This topic includes example values. You might use different values, depending on your business requirements.

## Set the Status Catalog

Set the status catalog:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, click **Orders**, search for, and then open Manage Orchestration Process Definitions.
4. On the Manage Orchestration Process Definitions page, locate the CallCustomerWhenLargeInvoice orchestration process, click **Actions**, and then click **Edit**.

For this example, you edit the CallCustomerWhenLargeInvoice orchestration process. For details about how to create this process, see Branching Orchestration Processes: Procedure.

5. On the Edit Orchestration Process Definitions page, set the following value.

Field	Value
Status Catalog	Retail_Merchandising_Catalog
	Retail_Merchandising_Catalog is an example catalog. You must use the Oracle Product Model to define it and the values that it references before you can specify it for this orchestration process.

If a warning dialog displays, then click **OK**.

6. Click **Save**.

## Add the Default Status Condition

Add the status condition that this orchestration process will use, by default:

1. In the Process Details area, click **Status Conditions**, and then click **Fulfillment Line Status Values**.
2. In the Default row, in the Status Rule Set field, click the **arrow**, and then click **Create**.  
You will select the statuses that this orchestration process uses, by default. Every row in the Fulfillment Line Status Values list must reference a status rule set, and each rule set must contain at least one status condition.
3. In the Create Status Rule Set dialog, set the following values, and then click **Save and Close**.

Field	Value
Code	LargeOrders
Name	Large Orders
Create New	Chosen

4. Click **Save**.
5. In the Default row, click **Edit Status Rule Set**.

6. On the Edit Status Rule Set page, click **Actions**, and then click **Add Row**.
7. In the Sequence field, enter 1.
8. Set the status value:
  - o In the Status Value field, click the **arrow**, and then click **Search**.
  - o In the Search and Select dialog, in the Status Code field, enter **SCHED**, and then click **Search**.
  - o In the list, click **Scheduled**, and then click **OK**.
9. Add the expression:
  - o On the Edit Status Rule Set page, in the Expression column, click the icon.
  - o In the Expression Builder dialog, click **CallCustomerWhenLargeInvoice - Schedule**, and then click **Insert Into Expression**. Make sure you do not expand CallCustomerWhenLargeInvoice - Schedule.  
Notice that the dialog added a value of "Schedule" in the Expression window of the Expression Builder.
  - o In the Expression window, click anywhere after "Schedule", and then enter an equal sign (=).
  - o Expand **CallCustomerWhenLargeInvoice - Schedule**, click **SCHEDULED**, and then click **Insert Into Expression**.
  - o Notice that the Expression window contains "Schedule"="SCHEDULED", and then click **OK**.
10. On the Edit Status Rule Set page, add a check mark to Notify External Systems, and then click **Save and Close**.  
Notify External Systems allows Order Management to communicate the status to an external system. For details, see [Sending Notifications from Order Management Cloud to External Systems: Procedure](#).

## Add the Status Condition for an Item

Add the status condition for an item:

1. In the Process Details area, in the Fulfillment Line Status Values list, click **Actions**, and then click **Select and Add: Category**.
2. In the Select and Add: Category dialog, select **Retailer**, and then click **Save and Close**.
3. In the Fulfillment Line Status Values list, in the Retailer row, click the **arrow** in the Status Rule Set column, and then click **Create**.
4. In the Create Status Rule Set dialog, set the following values, and then click **Save and Close**.

Field	Value
Code	Shirts
Name	Shirts
Create New	Chosen

5. Click **Save**.
6. In the Fulfillment Line Status Values list, in the Retailer row, click **Edit Status Rule Set**.
7. Repeat steps 6 through 10 of the Add the Default Status Condition topic, above.

### Related Topics

- [Adding Branches to Orchestration Processes: Procedure](#)

## Closing Fulfillment Lines That Remain Open: Explained

Order Management Cloud closes a fulfillment line when it finishes all steps of the orchestration process. In some situations, an orchestration process step might indicate that fulfillment is complete, such as shipping complete for a shipping step. This indicator might mean that your organization considers fulfillment complete when the orchestration process reaches this step, but the fulfillment line is still open.

If the scheduled process named Update or Close Sales Orders did not run since Order Management closed the fulfillment line, then Order Management might display sales orders and order lines might as open even if it closed all the fulfillment lines that these orders and lines reference. To fix this situation, you can run this scheduled process in the Scheduled Processes work area.

## Constraining Orders

### Managing Processing Constraints: Explained

If a user attempts to submit or change a sales order, order line, or order fulfillment line, and if a processing constraint does not allow this submit or change, then Order Management Cloud rejects it and displays a message. If an order capture system attempts to submit this order or make this change, then Order Management rejects it and sends a return message. Order Management also uses the processing constraint to make sure each fulfillment request includes the attributes that it requires.

The processing constraint uses the following logic:

- If the validation rule set that Order Management applies to a record set is true, then apply the processing constraint to the constraint entity.

The processing constraint includes the following parts.

Part	Description
Role	The user role that the constraint does not allow to make a change. For example, you can constrain an order manager from changing a sales order when the orchestration process proceeds beyond a step that you specify.
Action	The action that the constraint does not allow. You can constrain the user from performing the following actions: <ul style="list-style-type: none"> <li>• Create</li> <li>• Validate</li> <li>• Update</li> <li>• Split</li> <li>• Submit</li> <li>• Cancel</li> <li>• Delete</li> </ul>
Condition	The condition that the constraint evaluates to determine whether or not to apply the constraint. For example, you can create a condition that applies a constraint if an order is booked.

You can use the Manage Processing Constraints page to manage predefined constraint objects, such as predefined record sets, validation rule sets, and constraint entities, or to create custom ones. If you create a processing constraint that does not use a condition, then the constraint is always true. For example, the predefined processing constraint that prevents a user from deleting a sales order prevents deletion in all situations.

## Objects That You Specify for Processing Constraints

In general, you specify the following objects when you define a processing constraint.

Object	Description
Validation rule set	<p>A group of one or more If statements. For example, you can use the following rule to reject a change on a closed order:</p> <ul style="list-style-type: none"> <li>If the order is closed, then reject the change.</li> </ul> <p>In this example, a predefined validation rule set named Order is Closed examines the Open attribute on the order header to determine whether or not the value is N.</p> <p>Note the following:</p> <ul style="list-style-type: none"> <li>You cannot modify or delete a predefined validation rule set, but you can create a custom one.</li> <li>You can apply a processing constraint when the condition is true or is not true.</li> </ul>
Record set	<p>A set of records that Order Management groups according to common attribute values so that it can evaluate a constraint. For example, to evaluate all orders for a customer, you can specify to evaluate one of the following entities when you create a record set:</p> <ul style="list-style-type: none"> <li>Order header</li> <li>order line</li> <li>Order fulfillment line</li> </ul> <p>You can then select an attribute to refine the record set. For example, to evaluate all orders for a customer, select the Order Header entity, and then select the Sold-to Customer attribute.</p>
Constraint entity	<p>The business object or orchestration process that a processing constraint constrains. For example, an order header, or an attribute of an order header, such as the Latest Acceptable Ship Date field that the order header displays. For details, see <a href="#">Managing Constraint Entities: Explained</a>.</p>
Constraint package	<p>A set of triggers that Order Management applies to a table in an Oracle database. A background process sets these triggers when you create a constraint package.</p> <p>A constraint package can activate a new or modified:</p> <ul style="list-style-type: none"> <li>Validation rule set that is of a table type</li> <li>Record set for a processing constraint</li> </ul> <p>To create a constraint package, you can do the following:</p> <ul style="list-style-type: none"> <li>Use the Manage Processing Constraints page or run the Generate Constraint Packages scheduled process.</li> <li>Create or modify a record set or a validation rule set that is of a table type.</li> </ul>

Object	Description
	You do not need to create a constraint package for any validation that is not of a table type.

For example, the following processing constraint prevents a change to any order fulfillment line for a specific customer that is part of an orchestration process that is in the shipping stage:

Object	Example
Validation rule set	If the Status Code attribute of a sales order is equal to Awaiting Shipment. . .
Record set	. . .for the customer that the Bill-to customer ID attribute of the order fulfillment line identifies. . .
Processing constraint	. . .then do not allow an update. . .
Constraint entity	. . .on any of these order fulfillment lines.

## Managing Processing Constraints: Examples

This topic includes examples of some of the ways that you can use a processing constraint.

### Reject Changes at the Shipping Stage

Assume an orchestration process achieves the shipping stage for a sales order, and a user submits a change for this sales order. The shipping stage occurs late in the orchestration process. It is expensive and not practical to change this sales order. You can create a processing constraint that rejects the change if the orchestration process has achieved the shipment step.

### Reject Orders That Do Not Include Required Attributes

Assume your company does not deliver items to an address that does not include a ship-to contact. You can create a processing constraint that rejects a sale order if the user does not include a ship-to contact.

### Reject Changes That Require Approval

Assume your company does not allow a customer service representative to submit a change if the transaction value exceeds \$100, and if a manager has not approved the change. You can create a processing constraint that rejects a change if the user uses the customer service representative role, and if the transaction exceeds \$100, and if a manager has not approved the change.

## Managing Processing Constraints: Procedure

This topic describes how to define a processing constraint.

### Summary of the Work

To define a processing constraint, do the following work:

1. Create a record set.

2. Create a validation rule set.
3. Create a processing constraint.
4. Test your work.

In this example, you define a processing constraint that prevents a change to any order fulfillment line that is part of an orchestration process that is in the shipping stage. This topic includes example values. You might use different values, depending on your business requirements.

## Create a Record Set

Create a record set:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, click **Orders**, search for, and then open Manage Processing Constraints.
4. On the Manage Processing Constraints page, click **Record Sets**, click **Actions**, click **Add Row**, and then enter the following values.

Field	Value
Name	Fulfillment Lines That Belong to Same Customer
Description	A record set created on fulfillment lines that belong to the same customer.
Short Name	FCST
Entity	Order Fulfillment Line

5. In the Fulfillment Lines That Belong to Same Customer area, click **Actions**, click **Add Row**, and then set the following value.

Field	Value
Attribute Name	Bill-to customer ID

6. Click **Save**, and then click **Generate Packages**.
7. In the Confirmation dialog, click **OK**.

## Create a Validation Rule Set

Create a validation rule set:

1. On the Manage Processing Constraints page, click **Validation Rule Sets**.
2. Click **Actions**, click **Add Row**, and then enter the following values.

Field	Value
Name	Shipment VRS
Description	The validation rule set for lines with status Awaiting Shipment.
Short Name	SHIP

Field	Value
Validation Type	Table
Entity	Order Fulfillment Line

3. In the Shipment VRS area, click **Actions**, click **Add Row**, and then set the following values.

Field	Value
Attribute Name	Status Code
Validation Operation	Equal to
Value String	"Awaiting shipment" You must include the double quotation marks.

4. Click **Save**, and then click **Generate Packages**.
5. In the Confirmation dialog, click **OK**.

## Create a Processing Constraint

Create a processing constraint:

1. On the Manage Processing Constraints page, click **Constraints**.
2. Click **Actions**, click **Add Row**, and then enter the following values.

Field	Value
Constraint Name	Shipping Constraint
Display Name	Shipping Constraint
Constraint Entity	Order Fulfillment Line
Constrained Operation	Update
Enabled	Does not contain a check mark. If Enabled contains a check mark, then your constraint might affect other activities.

3. In the Shipping Constraint area, in the Conditions list, click **Actions**, click **Add Row**, and then set the following values.

Field	Value
Group Number	1
Validation Entity	Order Fulfillment Line

Field	Value
Validation Rule Set	Shipment VRS
Record Set	Fulfillment Lines That Belong to Same Customer
Message	You cannot update the fulfillment line because it is in the shipping stage.

- In the Shipping Constraint area, click **Applicable Roles**, make sure All Roles is chosen, and then click **Save**.

## Test Your Work

Test your work:

- In the Order Management work area, access a fulfillment line that includes a Status of Awaiting Shipment.
- Attempt to change any attribute.
- Verify that Order Management does not allow you to make the change, and displays the message that you specified in the Message field when you defined the constraint.

## Managing Constraint Entities: Explained

You can use a constraint entity to identify the business object or orchestration process that a processing constraint constrains.

You can specify the following types of constraint entities on the Manage Constraint Entities page.

Type of Constraint Entity	Description
View	<p>Constrains changes according to the value of an attribute that a user can view in the Order Management work area. The order header is an example of a view constraint entity. For example, if you specify Order Header as the constraint entity of a processing constraint, then the constraint constrains changes according to the order header. You can choose one of the following types of view constraint entities:</p> <ul style="list-style-type: none"> <li>Order Fulfillment Line</li> <li>Order Line</li> <li>Order Header</li> </ul> <p>You can use a predefined view constraint entity, but you cannot create a new one.</p> <p>To specify the attributes to constrain, you can use the Attribute Details list that the Manage Constraint Entities page displays when you choose a view constraint entity. For details, see <a href="#">Constraining Changes to Attributes: Procedure</a>.</p>
Process	<p>Constrains an action from occurring at some point in an orchestration process, such as updating an attribute or deleting a table entity, according to a combination of orchestration process, task, and service.</p>

Type of Constraint Entity	Description
	<p>For example, the task layer service named Update Shipping references a process constraint entity named OrderOnlyProcess that uses the following configuration:</p> <ul style="list-style-type: none"> <li>• Orchestration Process is OrderOnlyProcess</li> <li>• Task is Shipping</li> <li>• Service is UpdateShipment</li> </ul> <p>This process constraint entity constrains a change only if the OrderOnlyProcess orchestration process is running, and if this process is currently on the Shipping task, and if this process calls the UpdateShipment service.</p> <p>Note the following:</p> <ul style="list-style-type: none"> <li>• A process constraint entity considers the current position of the transaction in the orchestration process flow.</li> <li>• A process constraint entity can validate required attributes for a fulfillment request, such as a Create Shipment request, Update Shipment request, or Create Reservation request.</li> <li>• You can use a predefined process constraint entity or create a custom one.</li> </ul>

## Constraining Changes to Attributes: Procedure

Order Management Cloud comes predefined to allow processing constraints on some attributes, but not for others. For example, if you must constrain changes to the Latest Acceptable Ship Date attribute of an order header, then you must enable processing constraints for this attribute.

Constrain changes on an attribute:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, click **Orders**, search for, and then open Manage Constraint Entities.
4. On the Manage Constraint Entities page, in the Entity Type field, click **View Entity**, and then click **Search**.

The Manage Constraint Entities page displays view constraint entities and process task constraint entities. Order Management can constrain changes to any process task constraint entity, so it is not necessary to enable them for constraint.

5. In the Search Results, click the constraint entity that displays the attribute that you must constrain.

For example, to constrain changes to the Latest Acceptable Ship Date field that the order header displays, in the Display Name column, click **Order Header**.

6. In the Attribute Details list, add a check mark to the **Constraint Enabled** option next to the attribute that you must constrain, such as Latest Acceptable Ship Date.

## Constraining Changes That Users Make in Extensible Flexfields: Procedure

You can constrain the changes that a user can make in an extensible flexfield. For example, you can use a processing constraint to not allow the user to update an extensible flexfield if a fulfillment line is closed, or to require the user to enter a value in an extensible flexfield at a step of an orchestration process.

Note the following points:

- You use the Manage Constraint Entities page to enable an extensible flexfield so that you can use it on the Manage Processing Constraints page to create a processing constraint. These pages display the extensible flexfields that are currently configured.
- Extensible flexfields are not available on the Attributes menu of the Record Sets tab.
- You must enable an extensible flexfield before you can use it.

The example in this topic describes how to create a processing constraint that implements the following logic:

- If a fulfillment line is closed, then do not allow the user to change the value of the extensible flexfield named Subcontractor ID

You can use an extensible flexfield to meet other business requirements. For details, see *Configuring Extensible Flexfields in Order Management: Explained*.

## Summary of the Work

To constrain the changes that a user can make in an extensible flexfield, do the following work:

1. Enable the extensible flexfield.
2. Create a validation rule set.
3. Create a processing constraint.
4. Test your work.

This topic uses example values, such as Subcontractor ID. You might use different values, depending on your business requirements.

## Enable the Extensible Flexfield

Enable an extensible flexfield so that a processing constraint can reference it:

1. If necessary, publish and deploy the extensible flexfield that you require.  
For details, see *Publishing and Deploying Extensible Flexfields in Order Management: Procedure*.
2. In the Navigator, click **Setup and Maintenance**.
3. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
4. On the Setup page, click **Orders**, search for, and then open **Manage Constraint Entities**.
5. On the Manage Constraint Entities page, set Entity Type to **Equals**, View Entity, and then click **Search**.
6. In the Search Results area, select **Order Fulfillment Line**.
7. In the Attribute Details area, locate the extensible flexfield that you must enable, make sure **Constraint-Enabled** contains a check mark, click **Save and Close**, and then click **Done**.

For this example, enable Subcontractor:ID.

## Create a Validation Rule Set

Create a validation rule set:

1. On the Setup page, search for, and then open **Manage Processing Constraints**.
2. On the Manage Processing Constraints page, click **Validation Rule Set**.
3. Click **Actions**, click **Add Row**, set the following values, and then click **Save**.

Field	Value
Name	Fulfillment Line Is Closed

Field	Value
Description	Fulfillment line is closed.
Short Name	FLCLOSE
Validation Type	Table
Entity	Order Fulfillment Line

4. In the Details area, click **Actions**, click **Add Row**, set the following values, and then click **Save**.

Field	Value
Attribute Name	Open
Validation Operation	Equal To
Value String	N

5. Click **Generate Packages**. Order management makes the validation rule set active and allows you to use it in a processing constraint.

If you add multiple lines in the Details area of a validation rule set, then order management evaluates them together.

You cannot select different contexts of an extensible flexfield in a single validation rule set. If you select two context and segment attributes, then the attributes must use the same context value; that is, the context is mutually exclusive. For example, in the following context and segment combinations, you cannot simultaneously select the ID segment for the Dealer Information context and for the Warranty Information context.

Context	Segment
Dealer Information	ID
Dealer Information	Location
Warranty Information	ID

## Create a Processing Constraint

Create a processing constraint:

1. Click **Constraints**, click **Actions**, click **Add Row**, set the following values, and then click **Save**.

Field	Value
Name	Constrain Subcontractor ID

Field	Value
Display Name	Constrain Subcontractor ID
Constraint Entity	Order Fulfillment Line
Constrained Operation	Update
Attribute Name	Subcontractor: ID

2. In the Details area, click **Actions**, click **Add Row**, set the following values, and then click **Save**.

Field	Value
Group Number	100
Validation Entity	Order Fulfillment Line
Validation Rule Set	Fulfillment Line Is Closed
Scope	Any
Record Set	Fulfillment Line Default Record Set
Message	The fulfillment line is closed. You cannot modify it.

## Test Your Work

Test your work:

1. Attempt to modify the Subcontractor ID field of a fulfillment line that is closed.
2. Verify that order management does not allow this modification.

## Managing Change Orders

### Processing Changes That Occur During Order Fulfillment: How It Works

A change that affects a sales order during order fulfillment can originate from a variety of sources, such as from a user through an order capture system, from a user who is using the Order Management work area, or from change orders that you import. Order Management Cloud comes predefined to process these changes.

Order Management cross-references, transforms, validates, and orchestrates the change in a way that is similar to how it does this work for a new sales order. If rules that control how to process a change exist, then Order Management applies them. You can use the setup options to modify how Order Management does this processing.

Consider the following example. Assume a manager at Vision Corporation places a sales order for 12 employee appreciation gifts that include a company logo, and that come in a variety of colors. This sales order includes the following order lines, and each of these order lines references a fulfillment line:

Sales Order Line	Fulfillment Line
Order Line 1, 4 red	Fulfillment Line 1, 4 red
Order Line 2, 4 blue	Fulfillment Line 2, 4 blue
Order Line 3, 4 black	Fulfillment Line 3, 4 black

Assume Order Management assigns an orchestration process to each fulfillment line, and that this process contains the following steps:

1. Create Scheduling
2. Create Reservation
3. Create Shipping
4. Wait for Shipping
5. Create Billing
6. Wait for Billing

Assume that, during setup, you set the following settings on this orchestration process.

Setting	Value
Change Mode	Advanced
Order Attributes That Identify Change	<p>You include the following attributes:</p> <ul style="list-style-type: none"> <li>• Ordered Quantity</li> <li>• Demand Class Code</li> <li>• Requested Ship Date</li> </ul>
Cost of Change	<p>You specify the following rule.</p> <p>If the fulfillment line status is:</p> <ul style="list-style-type: none"> <li>• <b>Reserved.</b> The cost of change is 15.</li> <li>• <b>Shipped.</b> The cost of change is 100.</li> </ul> <p>A lower number indicates a lower cost.</p>
Use Transactional Item Attributes	Contains a check mark.

Assume that, within one day of placing the sales order, one employee resigns and one employee transfers to another division of the company. The manager calls the customer service number and requests to change the sales order to fewer gifts and to

replace blue with beige. The customer service representative searches for the existing sales order, and then notices that the fulfillment lines are on the Create Shipping step. The representative submits the following change order:

- Change order line 1 to a quantity of 3, color red.
- Change order line 2 to a quantity of 3, color beige.
- Change order line 3 to a quantity of 4, color black.

Order Management uses the following sequence to process this change order:

1. Receives a request to change a sales order.
2. Performs a lookup to determine whether or not it already received the order key. If it already received this order key, then it exits this sequence.
3. Sends a request to the fulfillment system that is responsible for the task that is running. This request includes information about whether or not to stop processing, whether or not to accept the change, and the current status.
4. Determines whether or not processing constraints on the order header prevent change processing, and whether or not the sales order is closed. If these constraints do not allow a change, or if the sales order is closed, then Order Management rejects the entire change and exits this change sequence. This header evaluation prevents needless processing of a closed sales order. Order Management evaluates these attributes when the user creates the revision, and also when the user submits the revision to make sure that other fulfillment changes did not occur after the user created the revision that might affect the sales order.
5. Determines whether or not the processing constraints on each fulfillment line allow the change. If any constraint does not allow the change on any fulfillment line, then Order Management rejects the entire change and exits this sequence. Some processing constraints on a fulfillment line do not allow some changes, by default, such as updating an order line if the fulfillment line status is Shipped. In this example, only one day has elapsed since the user placed the sales order, the orchestration process has not shipped, and Order Management allows the change.
6. Calls the delta service. This service identifies the orchestration process steps that reference the attributes that Order Management uses to identify change. It also analyzes the state of each step that Order Management records every time it called a task layer service when the orchestration process was running. It uses this state information to determine the processing that Order Management must perform to incorporate the change.

Order Management compares the change order to the existing sales order to determine whether or not the value of an attribute that you specify on the Order Attributes That Identify Change page changed. In this example, the customer changed the quantity on two order lines, and you specified Ordered Quantity as a change attribute, so Order Management starts compensation. You also added a check mark to Use Transactional Item Attributes, so Order Management examines transactional item attributes for changes when it compares the change order to the original sales order. Color is one of these attributes, so the color change would also start compensation.

7. If the delta service determined that the order attributes require compensation, then Order Management uses the compensation pattern that the step references to compensate each of these steps. You can use the following compensation patterns:
  - Undo
  - Redo
  - Update
  - Cancel
  - None

You can also specify a business rule that determines the action to take according to this compensation pattern. Most orchestration process steps do not include a compensation pattern, and they use Update, by default. In this example, Order Management compensates each of the following steps.

Steps That Order Management Compensates	Description
Create Scheduling	Cancels the Create Scheduling step, and then creates a new instance of this step. Global Order Promising determines availability. Order Management replans fulfillment, and then assigns a revised date to each orchestration process step.
Create Reservation	<p>Assume the orchestration process in this example includes the following compensation pattern for the Create Reservation step:</p> <ul style="list-style-type: none"> <li>○ If the Demand Class Code is not Gold, then cancel the Create Reservation step, and then create a new instance of this step.</li> </ul> <p>This rule configures Order Management to release supply and creates a new reservation for all customers except Gold customers. Order Management also updates the Create Reservation step according to the new dates, and updates the reserved quantity.</p>
Create Shipping	Updates the Create Shipping step with the new dates and new item.

8. The compensating services run, and then finish. These services use the FIFO (first in, first out) sequence to compensate the sales order, according to the orchestration process sequence, by default. If Order Management must cancel the entire sales order, then it uses a LIFO (last in, first out) sequence.
9. If the original orchestration process:
  - **Can accommodate the change.** Order Management uses the original orchestration process to continue processing.
  - **Cannot accommodate the change.** Order Management cancels the original orchestration process, and then starts a new one that can accommodate the change.
10. Compensation finishes. Processing for this orchestration process is now at the same step that it was on when Order Management received the change order in step 1.

Order Management sends an update message to each external system to update the original message with the changed order that includes the modified attributes. For example:

Values	Description
Original Values	Quantity is 12 and Arrival Date is August 15, 2015
Values after compensation sends an Update Inventory message for the change order that includes the changed attributes.	Quantity is 10 and Arrival Date is August 18, 2015

## Managing Changes That Occur During Order Fulfillment: Explained

You can administer the following features to manage how Order Management Cloud processes changes that occur during order fulfillment, including how it does order compensation.

Feature That You Can Administer	Description
Change Mode	<p>An option that you can set on an orchestration process that determines how frequently Order Management records the state of this process. It compares these states during order compensation. You can set it to one of the following values:</p> <ul style="list-style-type: none"> <li>• <b>None.</b> Do not record the state of the orchestration process, and do not allow a change.</li> <li>• <b>Simple.</b> Record the state of the orchestration process when it starts and at the step where the orchestration process receives the change.</li> <li>• <b>Advanced.</b> Record the state of the orchestration process at each orchestration process step.</li> </ul>
Cost of Change Rule	<p>A rule that you can define on an orchestration process that specifies the business costs that your organization will incur as a result of processing the requested change.</p>
Use Flexfield Attributes	<p>An option that you can set on an orchestration process or orchestration process step that specifies whether or not to examine flexfield attributes when determining whether or not to compensate a sales order. For example, if you select Use Flexfield Attributes, then Order Management examines the flexfield attributes that are associated with an item to determine whether or not it must compensate the orchestration process.</p>
Use Transactional Item Attributes	<p>An option that you can set on an orchestration process or orchestration process step that specifies whether or not to examine transactional attributes when determining whether or not to compensate a sales order. For details, see Transactional Attributes: Explained.</p>
Use Dynamic Attributes	<p>An option that you can set that specifies whether or not to examine dynamic attributes when determining whether or not to compensate a sales order.</p>
Compensation Pattern	<p>A rule that you can define on an orchestration process step that specifies the adjustments to make to a sales order when this order changes. For example, if Order Management receives a change order that specifies to ship an item from a different warehouse for the Create Shipment step, then it runs the Cancel service and the Create service again. If you do not specify a compensation pattern, then Order Management uses the predefined compensation pattern, by default. A predefined pattern uses the Update service, the Cancel service, or the Create service.</p>
Task Type	<p>A field that you can set on an orchestration process step. The task type that you select determines the order attributes that Order Management uses when it determines whether or not it must compensate a sales order for this step.</p> <p>Order Management comes predefined to use a set of attributes for each predefined task type. You cannot modify this predefined configuration, but you can add more attributes. Order Management does not add any of these attributes to a new task type that you add, and it does not evaluate this task until you add these attributes. To add them, you must click <b>Add All</b>.</p>
Order attribute	<p>You can administer Order Management so that it examines order attributes when it determines whether or not to compensate a sales order. For details, see Managing Order Attributes That Identify Change: Procedure.</p>
Processing constraints that control change	<p>A processing constraint that you can specify to control the changes that Order Management makes to a sales order, such as what data a user can modify, or when the user can modify data. If the user attempts to modify this data, then Order Management displays an error message. For details, see Processing Constraints: Explained.</p>

## Managing Order Attributes That Identify Change: Procedure

Order Management Cloud examines order attributes to identify the changes that a change order contains, and then uses this information to compensate a sales order.

For example, Ordered Quantity is an order attribute. If a user increases the Ordered Quantity of a sales order, then Order Management must schedule and ship more supply. To do this work, it compensates the Schedule step and the Shipment step of the orchestration process when it receives the change order that includes the new quantity. Each step references a task type, and each task type references order attributes that Order Management uses to determine whether or not to compensate a sales order. In this example, the Schedule step and the Shipment step each reference a task type that references the Ordered Quantity attribute.

In this topic, assume you are creating an orchestration process step that must use the Ordered Quantity attribute to identify change. This topic includes example values. You might use different values, depending on your business requirements.

Manage order attributes that identify change:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, click **Orders**, search for, and then open Manage Order Attributes That Identify Change.
4. In the Task Type list, click a task type that looks like it might meet the needs that your orchestration process step requires. Order Management comes predefined with a number of task types.

To reduce maintenance, it is recommended that you use one of these predefined task types before you create a custom task type.

5. Click **Actions**, and then click **Edit**. For this example, click **Shipment**.
6. On the Edit Order Attributes That Identify Change page, in the Orchestration Components list, click a component that you think might contain the attribute that Order Management must use to identify the change. For this example, click **Order Line**.
7. Examine the Attributes list.

When your orchestration process step runs, it will examine each attribute that this list contains to determine whether or not to compensate the order when it receives a change order. If this list includes the attributes that your step requires to identify change, then set the Task Type field of your orchestration process step to this task type, and then exit this procedure. For this example, the list does include Ordered Quantity, so you can set the Task Type field to Shipment. If the Predefined column of the Attributes list includes a check mark, then you cannot remove this attribute.

8. If the Attributes list does not include the attributes that your step requires to identify change, then do the following work:
  - o Click **Actions**, and then click **Select and Add**.
  - o In the Change Attributes dialog, click an attribute, such as **Scheduled Ship Date**.
  - o Click **Apply**, click **OK**, and then exit this procedure.

Note that you can also do the following work on the Manage Order Attributes That Identify Change page:

- To add an attribute to all tasks that the Task Type list displays, click **Actions**, and then click **Edit All**.
- To add a new task type, click **Actions**, and then click **Create**. On the Create Order Attributes That Identify Change page, add attributes, as necessary. If you create a new task type, and if you do not use this page to add at least one attribute to this task type, then Order Management will not examine any attributes when it identifies change.

## Measuring the Cost of Change: Explained

Cost of change is a numerical value that measures how much a change impacts an orchestration process. For example, the monetary cost to a company, or the difficulty that is associated with incorporating the change.

If the order capture system requests a determination for a cost of change, then Order Management Cloud calculates this value, and then returns it to the order capture system so that the customer service representative can choose whether or not to proceed with the change. The source system can request this value before it submits an order. Order Management also calculates this cost after it compensates a sales order.

You use a business rule to assign the cost of change to an orchestration process. If you choose not to use values for the cost of change, then Order Management uses a value of zero to calculate the cost.

## Measuring the Cost of Change: Procedure

You can create a rule that measures the cost of change for an orchestration process.

In this example, assume the order administrator of a flooring company requires a rule that measures the cost of change that the company will incur when a customer requests a change. If the fulfillment line status is:

- Scheduled, then the cost is low
- Shipped, then the cost is high

To create cost of change rules for this example, you create If and Then statements that implement the following rules:

- If the fulfillment line status is Scheduled, then the cost of change is 5.
- If the fulfillment line status is Shipped, then the cost of change is 50.

This example includes a simple rule that you can use with an orchestration process that includes only one line. To write a rule for an orchestration process that includes multiple lines, it is recommended that you use advanced rules. For details, see the topic that describes Usage of Oracle Business Rules in Article ID 1536633.1 on My Oracle Support at <https://support.oracle.com>.

This topic includes example values. You might use different values, depending on your business requirements.

### Summary of the Work

To create cost of change rules for this example, do the following work:

1. Configure your order capture system.
2. Create the If statement for the first rule.
3. Create the Then statement for the first rule.
4. Create the If statement for the second rule.
5. Create the Then statement for the second rule.
6. Test your work.

### Configure Your Order Capture System

Order Management Cloud does not display the result when it calculates the cost of change. Instead, you must configure your order capture system to get this result from Order Management, and then display it so that someone who uses this system can take the necessary action. To get this result, you can use the `GetOrderDetails` method of the `GetOrchestrationOrder`

service. For details about how to configure your order capture system, see the paper that describes how to call web services from an external order capture system in Article ID 1536633.1 on My Oracle Support, at <https://support.oracle.com>.

## Create the If Statement for the First Rule

In this section you create a business rule. For a demonstration, see *Creating Business Rules: Demo*.

Create the If statement for the first rule:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, search for, and then open Manage Orchestration Process Definitions.
4. On the Manage Orchestration Process Definitions page, locate the orchestration process where you must add the cost of change rule, and then open it for editing.
5. In the Orchestration Process area, click **Click for Rule** next to Cost of Change Rule.
6. In the Cost of Change Rule dialog, click **Add Rule**, and then click **Expand**.
7. Click **Left Value**.
8. In the Condition Browser dialog, expand **DooSeededOrchestrationRules**, expand **DOOFLine**, click **StatusCode**, and then click **OK**.
9. In the Right Value field, enter "**SCHEDULED**". You must include the double quotation marks.

## Create the Then Statement for the First Rule

Create the Then statement for the first rule:

1. In the Cost of Change Rule dialog, in the Then area, click **Add Action**, and then click **Assert New**.
2. Click **Select a Target**, and then click **DooSeededOrchestrationRules.Result**.
3. Click **Edit Properties**.
4. In the Properties dialog, in the resultObjKey row, enter the following value, and then click **OK**.

Field	Value
Value	5

## Create the If Statement for the Second Rule

Create the If statement for the second rule:

1. In the Cost of Change Rule dialog, click **Add Rule**, and then click **Expand**.
2. Click **Left Value**.
3. In the Condition Browser dialog, expand **DooSeededOrchestrationRules**, expand **DOOFLine**, click **StatusCode**, and then click **OK**.
4. In the Right Value field, enter "**SHIPPED**". You must include the double quotation marks.

## Create the Then Statement for the Second Rule

Create the Then statement for the second rule:

1. In the Cost of Change Rule dialog, in the Then area for Rule 2, click **Add Action**, and then click **Assert New**.
2. Click **Select a Target**, and then click **DooSeededOrchestrationRules.Result**.
3. Click **Edit Properties**.
4. In the Properties dialog, in the resultObjKey row, in the Value column, enter **50**, and then click **OK**.
5. In the Cost of Change Rule dialog, click **Save**.

6. On the Edit Orchestration Process Definition page, click **Save**.

## Test Your Work

Test your work:

1. Change a sales order that includes a fulfillment line status that is Scheduled.
2. Wait for Order Management to process the change.
3. Make sure your order capture system displays a value of 5 for the cost of change.
4. Change a sales order that includes a fulfillment line status that is Shipped.
5. Wait for Order Management to process the change.
6. Make sure your order capture system displays a value of 50 for the cost of change.

### Related Topics

- [Defining Orchestration Processes: Procedure](#)
- [Creating Business Rules: Demo](#)

## Compensating Sales Orders That Change: Procedure

You can create a compensation pattern that specifies the adjustments that Order Management Cloud must make when it processes a fulfillment task. It does this work to process the request it receives to change a sales order.

In this topic, assume you are an order administrator who requires a compensation pattern that allows sufficient time to reschedule an orchestration process when a customer requests to change the ship date. You create the following rule:

- If the requested ship date is less than or equal to the current date plus three days, then cancel, and then redo the Ship Product step.

This example includes a simple rule that you can use with an orchestration process that includes only one line. To write a rule for an orchestration process that includes multiple lines, it is recommended that you use advanced rules. For details, see the topic that describes Usage of Oracle Business Rules in DOO Order Fulfillment (Article ID 1536633.1) on My Oracle Support at <https://support.oracle.com>.

This topic includes example values. You might use different values, depending on your business requirements.

## Summary of the Work

To compensate sales orders that change, do the following work:

1. Create the If statement.
2. Create the Then statement.

## Create the If Statement

This statement specifies the following condition:

- If the requested ship date is less than or equal to the current date plus three days

For a demonstration, see [Creating Business Rules: Demo](#).

Create the If statement:

1. In the Navigator, click **Setup and Maintenance**.

2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, search for, and then open Manage Orchestration Process Definitions.
4. On the Manage Orchestration Process Definitions page, locate the CallCustomerWhenLargeInvoice orchestration process, click **Actions**, and then click **Edit**.

For this example, you edit the CallCustomerWhenLargeInvoice orchestration process. For details about how to create this process, see Branching Orchestration Processes: Procedure.

5. On the Edit Orchestration Process Definitions page, in the Process Details area, in the Step Definition list, click **Change Management**.
6. In the Ship Product row, in the Compensation Pattern column, click **Click for Rule**.
7. In the Compensation Pattern dialog, click **Add Rule**, and then click **Expand**.
8. In the If area, click **Left Value**, expand **DOOSeededOrchestrationRules**, expand **DOOFlne**, expand **requestShipDate**, click **time**, and then click **OK**.
9. In the Is list, click **Is**, and then click **Same or Less Than**.
10. Click **Right Value**, expand **CurrentDate**, expand **Date**, expand **Time**, and then click **timeInMillis**.

Do not click OK.

11. In the field near the top of the Condition Browser, add `+3*24*60*60*1000` after `CurrentDate.date.timeInMillis`.

Make sure this field now contains the following value:

```
CurrentDate.date.timeInMillis+3*24*60*60*1000
```

The equation  $3*24*60*60*1000$  calculates the total number of milliseconds that three days contains.

12. Click **OK**.

## Create the Then Statement

This statement cancels, and then restarts the Ship Product step so that Order Management can reschedule the shipment. Create the Then statement:

1. In the Then area, click **Add Action**, and then click **Assert New**.
2. Click **Select a Target**, and then click **DooSeededOrchestrationRules.Result**.
3. Click **Edit Properties**.
4. In the Properties dialog, in the ResultObj row, enter the following value. You must include the double quotation marks ("").

Field	Value
Value	"CANCEL_ CREATE"

5. Click **OK**.
6. Click **Validate**, make sure the Validation Log that displays does not contain any errors, and then click **Save**.
7. On the Edit Orchestration Process Definition page, click **Save**.

### Related Topics

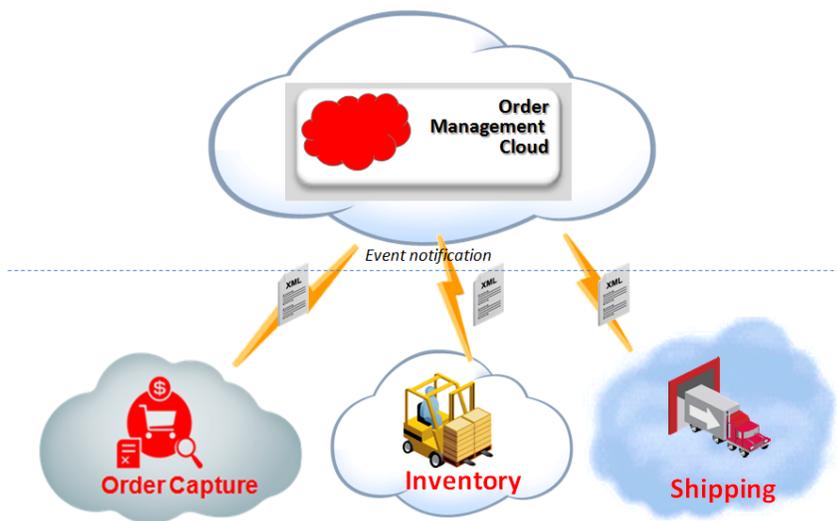
- [Creating Business Rules: Demo](#)

## Notifying Systems When Orders Change

## Sending Notifications from Order Management to External Systems: Explained

Order Management Cloud uses a business event to send a notification to an external system when a change occurs, such as when information in a sales order or fulfillment line changes. You can control these business events.

You can use a web service to communicate information to an upstream system, such as a source system, or to a downstream system, such as a fulfillment system. You can also use a predefined business event to send information to these systems about a significant event that occurs in the context of a business process. The following diagram illustrates how you can use a business event to communicate with these systems.



A business event can notify a system when one of the following changes occur:

- A change to an order attribute occurs.
- A hold is applied.
- A jeopardy priority changes.
- A fulfillment line splits.
- The status on an order header changes.
- A fulfillment line closes.
- A fulfillment line achieves a status.
- Compensation for an orchestration process finishes successfully or not successfully.

You can specify whether or not the business event is sufficiently important to communicate in the context of your business requirements.

## Managing Business Event Trigger Points

You can use the Manage Business Event Trigger Points page in the Setup and Maintenance work area to configure a business event trigger point. Note the following:

- You must associate a connector with any business event that you specify. For details, see [Sending Notifications from Order Management Cloud to External Systems: Procedure](#).
- The business events that this page displays are not active, by default, except for business events that monitor closing a fulfillment line.
- A modification that you make on this page takes effect immediately.
- The settings that you make affect only one instance of Order Management.
- You cannot add or delete the business events that this page displays.
- It is recommended that you configure only the business event trigger points that you require. Each business event trigger point might affect system performance.

## Using Predefined Business Events

Order Management comes predefined with the following business events. You can specify them on the Manage Business Event Trigger Points page.

Business Event	Description
Change Order Compensation Complete	Sends a notification when Order Management finishes processing for a change order. If a change order results in an error, then this business event reports the error.
Fulfillment Line Status Update	<p>Sends a notification when Order Management changes the status value of a fulfillment line. To specify the status values to notify, do the following:</p> <ol style="list-style-type: none"> <li>1. Navigate to the Manage Orchestration Process Definitions page.</li> <li>2. On the Edit Status Rule Set page, add a check mark to the Notify External System option next to each status value to notify.</li> </ol>
Fulfillment Line Closed	Sends a notification when Order Management closes a fulfillment line. Some applications, such as Cost Management, can use this information to perform downstream processing on the fulfillment line.
Hold	<p>Sends a notification when a user applies a hold, or when the order capture system or fulfillment system applies a hold on one of the following items:</p> <ul style="list-style-type: none"> <li>• Order</li> <li>• Order line</li> <li>• Fulfillment line</li> </ul> <p>Note the following:</p> <ul style="list-style-type: none"> <li>• If a hold is not immediately effective, for example, if the hold applies to a future task, then Order Management communicates a business event when the hold becomes effective, not when the request to add the hold occurs.</li> <li>• If change management releases or applies a hold, then Order Management does not communicate a business event.</li> </ul>
Jeopardy	Sends a notification when process planning changes a jeopardy priority value, such as High, Medium, or Low. You can specify the value to notify.

Business Event	Description
Order Attribute Update	<p>Sends a notification when Order Management changes a fulfillment line attribute during order fulfillment. You can choose each attribute from the set of attributes that a task references.</p> <p>Note the following:</p> <ul style="list-style-type: none"><li>• Each attribute that the Order Update: Attributes That Trigger Events area displays can trigger a business event. These attributes are predefined. You can also configure an extensible flexfield that modifies these attributes. If you do configure an extensible flexfield, then Order Management displays it in the Select and Add: Attributes That Trigger Events dialog. If you select one or more segments, and then click OK or Apply, then Order Management displays the segments that you select in the Order Attribute Update: Attributes That Trigger Events area.</li><li>• If Order Management updates more than one attribute of the Order Attribute Update business event at the same time, then it sends only one notification. This notification includes information about the attributes that it updated.</li><li>• The status values on the Order Header Status Update: Status Values That Trigger Events area start the Order Header Status Update business event. These values are predefined. You cannot add status values in this area.</li></ul>
Order Header Status Update	<p>Sends a notification when Order Management cancels, partially closes, or closes a sales order. You can specify the status value to notify.</p>
Split	<p>Sends a notification when a partial shipment occurs or when an integrated system splits a fulfillment line.</p>

## Sending Notifications From Order Management Cloud to External Systems: Procedure

Order Management Cloud uses a connector to communicate with your external system. The connector specifies the URL that locates your external system and the security credentials that your external system requires during communication. You must configure this connector, and then associate it with the business event that sends notifications to your external system.

Order Management monitors the conditions that your business events specify during order fulfillment. If Order Management determines that it must send a notification according to one of these business events, then it communicates this information to the endpoint URL for each connector that you associate with the event trigger point. For example, if you add a check mark to the Notify External System option for the Shipped status value on a status rule set, then Order Management communicates a business event any time it sets a fulfillment line status to Shipped, and it sends this business event to the endpoint URL of the connector that you associate with a business event named Fulfillment Line Status Update. For details about this option, see Adding Status Conditions to Fulfillment Lines: Procedure.

### Summary of the Work

To configure Order Management to send notifications to external systems, do the following work:

1. Configure the connector.
2. Manage business event trigger points.
3. Set the notification frequency.

## Configure the Connector

Configure the connector that Order Management uses to communicate with your external system:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, search for, and then open Manage External Interface Web Service Details.
4. On the Manage Connector Details page, click **Actions**, click **Add Row**, and then set the following values.

Field	Description
Target System	Select the target system that must receive the notification. This system subscribes to the business event.
Connector Name	Enter text that describes the purpose of the connector. For example, to indicate that you use this connector to send notifications to a legacy system, enter LEG_ Notification.
Connector URL	Specify an endpoint URL. This URL is the address of a web service that you deploy on the subscriber system. Order Management will call this web service, and the web service must accept the event payload that Order Management sends.
User Name	Enter the user name that the subscriber system requires.
Password	Enter the password that the subscriber system requires.

## Manage Business Event Trigger Points

Manage the business event trigger points that determine when to communicate business events to your external system:

1. On the Search page, search for, and then open Manage Business Event Trigger Points.
2. On the Manage Business Event Trigger Points page, choose a trigger point.

At runtime, if an order reaches the trigger point that you specify, then Order Management calls the web service that you referenced in the Connector URL field. This web service then sends details of the business event to the subscriber system. For example, to send a notification when a fulfillment line closes, choose the Fulfillment Line Closed trigger point.

3. Click **Actions**, and then click **Add Row**.

Note that some trigger points include an Associated Connectors tab in the details area. If this tab displays, then click it to add the connector.

4. Set the following values.

Field	Description
Connector Name	Select the connector that you configured in the Configure the Connector topic.
Override Default Visibility	Set one of the following values: <ul style="list-style-type: none"> <li>○ <b>Contains a check mark.</b> Send notifications about each sales order even if the target system that receives the notification does not contain knowledge about this sales order.</li> <li>○ <b>Does not contain a check mark.</b> Do not send notifications about each sales order to a target system that does not already contain knowledge about this sales order.</li> </ul>

Field	Description
	Assume your implementation includes Source System 1 and Source System 2, and that each of these systems can send a source order to Order Management Cloud. Assume that Source System 2 has no knowledge of source orders that originate in Source System 1, and that you do not want to notify Source System 2 when an event occurs that is associated with a source order that originates in Source System 1. In this situation, if Override Default Visibility does not contain a check mark, then Order Management Cloud will not call the connector service for the subscriber system, and Source System 2 will not receive the notification.

- Optional. Repeat step 3 through step 4.  
You can associate the same connector with multiple business events, as necessary.
- Optional. Repeat step 2 through 5 for each trigger point that you must administer.
- Click **Save and Close**.

## Set the Notification Frequency

You can use the Notify Frequency parameter to specify how frequently Order Management sends notifications. For details, see [Managing Order Management Parameters: Procedure](#).

### Related Topics

- [Managing Order Management Parameters: Procedure](#)

# Using Jeopardy to Manage Delays

## Jeopardy Priority: Explained

Jeopardy indicates the severity of a delay of a task in an orchestration process. You can modify the predefined range of jeopardy scores for a jeopardy priority to control how Order Management Cloud calculates and displays jeopardy.

For example, you can specify a minimum jeopardy score of 0, and a maximum score of 100 for the Low jeopardy priority. You cannot add or delete a jeopardy priority. You cannot customize the value of a predefined jeopardy priority, such as Low.

Note the following:

- Order Management uses forward planning and backward planning across an orchestration process to calculate the promise dates for each orchestration process task. If one of these tasks is delayed, then Order Management uses jeopardy to indicate the severity of the delay according to the jeopardy settings that you make.
- Order Management calculates the jeopardy score and jeopardy priority every time it plans or replans an orchestration process.
- You can use jeopardy on planned dates or on actual dates. This configuration allows an order manager to become aware that a jeopardy condition might exist before it actually occurs, and to take action to fix the condition and reduce the jeopardy.

## Jeopardy Score: Explained

Order Management Cloud maps the jeopardy score to a jeopardy priority, and then displays this score in the Order Management work area. This score provides a quick visual cue so that you can take action to reduce the delay.

## How Order Management Displays Jeopardy Priority

A jeopardy priority references the fulfillment task that contains the highest jeopardy score. If multiple fulfillment tasks are in jeopardy in an orchestration process, then Order Management uses the highest jeopardy score when it displays the jeopardy for this orchestration process.

For example, assume the supply in the warehouse that the Carpet Processing orchestration process references is not sufficient, and that this insufficient supply causes a delay for the following fulfillment tasks:

- A three day delay for the Deliver Carpet task that results in a jeopardy score of 100 and a jeopardy priority of Medium
- A three day delay for the Invoice Carpet task that results in a jeopardy score of 200 and a jeopardy priority of High

Two hundred is the higher score, so Order Management uses the jeopardy score for the Invoice Carpet task to represent the jeopardy of the Carpet Processing orchestration process. It displays a jeopardy priority of High in the Order Management work area.

## How Order Management Assigns Jeopardy Score to Fulfillment Tasks

Order Management assigns a jeopardy score to a fulfillment task according to the jeopardy threshold. If a fulfillment task is delayed, then Order Management calculates the difference between the required completion date and the planned completion date, and then searches for a threshold that applies to the largest number of entities that the fulfillment task references. It uses the following sequence when it searches for this threshold:

1. Search the process name, process version, task name, and task type.
2. Search the process name, process version, and task name.
3. Search the process name and task name.
4. Search the process name, process version, and task type.
5. Search the process name and task type.
6. Search the process task name.
7. Search the process name and process version.
8. Search the process name.
9. Search the task type.

For example, Order Management does the following work:

1. Search for a jeopardy threshold that applies to the task type, task name, process name, and process version. If Order Management does not find a threshold that applies to all of these attributes, then continue to step 2.
2. Search for a jeopardy threshold that applies to the process name, process version, and task name. If Order Management does not find a threshold that applies to all of these attributes, then continue to step 3.
3. And so on.

Order Management continues this work until it finds a threshold that applies to one attribute or a combination of attributes, and then sets the jeopardy score according to the threshold that it assigns to the fulfillment task.

The jeopardy score might change during replanning. If this change results in a change to the jeopardy priority, and if this jeopardy priority value is enabled, then Order Management updates the jeopardy values that it displays.

### *Related Topics*

- [Transforming Order Lines to Fulfillment Lines: How it Works](#)

## Jeopardy Threshold: Explained

You can use a jeopardy threshold to measure and monitor an orchestration process. You can define a set of ranges for each fulfillment task of an orchestration process, and then assign a score that indicates the severity of the delay. Order Management Cloud uses this information in the indicators that it displays on the Order Management work area. These indicators can help an order manager to quickly determine the severity of a delay, and then to take an appropriate action.

Note the following:

- Order Management uses the lead time of each orchestration process step and the dates from the sales order, such as required completion date, when it plans an orchestration process and assigns it to this sales order.
- Each fulfillment task includes a planned start date and a completion date.
- If a fulfillment task is delayed, then Order Management replans the entire orchestration process.
- If Order Management expects that a fulfillment task will finish after the required completion date of this task, then it assigns a jeopardy score to each fulfillment task according to the jeopardy threshold that you define.

You can define a jeopardy threshold for any combination of the following attributes. If you leave these attributes at their default setting of All, then Order Management applies the jeopardy threshold to all fulfillment tasks:

- Task type
- Task name
- Process name
- Process version

If you apply a jeopardy threshold, then you must first define the orchestration process, fulfillment task, and task type that this threshold must reference so that you can select these objects when you apply the threshold.

## Defining Jeopardy and Lead Time to Manage Delay: Procedure

To manage delay, you can define jeopardy priorities for your implementation, and then define lead times and jeopardy thresholds for each orchestration process.

### Summary of the Work

To define jeopardy and lead time to manage delay, do the following work:

1. Manage jeopardy priorities.
2. Specify lead time for orchestration process steps.
3. Manage jeopardy thresholds.

This topic describes how to manage jeopardy priorities, lead times, and jeopardy thresholds because you often do this work together. However, you can manage them independently of one another. This topic includes example values. You might use different values, depending on your business requirements.

### Manage Jeopardy Priorities

Assume your business requirement specifies a higher range for the MEDIUM jeopardy priority, and a higher but more narrow range for the HIGH jeopardy priority. You can manage these priorities. For details, see Jeopardy Priority: Explained.

Manage jeopardy priorities:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, search for Manage Jeopardy Priorities, and then open it.
4. On the Manage Jeopardy Priorities page, set the following values, and then click **Save and Close**.

Priority	Minimum Score	Maximum Score
LOW	10	199
MEDIUM	200	399
HIGH	400	600

It is recommended that you do not:

- o Overlap scores across priorities. For example, do not configure a Low priority of 10 to 210, and a Medium priority of 200 to 400. In this example, an overlap exists between 200 and 210.
- o Configure priorities that intersect at the same value. If you configure two priorities that intersect at the same value, then Order Management assigns the intersecting value to the lower priority. For example, if you configure a Low priority of 10 to 200, and a Medium priority of 200 to 400, then Order Management considers a score of 200 as a Low priority.

## Specify Lead Time for Orchestration Process Steps

Assume that, through research, you determine that your implementation requires a specific amount of lead time to complete each orchestration process step in an orchestration process that ships goods. You can specify these lead times. For details about lead times, see Defining Orchestration Process Steps: Points to Consider.

Specify lead time for orchestration process steps:

1. On the Setup page, search for, and then open Manage Orchestration Process Definitions.
2. On the Manage Orchestration Process Definitions page, search for an orchestration process, and then open it.

For this example, search for Orchestration\_Process\_1. For details about how to create this process, see Pausing Orchestration Processes Until an Event Occurs: Procedure.

3. On the Step Definition tab, set the Default Lead Time for each of the following steps. Set the Lead Time UOM to Days for each of these steps.

Step	Default Lead Time
Schedule Goods	2
Reserve Goods step	1
Ship Goods	4
Wait for Ship Goods	6
Prepare Documentation	1

Step	Default Lead Time
Wait for Documentation	1
Invoice Goods	3
Wait for Invoice Goods	2

4. Click **Save and Close**.

## Manage Jeopardy Thresholds

In this example, to meet your business requirements, assume you determine that you can use thresholds on the reservation tasks that your orchestration process performs as a way to manage how long this process can delay each of these tasks. For details, see Jeopardy Threshold: Explained.

Manage jeopardy thresholds:

1. On the Setup page, search for, and then open Manage Jeopardy Thresholds.
2. On the Manage Jeopardy Threshold Definitions page, click **Actions**, and then click **Create**.
3. On the Create Jeopardy Threshold Definition page, set the following values.

Field	Value
Code	Enter any text that Order Management can use as an abbreviation for this threshold. Order Management uses this text as a code to identify this threshold throughout the user interface, such as in lists. For this example, enter <b>t-shirt_threshold</b> .
Name	Clothing Jeopardy
Description	Jeopardy thresholds for company t-shirts.
Process Name	Choose the name of the orchestration process where Order Management must apply this jeopardy threshold. For this example, choose <code>Orchestration_Process_1</code> . For details about how to create this process, see Pausing Orchestration Processes Until an Event Occurs: Procedure
Task Name	Reserve
Task Type	Reservation

4. Set the threshold for the Low range of the jeopardy priority. Click **Actions**, click **Add Row**, and then set the following values.

Field	Value
Maximum Delay	2
Maximum UOM	Days

Field	Value
Jeopardy Score	Enter a value that resides in the Low range of the jeopardy priority that you modified above, such as 100.
Description	Threshold for the Low range of the jeopardy priority.

5. Set the threshold for the Medium range of the jeopardy priority. Click **Actions**, click **Add Row**, and then set the following values.

Field	Value
Maximum Delay	5
Maximum UOM	Days
Jeopardy Score	Enter a value that resides in the Medium range of the jeopardy priority that you modified above, such as 300.
Description	Threshold for the Medium range of the jeopardy priority.

6. Set the threshold for the High range of the jeopardy priority. Click **Actions**, click **Add Row**, and then set the following values.

Field	Value
Maximum Delay	8
Maximum UOM	Days
Jeopardy Score	Enter a value that resides in the High range of the jeopardy priority that you modified above, such as 500.
Description	Threshold for the High range of the jeopardy priority.

7. Click **Save and Close**.

#### Related Topics

- [Defining Orchestration Processes: Explained](#)
- [Pausing Orchestration Processes Until an Event Occurs: Procedure](#)
- [Defining Orchestration Process Steps: Points to Consider](#)

## Customizing Task Types

## Task Types: Explained

You can use a task type to determine how to process a sales order.

A task type includes the following tasks and services:

- An internal web service that communicates with the task layer
- A task that represents the services that the task type uses

Note the following:

- Each task type is read-only. You cannot modify or delete it.
- You can create a custom task type. For details, see [Creating Custom Task Types: Procedure](#).
- You can modify the name of a task type that you create, but it is not recommended.
- You can edit the service names of an activity or a custom task type that you create.
- You can add a service from the services that are available, but you cannot edit or delete the services that a custom task type references.
- You can use a task to represent the services that a task type references. For example, if you create a Ship Goods task that references services from the Shipment task type, then the Order Management work area displays Ship Goods when a Shipment service runs, regardless of whether or not it calls the Create Shipment service or the Update Shipment service. The Order Management work area does not display the service. It displays only the task.
- You can define several tasks for a task type, such as ShipGoods or ShipWidgets.

Order Management Cloud comes predefined with the following task types. For details about how Order Management uses these task types, see [Task Layer: Explained](#).

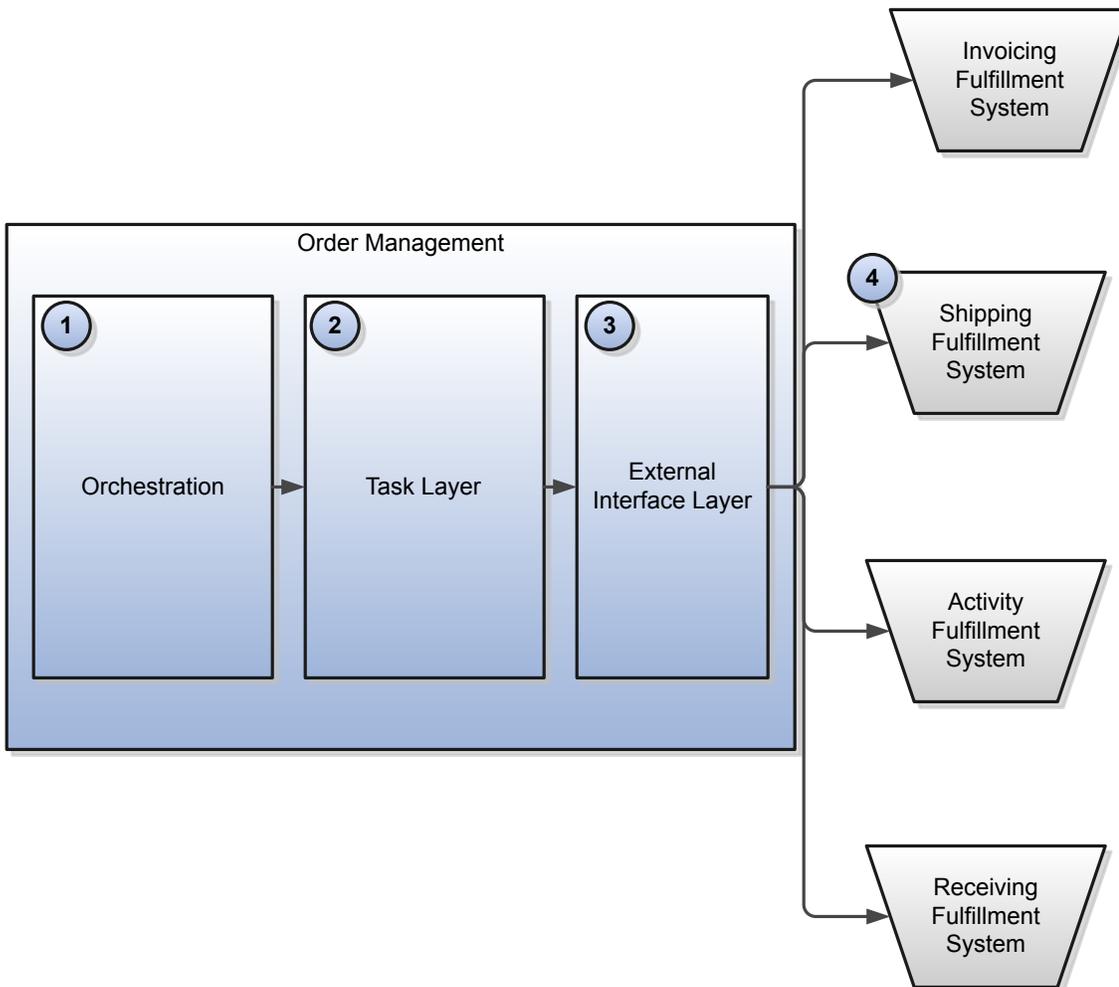
Task Type	Description
Activity	Set of services that communicate with a fulfillment system to cause a human activity, such as an installation.
Fulfillment Order	Set of services that integrate between Order Management and an Enterprise Resource Planning (ERP) system. Note that the fulfillment task layer can run multiple fulfillment actions through a single request, such as shipment and invoice.
Invoice	Set of services that communicate with a billing system to create invoices for an order line.
Pause	Set of services that temporarily pauses processing to wait until a date or event occurs before proceeding to the next orchestration process step. You can use the pause task to coordinate processing across multiple fulfillment lines in a single sales order.
Procurement	Set of services that source and ship the items that a fulfillment line references when these items reside in an outside organization.
Schedule	Set of services that schedules a fulfillment line.
Supply	Set of services that communicate with Oracle Fusion Supply Chain Orchestration so that you can use more complex logic when you source the items that a fulfillment line references.
Reservation	Set of services that reserve the items that a fulfillment line references.

Task Type	Description
Return	Set of services that communicate with a fulfillment system to return the items that a fulfillment line references.
Shipment	Set of services that communicate with a shipping fulfillment system to ship the items that a fulfillment line references.

## Task Layer: Explained

The task layer that Order Management Cloud uses is a logical component that contains services that fulfill fulfillment requests. Some other Order Management component or an external component can request one of these services.

The following diagram illustrates an example of how Order Management uses the Order Orchestration task layer to fulfill a shipment request during order fulfillment.



## Explanation of Callouts

1. Orchestration receives a request to call a task layer service. For example, Create Shipment Request. Orchestration then runs an orchestration process that processes shipment requests. At some point during processing, this process sends a request to the Task Layer to run a task layer service named Create Shipment Request.
2. The Task Layer runs the Create Shipment Request service. This service sends a request to an external interface layer named Create Shipment Request.
3. The external interface layer named Create Shipment Request sends a request to the Shipping fulfillment system.
4. The Shipping fulfillment system fulfills the shipment request.

Order Management typically uses the same process to fulfill other requests. For example, to complete an invoice, Orchestration runs an orchestration process that processes invoice requests, the Task Layer runs a service that processes a request to complete an invoice, and so on.

## Activity Service

The Activity service sends an activity request to the activity fulfillment system. This system creates and fulfills the activity, and then sends replies and updates to the activity service. The activity service interprets these replies and updates.

Note the following:

- Each activity includes attributes, such as subject, activity type, earliest start date, due date, scheduled duration, actual duration, percent completion, and assignee.
- Order Management can associate an activity with one or more fulfillment lines.
- The activity service does not typically immediately fulfill an activity task, so a wait service allows the orchestration process to wait for the activity to complete.
- Order Management does not support partial fulfillment. An activity step must finish before the orchestration process can proceed to the next orchestration process step.

The Activity service provides the following features.

Feature	Description
Send requests to the activities management system	<p>Sends a Create Activity request to the activities management system that creates the activity.</p> <p>If the task layer service receives a change order, then it changes or cancels the activity, as necessary.</p> <p>If Order Management applies a hold on a sales order, then the activity task layer service sends a request to the activities management system to hold the activity that is currently in progress.</p>
Receive activity status updates	You can schedule an orchestration process that periodically gets the updated activity status.
Customize activities	<p>An activity is a predefined task type. You can create a custom activity task type. You can use an extensible flexfield to modify an activity.</p> <p>You can enable an activity default in a task type or orchestration process step. Defaulting the activity type allows you to categorize activities so that the activity management system can perform the business logic and do validation according to the activity type. If you do not set the default value for the task type or the orchestration process step, then Order Management sets the subject of the activity to the name of this step.</p> <p>Some activity management systems allow you to create an activity template that supports a human task. To use an activity template in an activity management system, you can specify the activity template for a task type or an orchestration process step that creates the activity according to the predefined template when you define the task type and the orchestration process.</p>

## Reserve, Schedule, and Hold Services

The Reserve service, Schedule service, and Hold service provides the following features.

Service Name	Description
Reserve Service	Sends a reservation request to an inventory fulfillment system. A reservation reserves the supply for a sales order so that no other sales order or system can use this supply.
Schedule Service	Sends a scheduling request to order promising. For example, to schedule a sales order, unschedule a sales order, or to check the availability of the item. This scheduling applies to fulfillment lines that are waiting for manual scheduling and for fulfillment lines that fail scheduling in the automated or manual process. This scheduling works only for unscheduled fulfillment lines. Order Management does not allow automatic rescheduling from the Order Management work area.
Hold Service	Sends a hold request to a fulfillment system. For example, the HoldShipmentRequest service can send a hold request from Order Management to a shipping fulfillment system. In this example, the Shipping task layer service contains the hold.

## Shipment Service

The Shipment service sends a shipment request to a shipping system. Most shipping services start in orchestration. The Shipment service provides the following features.

Feature	Description
Send a request to the fulfillment system that creates the shipment request	<p>If Order Management receives a change order, then the shipment service changes or cancels the shipment request, if necessary.</p> <p>If someone applies a shipping hold, and if the Hold on Running Task feature is selected, and if the task that is running is a Shipping task, then the shipment service sends a request to the shipping fulfillment system to hold the shipment request that it is currently processing.</p>
Consolidate fulfillment lines	<p>Consolidates the fulfillment lines of:</p> <ul style="list-style-type: none"> <li>• A shipment set, and then sends them to the shipping fulfillment system together.</li> <li>• A configured item, and then sends them to the shipping fulfillment system together.</li> </ul>
Receive fulfillment line details and status updates from the shipping system and update business objects in Order Management.	<p>The shipping system might send fulfillment line details to Order Management at any time when a status update occurs, including before it confirms a shipment. These details might include freight cost, tracking number, way bill number, and so on.</p> <p>Order Management interprets the update that it gets from the shipping system, and then uses one of the following predefined values to update the status:</p> <ul style="list-style-type: none"> <li>• Picked</li> <li>• Packed</li> <li>• Shipped</li> <li>• Backordered</li> </ul>

Feature	Description
	<p>Note the following:</p> <ul style="list-style-type: none"> <li>• The shipment service continues to interpret the updates that it receives from the shipping system even after the shipping system ships the fulfillment lines. It sends information about these updates to Order Management.</li> <li>• If the shipping system uses multiple currencies to represent cost, then the shipment service converts them before it sends the update to Order Management.</li> <li>• If the shipping system uses a unit of measure to represent shipping that is different from the unit of measure that Order Management uses in the sales order, then the fulfillment system converts this unit of measure back to the unit of measure that the sales order uses, and then communicates the shipped quantity to Order Management.</li> </ul>
<p>Split a fulfillment line, a shipment set, or a configured item when only part of a shipment ships</p>	<p>If only part of a fulfillment line ships, then the shipment service splits this line into the following lines:</p> <ul style="list-style-type: none"> <li>• One fulfillment line that includes the quantity that was shipped</li> <li>• Another fulfillment line that includes the quantity that was not shipped</li> </ul> <p>If only some fulfillment lines of a shipment set ship, then the shipment service removes the unshipped lines from the shipment set.</p> <p>If half of the fulfillment lines of a configured item ships, then the task layer service splits the configured item into a shipped item and an unshipped item. If the split is not proportional, then a remnant results.</p>

## Invoice Service

The Invoice service sends requests to a billing system and interprets the replies that it receives from this system. The billing system creates the invoice and the credit transactions. Order Management does not allow you to modify a fulfillment line after the invoice service sends this request. If your deployment uses a source system, then this system provides most of the data that the billing system requires to complete billing. Order Management stores this information, and then routes the billing request to the billing system.

 **Note:** A change order from Order Management cannot update an invoice. A change to an invoice is typically a credit from a return order or a prepayment. It is not typically a cancellation.

The Invoice service sends the following information to the billing system. The Description column describes the details that the Invoice service sends to the billing system.

Information	Description
<p>Fulfillment line details from a fulfillment system</p>	<p>The invoice service uses the Create Billing Lines request to send billing information from the sales order or the return order each time the fulfillment line is eligible for billing. For example, it sends unit list price, unit selling price, discounts, charges, tax attributes, sales credits, and fulfillment details.</p> <p>Note the following:</p> <ul style="list-style-type: none"> <li>• The invoice service sends charges that occur in the order header with the first fulfillment line that Order Management fulfills for an order.</li> <li>• If the fulfillment line does not include payment information or sales credits, then the invoice service sends this information from the order header.</li> <li>• The invoice service sends prepayment information from the order header for every fulfillment line.</li> </ul>

Information	Description
	<ul style="list-style-type: none"> <li>Order Management does not support discounts that occur in the order header.</li> </ul>
Return lines	The invoice service sends the reference to the original sales order line, the return reason, received quantity, and delivered quantity.
Shipment set or configured item	The invoice service sends the fulfillment lines that the shipment set or the configured item contains together. If only some lines in the shipment set or configured item are fulfilled, then it sends only the fulfilled lines.

The billing system processes the information that it receives, and then sends the following information to the Invoice service:

- Invoice information
- Credit memo information
- Billing amount
- Billing date
- Invoice date or credit memo date
- Number
- Status
- Legal entity information

It sends one of the following statuses:

- Await Billing
- Billed

## Return Service

The Return service sends a request to a receiving system and interprets the reply and update that it receives from this system. Note the following:

- The Return service creates a change receipt advice or a cancel receipt advice when Order Management receives a return request.
- The request might include one or more attribute updates, such as to increase the receipt quantity.
- If Order Management receives a request to change the original copy of the sales order that the customer returned, then the return service sends a request to the fulfillment system that creates this receipt advice.
- If Order Management receives a request to cancel the original copy of the sales order line that the customer returned, then the task layer service cancels the receipt advice. A cancel is typically allowed up until the fulfillment system receives the returned items.
- If the ordered quantity is greater than the delivered quantity on the receipt advice, and if the customer does not require the ordered quantity, then the Return service can request to cancel the remaining quantity.

### How the Return Service Handles Partial Returns

The Return service can process a partial receipt, such as the return of only some items of a configured item.

If the customer returns only part of the return, then the Return service splits the fulfillment line into the following lines:

- One line that includes a status of Delivered for the items that the customer returned
- One line that includes the items that the customer did not returned

If the customer returns only part of the return, and if the return includes a configured item or kit, then the Return service splits the fulfillment line into the following orchestration groups:

- One group that includes the fulfillment lines that the customer returned
- One group that includes the fulfillment lines that the customer did not return or that are not returnable

### How the Return Service Handles Events

The Return service might receive a status update from the receiving system that starts an event. For example, if the receiving desk receives goods, then the Return service might receive a status update that starts an event that creates a receipt advice. The following events in the receiving system might start a status update in Order Management:

- Receive goods on the receiving deck when the receipt is created.
- Deliver goods into inventory.
- Return goods to a customer.
- Correct an order after a receipt transaction occurs. For example, a customer cannot return a deliver transaction.

## Fulfillment Order Service

The Fulfillment Order service sends a request to an enterprise resource planning (ERP) system. It provides the following features.

Feature	Description
Send a fulfillment request to an enterprise resource planning system and receive a status update.	<p>The Fulfillment Order service can send a request that does the following modification on a sale order that resides in Order Management, and that the enterprise resource planning system uses:</p> <ul style="list-style-type: none"> <li>• Create.</li> <li>• Update.</li> <li>• Place a hold.</li> <li>• Release a hold.</li> <li>• Update the status.</li> <li>• Cancel.</li> </ul>
Aggregate the requests that it sends to the enterprise resource planning system to help minimize problems that might occur with the timing of the requests.	<p>The Fulfillment Order service also does the following:</p> <ul style="list-style-type: none"> <li>• Sends an update to the enterprise resource planning system each time Order Management accepts a change order that affects sales order fulfillment.</li> <li>• Receives interim and final status updates from the enterprise resource planning system. Replies are not delivered immediately. They are delivered when the fulfillment activities run.</li> </ul> <hr/> <p>An aggregator sends the requests that it has aggregated as a single request when a time limit expires or when it has aggregated a number of fulfillment lines for a sales order. If a sales order includes multiple fulfillment lines, and if these lines complete the task before the timeout occurs, then it sends all requests when the task completes. Note the following:</p> <ul style="list-style-type: none"> <li>• The aggregator can aggregate only one time for each system for each sales order. If Order Management receives more fulfillment lines for a sales order after the aggregator sends a request, then it sends each of these lines individually.</li> <li>• The default timeout is five minutes. You can create a business rule that customizes the timeout in minutes.</li> </ul>

Feature	Description
	<p>You can use the following parameters when you use the aggregator with an external interface routing rule. If you specify both parameters, then the fulfillment order service calls the external system when the rule meets either condition:</p> <ul style="list-style-type: none"><li>• <b>AGGREGATOR_MAX_LINES.</b> Specify the number of pending requests that must aggregate before calling the external fulfillment system.</li><li>• <b>AGGREGATOR_WAIT_TIME.</b> Specify the amount of time to wait before calling the external fulfillment system. This time starts when the aggregator receives the first request.</li></ul>

### Related Topics

- [Architectural Layers in Order Management: How They Work](#)

## Creating Custom Task Types: Explained

The template task layer is a web service wrapper that you can use to create a custom task type while maintaining data integrity in Order Management Cloud.

You can use the template task layer to do the following work:

- Define a custom fulfillment task type, fulfillment task, or service that an orchestration process can call.
- Validate that the service data object includes data for each required attribute.
- Add processing logic to the outbound request or to the inbound reply for an action that comes predefined with the template task layer service. You can add logic that defaults data onto the outbound request or that validates the data. For the reply, you can add logic that interprets attributes or messages that the fulfillment system sends that might require split processing.
- Determine the transaction data to update as a result of the external service call. This data resides in the transaction tables that Order Management uses.

Maintaining data integrity makes sure that the task type you create displays correctly throughout the Order Management work area. Using the template task layer also makes sure that the following functionality works correctly:

- Status update
- Wait step
- Forward planning
- Jeopardy
- Hold processing
- Split processing
- Change management
- Error recovery

A custom service that you create with the template task layer works in the same way that a predefined task layer service works. For details about how to implement the template task layer, see Article ID 1536633.1 on My Oracle Support at <https://support.oracle.com>.

## Creating Custom Task Types: Procedure

This topic describes the required and optional setup work that you must do to create a custom task type. For details about using task types, see Article ID 1536633.1 on My Oracle Support at <https://support.oracle.com>.

Create a custom task type:

1. Add a custom task type:
  - o In the Navigator, click **Setup and Maintenance**.
  - o On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
  - o On the Setup page, search for, and then open Manage Task Types.
  - o On the Manage Task Types page, click **Actions**, and then click **Add Custom**.

Order Management Cloud adds the following services in the Services list:

- o One service that references the outbound Create operation code
- o One service that references the Inbound operation code

You can specify a name for each service, and you can add a service that references some other operation code, such as Change, Get Status, Apply Hold, Release Hold, or Cancel. You must create at least one task for each new task type.

2. Assign a status code to the task type.

Order Management sets a default value for some system status codes, such as Pending, Change Pending, Cancel, or Canceled. The status code that each task type references also controls the values for the exit criteria for a wait step that uses this task type, and the value of the task status in the next orchestration process step. You can create a new status code, or you can assign an existing status code to the custom task type.

3. Connect Order Management to the fulfillment system that will perform the tasks and services that the new task type references. For details, see [Connecting Order Management to Fulfillment Systems: Procedure](#).
4. Reference the custom task type when you define the orchestration process.

You can reference your custom task type in the same way that you reference a predefined task type. Note that Order Management can use these services only in an orchestration process branch that allows split services.

For details about exit criteria, branches, wait steps, and using a task type in an orchestration process, see the [Customizing Orchestration Processes](#) section in the guide titled [Oracle SCM Cloud, Implementing Order Management](#).

### Optional Setup for Custom Task Types

The following setups are optional.

Type of Setup	Description
Preprocessing service	<p>Add preprocessing logic to the actions that come predefined with the template task layer service. For example:</p> <ul style="list-style-type: none"> <li>• Default data onto the outbound request.</li> </ul>

Type of Setup	Description
	<ul style="list-style-type: none"> <li>Validate data on the outbound request.</li> </ul>
Postprocessing service	<p>Add postprocessing logic to the actions that come predefined with the template task layer service. For example:</p> <ul style="list-style-type: none"> <li>Default data onto the inbound request.</li> <li>Validate data on the inbound request.</li> <li>Interpret attributes or messages that the fulfillment system returns that might require Order Management to split processing across multiple orchestration processes.</li> </ul>
Change management	<p>To use change management on an orchestration process step that references your custom task type, specify the attributes for the task type on the Manage Order Attributes That Identify Change page. Make sure your task type references the Update service and the Cancel service, and the connectors that these services require.</p>
Hold code	<p>To apply a hold to your custom service, create hold codes for it. Hold All applies to the new services and to the existing services.</p>
Jeopardy threshold	<p>To include a jeopardy score for your custom task, define a jeopardy threshold for it.</p>
Processing constraint	<p>Define a processing constraint that controls when to use your custom task. For example, you can use a processing constraint that specifies the attributes that are required in the outbound request or on the inbound reply.</p>
Data set used as part of outbound request	<p>The template task layer uses a complete data set to communicate Order Management attributes between the following items:</p> <ul style="list-style-type: none"> <li>GetValidFLineData</li> <li>The preprocessing service</li> <li>Routing rules for the external interface layer</li> </ul> <p>You can reduce the data set to make processing more efficient.</p>
Register error messages	<p>If your external fulfillment system sends error messages to Order Management that you prefer to process and display in the messaging framework that Order Management uses, then you must register these messages.</p>

### Related Topics

- Connecting Order Management to Fulfillment Systems: Procedure
- Defining Orchestration Processes: Explained

## Customizing Fields

## Configuring Extensible Flexfields in Order Management: Explained

You can use an extensible flexfield to gather, organize, and display data that the predefined data structure does not contain. It is similar to a descriptive flexfield in that it is an expandable data field that contains segments, where the application database represents each segment as a single column.

Note the following:

- An extensible flexfield supports a one-to-many relationship between one entity and many extended attribute rows. You can use it to add multiple segments that are context sensitive.
- You can set up an extensible flexfield for a fulfillment line, or on some other object that supports an extensible flexfield.
- You use an extensible flexfield primarily when you must separate attributes according to task layer, or when you must capture multiple contexts and group them according to function.

### Example Usages of Extensible Flexfields in Order Management

Consider the following example usages of extensible flexfields in Order Management Cloud.

Example Usage	Description
Get information from an order capture system.	A source order that Order Management receives contains a set of attributes. If your organization requires more information or more attributes, then you can use an extensible flexfield to get it, and then use this information during order fulfillment.
Send information to a fulfillment system.	Order Management uses a fulfillment request object to send a fulfillment request that includes a predefined set of attributes to a fulfillment system. You can use an extensible flexfield to send information that these attributes do not include, but that the fulfillment system requires to complete the fulfillment request.
Receive information from a fulfillment system.	A fulfillment system might send attributes that provide some business value, and that a user must view in Order Management or in an order capture system. In some situations, Order Management might also use this information in the next set of tasks. You can use an extensible flexfield to receive these attributes.
Write a business rule.	You can use an extensible flexfield attribute to write a business rule for the following items: <ul style="list-style-type: none"> <li>• Pretransformation rule</li> <li>• Transformation rule</li> <li>• Posttransformation rule</li> <li>• Process assignment rule</li> <li>• External interface routing rule</li> </ul>
Manage changes.	You can configure Order Management to use an extensible flexfield as an order attribute that indicates that a change occurred.
Display attributes in the Order Management work area.	The Order Management work area displays the following extensible flexfields: <ul style="list-style-type: none"> <li>• Headers</li> <li>• Order lines</li> <li>• Fulfillment lines</li> <li>• Fulfillment line details</li> </ul>

Example Usage	Description
	<ul style="list-style-type: none"> <li>• Activities</li> </ul>
<p>A user can read but not edit an extensible flexfield attribute in the Order Management work area.</p>	

## Objects That Support Extensible Flexfields

You can use an extensible flexfield for the following objects on a sales order:

- Header
- Order line
- Fulfillment line
- Fulfillment line detail
- Price adjustment
- Sales credit
- Payment
- Lot serial
- Activity

## Deploying Extensible Flexfields in Order Management: Procedure

This topic describes how to publish and deploy extensible flexfields in Order Management Cloud. For details about how to configure Order Management to use an extensible flexfield with a source system, how to use an extensible flexfield in a business rule, or how to use an extensible flexfield with a fulfillment system, see the paper that describes extensible flexfield usage (Article ID 1536633.1) on My Oracle Support at <https://support.oracle.com>.

Publish and deploy extensible flexfields in Order Management:

**1.** Publish the predefined extensible flexfields:

- In the Navigator, click **Scheduled Processes**.
- On the Scheduled Processes page, click **Schedule New Process**, and then run the Publish Extensible Flexfield Attributes scheduled process.

This scheduled process publishes the extensible flexfields into Oracle Business Rules in the Oracle Metadata Services Repository, and to the processing constraints framework in Order Management. You must use the DOO\_ORDER\_ADMINISTRATOR job role to run this scheduled process.

- In the Navigator, click **Setup and Maintenance**.
- On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
- On the Setup page, search for, and then open Manage Order Extensible Flexfields.
- Verify that Order Management successfully published the extensible flexfields.

You can publish only the following extensible flexfields. The Category Hierarchy and Preconfigured Context Values are not available for these extensible flexfields.

Entity	Category
Headers	DOO_HEADERS_ADD_INFO
Lines	DOO_LINES_ADD_INFO
FulfillLines	DOO_FULFILL_LINES_ADD_INFO
Payments	DOO_PAYMENTS_ADD_INFO
PriceAdjustments	DOO_PRICE_ADJUSTMENTS_ADD_INFO
SalesCredits	DOO_SALES_CREDITS_ADD_INFO
OrchestrationTaskActivity	DOO_ACTIVITIES_ADD_INFO
FulfillLineDetails	DOO_FULFILL_LINE_DTLS_ADD_INFO
LotSerialNumber	DOO_LOT_SERIAL_NUM_ADD_INFO

2. Navigate to the Edit Extensible Flexfield page, and then do the following work:
  - o Define contexts. A context determines the attributes to display according to the criteria that you specify. A context is part of a hierarchy of attributes, where the entity and entity category represent the top level of the hierarchy, contexts represent the lower levels, and attributes represent the lowest level.
  - o Use the Pages tab of the Categories: Details area to define pages and to associate contexts to Order Management pages.
3. Use the Manage Order Extensible Flexfields page to deploy each of your custom extensible flexfields.
4. Repeat step 1 to publish your custom extensible flexfields.



# 7 Customizing Orchestration Processes

## Defining Orchestration Processes

### Defining Orchestration Processes: Explained

You can define a custom orchestration process so that it meets the requirements of your organization. You specify the rules that determine how Order Management Cloud creates the orchestration process at run-time, and then assigns objects to the fulfillment lines that Order Management creates as part of this order processing.

You can use the predefined ShipOrderGenericProcess or ReturnOrderGenericProcess orchestration processes. You must create them and then deploy them. It is not necessary to release them.

### Specifying How to Plan and Replan Orchestration Processes

You can specify how Order Management plans and replans each orchestration process. To specify planning and replanning, see Defining Orchestration Process Steps: Points to Consider.

Type of Planning	Description
Process planning	<p>Sets and helps meet the completion date for each orchestration process step and task in an orchestration process.</p> <p>Process planning allows you to specify how to use the transformation rules that an orchestration process references to transform each source order, including planning for each step after it receives a source order from an order capture system.</p>
Process replanning	<p>Updates completion dates when a change occurs to a sales order at any point in an orchestration process. You can configure replanning in the following ways:</p> <ul style="list-style-type: none"> <li>• Replan an orchestration process immediately after each orchestration process step finishes. For details, see Defining Orchestration Processes: Points to Consider.</li> <li>• Replan an orchestration process according to some event, such as every time it receives an update from the order capture system. To control when it does this replanning, you can use a scheduled process named Plan Orchestration Processes to schedule an update at regular intervals according to the frequency that your deployment requires. For example, if your orchestration process requires planning data that is current, then you can administer this scheduled process to run the orchestration process and update the planning data each day. To view an example that uses a scheduled process, see Using Scheduled Processes to Fix Errors in Multiple Sales Orders: Procedure.</li> </ul>

#### Related Topics

- [Fixing Errors in All Sales Orders: Procedure](#)

## Defining Orchestration Processes: Examples

An orchestration process provides a way for you to define a business process. It can include steps that call the task layer service, get planning details, modify change management settings, set status conditions, and so on. The following examples describe some of the ways that you can use an orchestration process to define a business process.

### Call the Task Layer Service

Assume you are an order administrator who works at a company that sells widgets. This process must incorporate the following company policy:

- If an invoice exceeds \$100,000, then a representative must phone the customer.

You must identify the steps that your company requires to fulfill a sales order, and then define an orchestration process that includes these steps so that the orchestration process mirrors your business process. To get started, create the following pseudocode that specifies the sequence of calls to the Task Layer service:

1. Plant Acknowledgment.
2. Assemble.
3. Wait for status to equal COMPLETE.
4. Ship.
5. Wait for status to equal SHIPPED.
6. Call Customer.
7. Wait for Call Customer to equal COMPLETE.
8. Test for the following conditions:

If the invoice exceeds \$100,000, then do the following work:

- Send High Value Invoice.
- Wait for the status for High Value Invoice to equal BILLED.

If the invoice does not exceed \$100,000, then do the following work:

- Invoice.
- Wait for the status to equal BILLED.

9. End the condition.

### Get Planning Details

Assume you are an order administrator who works at a company that sells carpet. Your company has established the following lead-times that allow the sales representative to monitor the fulfillment process and to determine when an order will be fulfilled:

Step	Lead Time
Schedule	Two days
Reservation	One day
Shipment	Six days

Step	Lead Time
Invoice	Two days

To define an orchestration process that includes this information, you add the default lead time to each orchestration process step. If a step is delayed, then a process runs in the background that replans the process and resets the expected completion dates.

## Set a Status

Assume you are an order administrator who works at a company that sells carpet. You have an important customer who requires that you notify the receiving clerk one day before shipping system ships the carpet. You define an orchestration process that handles the sales orders for this customer. You define an orchestration process class for this process named Carpet Process. You determine that this class must include the following statuses:

- SHIPPED
- RESERVED
- READY TO SHIP
- SHIPPED
- INVOICED

You use the Orchestration Process Status tab to define the status conditions. For example:

- If the status of the Create Shipment step is PRESHIP READY, then use the READY TO SHIP status to indicate the orchestration process status.

The order manager can use the Order Management work area to determine whether or not the orchestration process status is READY TO SHIP.

### Related Topics

- [Order Management Statuses: Explained](#)

## Defining Orchestration Processes: Points to Consider

This topic describes some important points to consider when you define an orchestration process.

- ▲ **Caution:** If you use your implementation project to migrate an orchestration process instance from a development environment to a production environment, then do not modify the process name in either environment. Modifying the name might prevent Order Management Cloud from updating references to other data in the orchestration process. For details about using an implementation project, see [Implementing Order-to-Cash: Points to Consider](#).

## Use Predefined Orchestration Processes

It is recommended that you use one of the following predefined orchestration processes before you define a custom orchestration process. Using a predefined orchestration process and other predefined objects will help to reduce your development and maintenance effort. If you find that these predefined orchestration processes do not meet your business requirements, then define a custom orchestration process.

Predefined Orchestration Process	Description
ShipOrderGenericProcess	<p>Includes the following tasks:</p> <ul style="list-style-type: none"> <li>• Schedule</li> <li>• Reservation</li> <li>• Shipment</li> <li>• Invoice</li> </ul>
ReturnOrderGenericProcess	<p>Includes the following tasks:</p> <ul style="list-style-type: none"> <li>• Return Receipt</li> <li>• Invoice</li> </ul>
OrderFulfillmentGenericProcess	<p>Includes the following tasks. You can use it for various fulfillment requirements, such as back-to-back shipments, drop shipments, and so on:</p> <ul style="list-style-type: none"> <li>• Schedule Conditional. Starts the branching.</li> <li>• Request Supply. Starts the back-to-back branch.</li> <li>• Pause.</li> <li>• Create Back to Back Shipment Request.</li> <li>• Wait for Back to Back Shipment Request.</li> <li>• Create Purchase Request. Starts the drop ship branch.</li> <li>• Wait for Procurement.</li> <li>• Create Reservation. Starts the warehouse shipment branch.</li> <li>• Create Shipment Request.</li> <li>• Wait for Shipment Advice.</li> <li>• Merge. Ends the branching.</li> <li>• Invoice.</li> <li>• Wait for Invoice.</li> </ul>

## Perform the Required Configurations

To define an orchestration process that accurately represents your desired business process, it is recommended that you configure the following items, and that you configure them in the same order that the following table lists them.

Item That Requires Configuration	Work That You Perform
Task type	Define the task types that arrange fulfillment tasks in groups. Each task type references services that communicate with a type of fulfillment system. For example, a billing system. For details, see <a href="#">Task Types: Explained</a> .
Orchestration process	Define the orchestration process that represents your business process that fulfills a fulfillment line. It includes the sequence of calls to the task layer service and planning details, change management settings, and status conditions. For details, see <a href="#">Defining Orchestration Processes: Procedure</a> .
Change logic	Define the set of rules that determine how the orchestration process handles changes to sales orders. For details, see <a href="#">Processing Changes That Occur During Order Fulfillment: How It Works</a> .
Process planning	Define the schedule that displays the completion date of each task and of the orchestration process. For details, see <a href="#">Order Management Process Planning: How It Works</a> .

Item That Requires Configuration	Work That You Perform
Jeopardy threshold and jeopardy priority	Define the level of risk that is associated with the delay of an orchestration process task as low, medium, or high. For details, see Jeopardy Threshold: Explained.
Statuses and status conditions	Define the status of a sales order or an object, including a task, orchestration process, fulfillment line, or order line. For details, see Orchestration Process Status: Explained.
Orchestration process deployment	You must deploy an orchestration process that you create or update. For details, see Deploying Orchestration Processes: Procedure

## Configure Replanning

You can enable the Replan Instantly feature when you define an orchestration process. It replans the orchestration process immediately after this process completes the orchestration process step, and then displays the revised order data according to the replanning work that it does. It is recommended that you use Replan Instantly only for high priority sales orders, or with sales orders that include a jeopardy threshold of less than one day. For performance reasons, it is recommended that you do not use it with an orchestration process step that is long or complex. If Replan Instantly does not contain a check mark, then Order Management updates the planning data only during the scheduled replanning.

For details about how to configure replanning for each orchestration process step, see Defining Orchestration Process Steps: Points to Consider.

## Defining Orchestration Process Steps: Points to Consider

An orchestration process includes the orchestration process steps that Order Management Cloud requires to process a fulfillment line. Each step includes information that Order Management uses when it runs the step, such as the task type, task, service, dependencies, planning, change management, and so on.

To define an orchestration process step, you access the Edit Orchestration Process Definition page, and then use the Step Definition list in the Process Details area.

## Specify the Orchestration Process Step

Consider the following fields that you can set when you specify an orchestration process step.

Field Name	Description
Step Type	<p>Specify the behavior of the orchestration process step. You can view these steps in an orchestration process diagram. You can set this field to one of the following values:</p> <ul style="list-style-type: none"> <li>• <b>Conditional.</b> Runs a path in an orchestration process according to the results of a condition. You must specify a branching condition on a step immediately after the conditional step.</li> <li>• <b>Merge.</b> Identifies the point where two or more orchestration paths reunite.</li> <li>• <b>Service.</b> Allows this step to reference a service.</li> <li>• <b>Subprocess.</b> Allows this step to reference an orchestration subprocess.</li> </ul>
Task Type	<p>Specify the task type. Each task type includes services that you can use to communicate with a type of fulfillment system, such as a billing system. You can use the following predefined task types:</p> <ul style="list-style-type: none"> <li>• Schedule</li> <li>• Reservation</li> </ul>

Field Name	Description
	<ul style="list-style-type: none"> <li>• Shipment</li> <li>• Invoice</li> <li>• Return</li> </ul>
Task	<p>Specify the task to run. A task can include more than one step. For example, the Ship task calls the Create Shipment service. It also calls the Wait for Shipment service to wait for different status values to occur in the fulfillment system updates.</p> <p><b>⚠ Caution:</b> If you use your implementation project to migrate an orchestration process instance from a development environment to a production environment, then do not modify the task name in either environment. Modifying this name might prevent Order Management from updating references to other data in the orchestration process. For details about your implementation project, see <i>Implementing Order-to-Cash: Points to Consider</i>.</p>
Service	Identifies the fulfillment task service that this step calls.
Manual	Specify whether or not to wait for user input. If you define a manual task, then the orchestration process waits until the user manually completes the task in the Order Management work area. It is recommended that you define some tasks as manual tasks. For example, you can define the Schedule task as a manual task so that an order manager can manually schedule all fulfillment lines at the end of the day.
Exit Criteria	<p>Specify the task status that determines when to exit a wait step. For example, if the status of a shipment task changes to Shipped, then exit the wait step.</p> <p>If your task includes multiple wait steps, then make sure these steps do not include the same set of exit criteria.</p> <p>Use the Manage Task Status Conditions page to make sure the last or only wait step uses the exit criterion.</p>
Line Selection Criteria	Specify an expression that filters the criteria when the orchestration process does not require all of the order lines or the fulfillment lines when it calls the fulfillment task service that you specify for the step.

## Specify Branching

Consider the following fields that you can set when you specify an orchestration process that includes branching.

Field Name	Description
Branching Condition	Specify the criteria that must be met to run the steps in a branch.
Evaluation Sequence	Specify the sequence that the orchestration process uses when it evaluates branch conditions.
Otherwise	Specify whether or not this step starts the branch that does not meet the branching condition.

## Specify Orchestration Planning

Consider the following fields that you can set when you specify orchestration planning for an orchestration process step.

Field Name	Description
Planning Default Branch	Specify the default path that Order Management uses for planning. Order Management uses this setting only if the orchestration process includes one or more conditional branches.
Fulfillment Completion Step	Add a check mark to this field to indicate that the fulfillment lines are fulfilled when this step finishes. Order Management uses this setting when it plans an orchestration process to make sure it meets the request date. The last step that occurs in chronological order in the orchestration process is not necessarily the Fulfillment Completion Step. For example, an orchestration process might use the requested ship date to indicate the completion date as the last step instead of using the shipped date to indicate the completion date.
Default Lead Time and Lead Time UOM	Specify the duration that Order Management requires to complete a unit of work. If you do not define a lead-time expression for a step, then it uses the default lead time.
Lead-Time Expression	Specify to use Oracle Business Rules to define the lead time. This feature allows you to define a complex lead-time expression.

## Specify Change Management

Consider the following fields that you can set when you specify change management for an orchestration process step.

Field Name	Description
Hold on wait	Sends a message to the fulfillment system for each active step when the orchestration process receives a change order.
Use Transactional Item Attributes	If you enable transactional item attributes, then Order Management examines the transactional item attributes to help it determine the differences that exists between the change order and the previous version of this order.
Use Flexfield Attributes	If you enable flexfield attributes, then Order Management examines them to help it determine the differences that exists between the change order and the previous version of this order.
Compensation Pattern	<p>Specify the set of rules that determine how to handle each step that ran before Order Management received a change order.</p> <p>For example, assume a change order requests a change from carpet to tiles. In this example, Order Management must cancel a number of the previous steps that it ran. If it already scheduled carpet for shipping, then Order Management must cancel the steps that scheduled the shipping. Order Management creates a different set of fulfillment lines for tiles when it processes the change order, so it must use a different orchestration process because a tile order requires more time to fulfill and it uses a different contractor. Order Management must cancel most of the previous steps, but it does not cancel the Measure step because the room dimensions are still accurate.</p> <p>If you do not specify a compensation pattern, then Order Management might process a step as an update, depending on the context of the operation. It might rerun some steps. If Order Management cannot compensate a step for some reason, then it bypasses the step, and then compensates the next step that it encounters.</p>

### Related Topics

- [Implementing Order-to-Cash: Points to Consider](#)

## Defining Orchestration Processes: Procedure

This topic describes how to define an orchestration process. It uses an example that calls a scheduling service.

### Summary of the Work

To define an orchestration process, do the following work:

1. Prepare to define the orchestration process.
2. Define the orchestration process.
3. Define the orchestration process steps.
4. Deploy the orchestration process. For details, see *Deploying Orchestration Processes: Procedure*.

This topic includes example values. You might use different values, depending on your business requirements.

### Prepare to Define the Orchestration Process

Depending on the complexity of your orchestration process, you might need to perform one or more of the following steps:

1. Do the tasks in the Setup and Maintenance work area that allow you to define an orchestration process.
2. Define the task types that the orchestration process requires. Also define the tasks and services that these task types reference.
3. Define the status codes that the orchestration process requires and specify how the task types, fulfillment lines, and your orchestration process will use them.
4. Define the orchestration subprocesses that the orchestration process will reference.
5. Define the status catalogs that the orchestration process will use for status conditions. Define these catalogs in one of the following items:
  - o Oracle Product Model
  - o Oracle Product and Catalog Management
  - o Oracle Product Hub

### Define the Orchestration Process

Define the orchestration process:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, search for, and then open *Manage Orchestration Process Definitions*.
4. On the *Manage Orchestration Process Definitions* page, click **Actions**, and then click **Create**.
5. On the *Create Orchestration Process Definition* page, set the following fields, and then click **Save**.

Field	Description
Process Name	Enter an orchestration process name that does not include any spaces, such as <code>OrchestrationProcess1</code> .
Process Display Name	Enter any value. Do not include spaces.
Process Class	Select an orchestration process class that includes the statuses that you require for the orchestration process. For this example, select <b>Ship Order Class</b> .

Field	Description
Set	<p>Select a set that allows the business unit to use this orchestration process. For this example, select <b>Common Set</b>.</p> <p>A set is a collection of business units. It provides a way to organize business units and control the business units that can access an orchestration process. The predefined Common Set allows you to provide access to many business units. You must add your business unit to the set that you specify.</p>

## Define the Orchestration Process Steps

Define the orchestration process steps:

1. On the Step Definition tab, click **Actions**, and then click **Add Row**.
2. Set the following fields, and then click **Save**.

Field	Description
Step Name	Enter text that describes the purpose of the step. For this example, enter Schedule Product.
Step Type	Select a value that indicates the type of processing that this step performs. For example, if this step must do conditional branching, then choose Conditional. This example calls a service, so select <b>Service</b> .
Task Type	Select the group of services that Order Management Cloud uses to perform a fulfillment task. For this example, select <b>Schedule</b> .
Task	Select the task that this step must perform. For this example, select <b>Schedule</b> .
Service	Identify the service that this step must call. For this example, select <b>Create Scheduling</b> .

3. Repeat steps 1 and 2, using the following values.

 **Tip:** To maintain the correct sequence when you add each step in this procedure, click the step that you most recently added, and then click Add Row. To make sure you do not lose any work, click Save after you add each step.

Field	Value
Step Name	Reserve Product
Step Type	Service
Task Type	Reservation
Task	Reserve
Service	Create Inventory Reservation

Field	Value
-------	-------

4. Repeat steps 1 and 2, using the following values.

Field	Value
Step Name	Ship Product
Step Type	Service
Task Type	Shipment
Task	Ship
Service	Create Shipping

5. Repeat steps 1 and 2, using the following values. Note that you also specify the exit criteria in this step.

Field	Value
Step Name	Wait for Product Shipment
Step Type	Service
Task Type	Shipment
Task	Ship
Service	Wait for Shipment
Exit Criteria	Shipped

6. Repeat steps 1 and 2, using the following values.

Field	Value
Step Name	Create Invoice
Step Type	Service
Task Type	Invoice
Task	Invoice
Service	Create Billing Lines

- Repeat steps 1 and 2, using the following values.

Field	Value
Step Name	Wait for Invoice
Step Type	Service
Task Type	Invoice
Task	Invoice
Service	Wait for Billing

## Deploying Orchestration Processes: Procedure

You must release, and then deploy an orchestration process that you define so that it is available throughout Order Management Cloud. You deploy it on an instance of Order Management.

If you use the ShipOrderGenericProcess or the ReturnOrderGenericProcess predefined orchestration process, then you must deploy it, but it is not necessary to release it.

### Summary of the Work

To deploy an orchestration process that you have defined, do the following work:

- Release the orchestration process.
- Deploy the orchestration process.

### Release the Orchestration Process

Release the orchestration process:

- In the Navigator, click **Setup and Maintenance**.
- On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
- On the Setup page, search for, and then open Manage Orchestration Process Definitions.
- On the Manage Orchestration Process Definitions page, locate the orchestration process definition that you must release, click **Actions**, and then click **Edit**.
- On the Edit Orchestration Process Definition page, click **Actions**, and then click **Release**.

Order Management does the following work when you release an orchestration process:

- Validates the orchestration process logic to make sure that it correctly created this process. If it:
  - Identifies any errors during validation, then it stops the release, and then displays an error icon next to the orchestration process name. It keeps these errors until the next time it runs the validation. You must resolve these errors before you can continue the release.
  - Does not identify any errors during validation, then it continues the release. An orchestration process that does not contain any errors is valid. A valid orchestration process might include warning messages, but these messages do not stop the release from proceeding.

2. Finishes the validation.
3. Updates the status of the orchestration process to Released.
4. Makes the orchestration process definition read-only.
5. Creates, and then saves the BPEL artifacts that Order Management uses to deploy and run the orchestration process in a production environment.

You can release multiple versions of an orchestration process in a single day. You must reject the previous version before you can release the next version on the same day.

## Deploy the Orchestration Process

Deploy the orchestration process:

1. Click **Actions**, and then click **Deploy Process**.
2. In the Deploy Process dialog, enter your password, and then click **Deploy**.

If you cannot access the Manage Orchestration Process Definition pages to deploy an orchestration process for some reason, then you can use Oracle Middleware to deploy it. For details, see [Using Oracle Middleware to Deploy Orchestration Processes: Procedure](#).

## Using Oracle Middleware to Deploy Orchestration Processes: Procedure

If you cannot use the Manage Orchestration Process Definition page to deploy an orchestration process for some reason, then you can use Oracle Middleware to deploy it.

### Summary of the Work

Use Oracle Middleware to deploy an orchestration process that you have defined:

1. Release the orchestration process. For details, including important guidelines, see [Deploying Orchestration Processes: Procedure](#).
2. Download the orchestration process.
3. Modify the SOA configuration plan.
4. Deploy the JAR file.

## Download the Orchestration Process

You deploy the downloaded artifacts to the server. You can use Oracle Setup Manager to export the artifacts, and Oracle Middleware to deploy them.

Download the orchestration process:

1. On the Manage Orchestration Process Definitions page, select the orchestration process that you must deploy, and then click **Edit**.
2. In the Download Generated Process dialog, click **Download**.
3. Save the archive file that Order Management Cloud displays to a local directory.
4. Open the archive file that you saved.

## Modify the SOA Configuration Plan

You can use an SOA (Service Oriented Architecture) configuration plan to define the URL and properties to use for different environments. Order Management uses this plan to search the SOA project for values that it must replace so that the project supports each environment.

Modify the following SOA configuration plan. Replace each host name with the ADF server and replace each port with the port that your organization uses for Order Management and the server and port that Supply Chain Management uses for SOA. Use the external-facing URLs of the servers:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAConfigPlan
xmlns:jca="http://platform.integration.oracle/blocks/adapter/fw/metadata"
xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
xmlns:orawsp="http://schemas.oracle.com/ws/2006/01/policy"
xmlns="http://schemas.oracle.com/soa/configplan">
<composite name="*">
<import>
<searchReplace>
<search/>
<replace/>
</searchReplace>
</import>
<service name="client">
<binding type="ws">
<attribute name="port">
</attribute>
</binding>
</service>
<reference name="*">
<binding type="ws">
<attribute name="location">
<searchReplace>
<search>http://localhost_am:port</search>
<replace>http://actualDOOADFserver:port</replace>
</searchReplace>
<searchReplace>
<search>http://localhost_soa:port</search>
<replace>http://actualDOOSOAServer:port</replace>
</searchReplace>
</attribute>
</binding>
</reference>
</composite>
</SOAConfigPlan>
```

## Deploy the JAR File

Deploy the JAR file. You can use one of the following tools:

- Oracle Enterprise Manager Middleware Control
- Ant command line tool
- Oracle WebLogic Scripting Tool

The JAR file resides in a Deploy folder that resides in a folder that uses the name of the orchestration process that you downloaded.

For information about how to deploy an SOA composite application, see Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suite.

## Lead-Time: Explained

Order Management Cloud uses lead-time to plan an orchestration process and to predict the completion date for this process. If actual completion dates are available, then the orchestration process uses actual dates instead of estimated

dates. The Gantt chart in the Order Management work area displays the planned orchestration process. Order Management uses the number of days that are past the lead time when it calculates jeopardy.

### Related Topics

- [Jeopardy Score: Explained](#)

## Defining Lead-Times for Orchestration Process Steps: Procedure

This topic describes how to define a rule that sets the lead-time for an orchestration process step according to a set of conditions.

Assume the order administrator of a flooring company requires an orchestration process that handles carpet orders. The lead-time that is required to ship the carpet if the inventory organization:

- Resides in Denver, Colorado, is two days
- Does not reside in Denver, Colorado, is four days

### Summary of the Work

To define the lead-time for an orchestration process step, do the following work:

1. Create the If statement for the first rule.
2. Create the Then statement for the first rule.
3. Create the If statement for the second rule.
4. Create the Then statement for the second rule.

This topic includes example values. You might use different values, depending on your business requirements.

For a demonstration that creates a business rule, see [Creating Business Rules: Demo](#).

### Create the If Statement for the First Rule

Create the If statement for the first rule:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, search for, and then open Manage Orchestration Process Definitions.
4. On the Manage Orchestration Process Definitions page, locate the CallCustomerWhenLargeInvoice orchestration process, click **Actions**, and then click **Edit**.  
For this example, you edit an orchestration process named CallCustomerWhenLargeInvoice. For details about how to create this process, see [Branching Orchestration Processes: Procedure](#).
5. On the Edit Orchestration Process Definitions page, in the Process Details area, in the Step Definition list, click **Planning**.
6. In the Ship Product row, set the following value.

Field	Value
Lead-Time UOM	Days

7. In the Lead-Time Expression column, click **Click for Rule**.
8. In the Lead-Time Expression dialog, click **Add Rule**, and then click **Expand**.

If you write a rule for an orchestration process that includes multiple fulfillment lines, then it is recommended that you use Advanced Mode. However, in this example, all lines are treated the same way, so Advanced Mode is not required. To use Advanced Mode, click **Show Advanced Settings**, add a check mark to **Advanced Mode**, and then edit the rule.

9. In the If area, click **Left Value**, expand **DOOSeededOrchestrationRules**, expand **DOOFLine**, click **InventoryOrganizationId**, and then click **OK**.
10. In the Right Value field, enter **1234440**.

For this example, assume the inventory organization ID for Denver is 1234440.

## Create the Then Statement for the First Rule

Create the Then statement for the first rule:

1. In the Then area, click **Add Action**, and then click **Assert New**.
2. Click **Select a Target**, and then click **DooSeededOrchestrationRules.Result**.
3. Click **Edit Properties**.
4. In the Edit Properties dialog, in the ResultObjKey row, enter the following value, and then click **OK**.

Field	Value
Value	BigDecimal. valueOf(2)

## Create the If Statement for the Second Rule

Create the If statement for the second rule:

1. In the Lead-Time Expression dialog, in the Lead-Time Expression Set area, click **Add Rule**.
2. In the lower portion of the dialog where the application added Rule 2, click **Expand**.
3. In the If area for Rule 2, click **Left Value**, expand **DOOSeededOrchestrationRules**, expand **DOOFLine**, select **InventoryOrganizationId**, and then click **OK**.

 **Tip:** In some fields, you can copy the value from one field to another instead of using the drop-down list. For example, you can copy `DooSeededOrchestrationRules.DOOFLine.inventoryOrganizationId` from the Left Value field of the first rule, and then paste it into the Left Value field of the second rule.

4. Click the **down arrow** for the operator, and then click **isn't**.
5. In the Right Value field, enter **1234440**.

## Create the Then Statement for the Second Rule

Create the Then statement for the second rule:

1. In the Then area for Rule 2, click **Add Action**, and then click **Assert New**.
2. Click **Select a Target**, and then click **DooSeededOrchestrationRules.Result**.
3. Click **Edit Properties**.
4. In the Edit Properties dialog, in the ResultObjKey row, enter the following value, and then click **OK**.

Field	Value
Value	BigDecimal. valueOf(4)

5. Click **Save**.
6. On the Edit Orchestration Process Definition page, click **Save**.

*Related Topics*

- [Creating Business Rules: Demo](#)

## Selecting Fulfillment Lines That Run for Orchestration Process Steps: Procedure

The example in this topic describes how to create a line selection rule that makes sure Order Management Cloud does not attempt to ship a nonshippable item.

Assume you work for a company that sells DVRs. The sales order includes multiple fulfillment lines for each of the following items:

- DVR
- Remote control
- Instruction manual
- Extended warranty

Your customers can purchase the extended warranty as a contract online, but it is not a shippable item, so Order Management must not send it to the fulfillment system during the Shipment task. To meet this requirement, you create a rule that includes the following statements when you define the SetUpShipment step.

Type of Statement	Description
If	Determines whether or not Order Management can ship the item.
Then	Selects the fulfillment line.

To fulfill a nonshippable item, you must create a separate step in the orchestration process that uses the template task layer to connect to a third-party system.

This example includes a simple rule that you can use with an orchestration process that includes only one fulfillment line. To write a rule that includes multiple lines, it is recommended that you use advanced rules. For details, see the topic that describes Usage of Oracle Business Rules in DOO Order Fulfillment (Article ID 1536633.1) on My Oracle Support at <https://support.oracle.com>.

### Summary of the Work

To select the fulfillment lines that run for an orchestration process step, do the following work:

1. Create the If statement.
2. Create the Then statement.

### Create the If Statement

Create the If statement:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.

3. On the Setup page, search for, and then open Manage Orchestration Process Definitions, and then open it.
4. On the Manage Orchestration Process Definitions page, locate the orchestration process named CallCustomerWhenLargeInvoice, click **Actions**, and then click **Edit**.  
For this example, you edit the CallCustomerWhenLargeInvoice orchestration process. For information about how to create this process, see Branching Orchestration Processes: Procedure.
5. On the Edit Orchestration Process Definitions page, in the Process Details area, in the Step Definition list, locate the Ship Product row.
6. In the Ship Product row, in the Line-Selection Criteria column, click **Click for Rule**.
7. In the Line-Selection Criteria dialog, click **Add Rule**, and then click **Expand**.
8. In the If area, click **Left Value**, expand **DOOSeededOrchestrationRules**, expand **DOOFLine**, click **ShippableFlag**, and then click **OK**.
9. In the Right Value field, enter "Y". You must include the double quotation marks.

--

## Create the Then Statement

Create the Then statement:

1. In the Then area, click **Add Action**, and then click **Assert New**.
2. Click **Select a Target**, and then click **DooSeededOrchestrationRules.Result**.
3. Click **Edit Properties**.
4. In the Properties dialog, in the ResultObjKey row, click **Value**.
5. In the Condition Browser, enter **DOOFLine**, wait a moment, expand **DooSeededOrchestrationRules.DOOFLine**, click **FulfillLineId**, click **OK**, and then click **OK** again.
6. In the Line-Selection Criteria dialog, click **Validate**, make sure the Validation Log that displays does not contain any errors, and then click **Save**.
7. On the Edit Orchestration Process Definition page, click **Save**.

# Defining Subprocesses In Orchestration Processes

## Orchestration Subprocesses: Explained

You can use an orchestration subprocess to increase the efficiency of your design. You can add a subprocess to an orchestration process, and then reuse this subprocess in another orchestration process. You can also use a subprocess to manage a large orchestration process. In general, it is easier to manage a large orchestration process that includes subprocesses than it is to manage a large orchestration process that does not include subprocesses.

### Example of a Subprocess

Assume your organization uses different steps when it processes an order, depending on the item. However, all orders use the following billing steps:

- Create invoice.
- Wait for invoice.
- Send request to activity system that notifies customer of payment receipt.
- Send request to activity system that creates coupons.

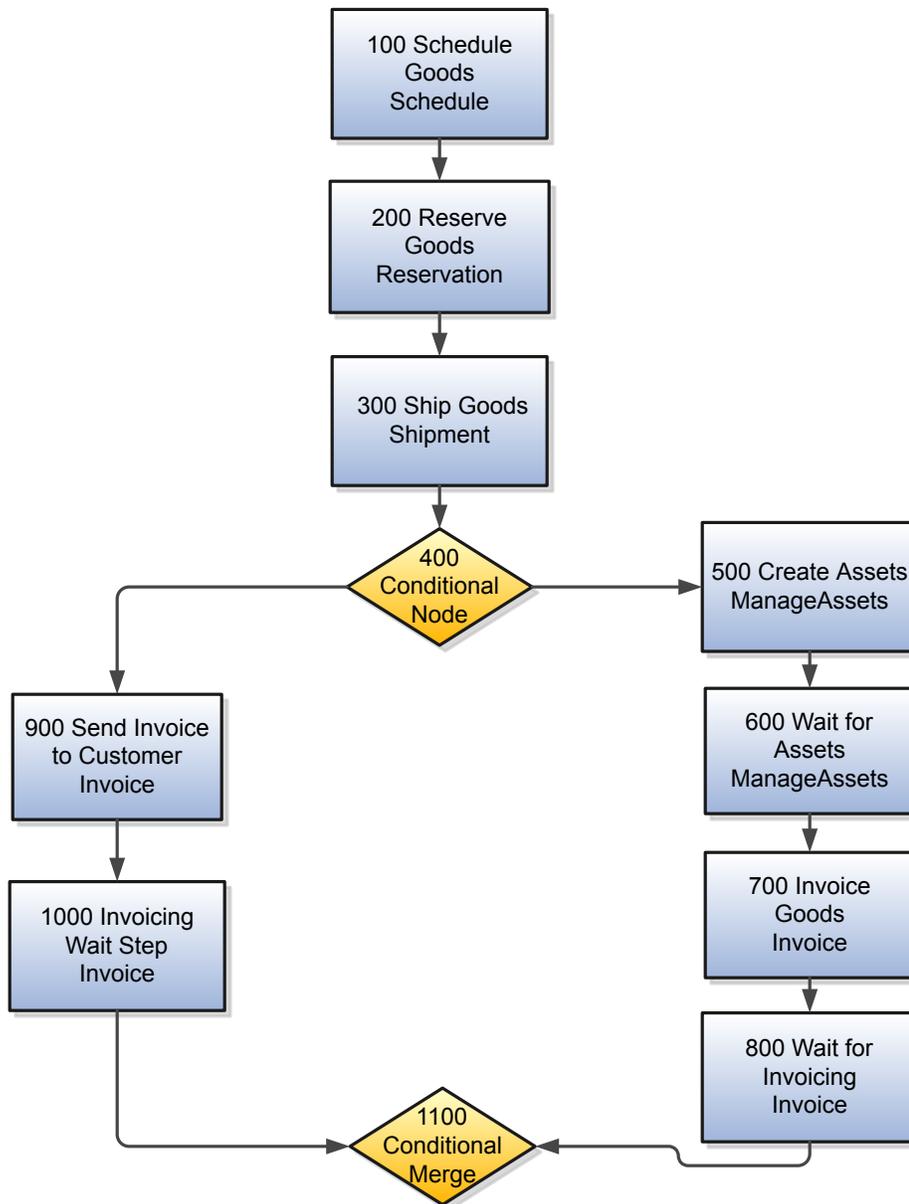
So, you create an orchestration subprocess that includes these steps, and then add this subprocess to an orchestration process that you define.

## Adding Branches to Orchestration Processes: Explained

You can add an orchestration process branch that includes a sequence of orchestration process steps that the orchestration process runs only if a condition is met.

For example, assume your company sells laptop computers. You define an orchestration process that allows a customer to purchase a laptop and then sends an invoice to this customer. You create a branch that runs only if this customer purchases a service agreement with the laptop computer. This branch creates an asset so that your company can track the computer as a durable good.

The following diagram illustrates the orchestration process flow for this example. Each step in this flow includes the step number, task name, task type, and service name. The Conditional Node indicates that an orchestration process is about to branch. The first step of the branch contains the condition.



If the condition is:

- **Met.** The orchestration process runs the steps on the branch that includes the Create Assets step.
- **Not met.** The orchestration process runs the steps on the branch that includes the Send Invoice step.

Order Management Cloud adds an empty default branch when it runs the orchestration process. If the orchestration process includes only one branch, then it is not necessary to set an Otherwise condition.

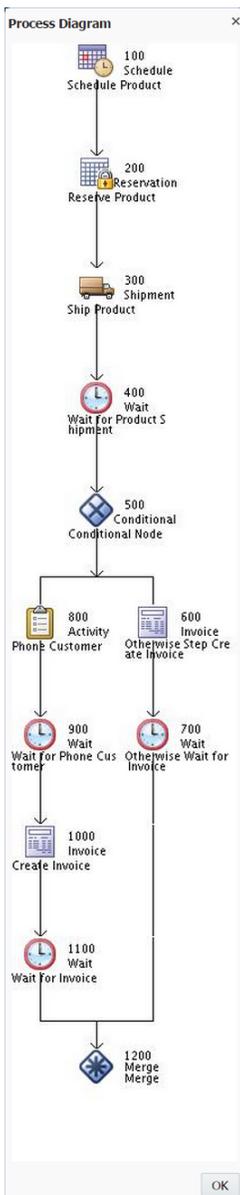
## Adding Branches to Orchestration Processes: Procedure

You can create a branching condition that determines whether or not to run an orchestration process branch.

For example, assume the order administrator of a flooring company must define an orchestration process that implements the following company policy:

- If a carpet order is valued at \$50,000.00 or more, then make sure a representative calls the customer before sending an invoice for this order.

You create the following orchestration process:



This example uses a rule that includes only one fulfillment line. To write a rule for an orchestration process that includes multiple fulfillment lines, it is recommended that you use advanced rules. For details, see the topic that describes Usage of Oracle Business Rules in DOO Order Fulfillment (Article ID 1536633.1) on My Oracle Support at <https://support.oracle.com>.

## Summary of the Work

Create a branch in an orchestration process:

1. Set up routing that notifies the representative.
2. Define the orchestration process.
3. Define the orchestration process steps.
4. Create the If statement.
5. Create the Then statement.
6. Test your work.

This topic includes example values. You might use different values, depending on your business requirements.

## Set Up Routing That Notifies the Representative

This example sends a notification to a representative to call the customer if the invoice is valued at \$50,000.00 or more. You must set up the routing that enables this notification. This setup is specific to this example. Other branching usages might not require this setup, or they might require some other setup.

Set up routing that notifies the representative:

1. Create the routing rules that enable the send. For details, see [Creating Routing Rules That Send Notifications: Procedure](#).
2. Create the connector that you referenced in the routing rules in step 1.
3. Use the [Manage Web Service Details](#) page to create the connector.

## Define the Orchestration Process

Define a new orchestration process. For this example, use the following values in the header of this orchestration process. For details about how to define an orchestration process and orchestration process steps, see [Defining Orchestration Processes: Procedure](#).

Field	Value
Process Name	CallCustomerWhenLargeInvoice
Process Display Name	Call the Customer
Process Class	Ship Order Class
Set	Common Set

## Define the Orchestration Process Steps

Define the orchestration process steps:

1. Add the following step.

Field	Value
Step Name	Schedule Product
Step Type	Service
Task Type	Schedule
Task	Schedule
Service	Create Scheduling

2. Add the following step.

 **Tip:** To maintain the correct sequence when you add each step in this procedure, click the step that you most recently added, and then click Add Row.

Field	Value
Step Name	Reserve Product
Step Type	Service
Task Type	Reservation
Task	Reserve
Service	Create Inventory Reservation

3. Add the following step.

Field	Value
Step Name	Ship Product
Step Type	Service
Task Type	Shipment
Task	Ship
Service	Create Shipping

4. Add the following step.

Field	Value
Step Name	Wait for Product Shipment
Step Type	Service
Task Type	Shipment
Task	Ship
Service	Wait for Shipment
Exit Criteria	Shipped
Fulfillment Completion Step	Contains a check mark.

5. Add the following step.

Field	Value
Step Name	Conditional Node
Step Type	Conditional

6. Add the following step.

Field	Value
Step Name	Phone Customer
Step Type	Service
Task Type	Activity
Task	Activity
Service	Create Activity
Evaluation Sequence	1

7. Add the following step.

Field	Value
Step Name	Wait for Phone Customer

Field	Value
Step Type	Service
Task Type	Activity
Task	Activity
Service	Wait for Activity
Exit Criteria	Completed

**8.** Add the following step.

Field	Value
Step Name	Create Invoice
Step Type	Service
Task Type	Invoice
Task	Invoice
Service	Create Billing Lines

**9.** Add the following step.

Field	Value
Step Name	Wait for Invoice
Step Type	Service
Task Type	Invoice
Task	Invoice
Service	Wait for Billing
Exit Criteria	Billed

**10.** In the Step Definition list, click the Conditional Node step, and then add the following step. This step creates the Otherwise branch.

Field	Value
Step Name	Otherwise Step Create Invoice
Step Type	Service
Task Type	Invoice
Task	<p>Otherwise Create Invoice</p> <p>Each task name that you associate with a task type must be unique. You associated the Invoice task with the Invoice task type earlier in this procedure, so you must create a new task name. To do this, In the Task field, click the Search down arrow, and then click <b>Create</b>. In the Create Task Name dialog, enter the following values, and then click <b>Save and Close</b>:</p> <ul style="list-style-type: none"> <li>○ Code: 1</li> <li>○ Name: Otherwise Create Invoice</li> <li>○ Display Name: Otherwise Create Invoice</li> <li>○ Task Type: Invoice</li> </ul>
Service	Create Billing Lines
Evaluation Sequence	2
Otherwise	Contains a check mark.
Planning Default Branch	Contains a check mark.

- 11.** In the Step Definition list, click the Otherwise Step Create Invoice step, and then add the following step. This step creates the Wait for Invoice Step on the Otherwise branch.

Field	Value
Step Name	Otherwise Wait for Invoice
Step Type	Service
Task Type	Invoice
Task	Otherwise Create Invoice
Service	Wait for Billing
Exit Criteria	Billed

- In the Step Definition list, click the Wait for Invoice step, and then add the following step. This step merges the branch back to the main flow.

Field	Value
Step Name	Merge
Step Type	Merge

## Create the If Statement

In this section you create a business rule. For a demonstration, see [Creating Business Rules: Demo](#).

Create the If statement:

- In the Step Definition list, in the Phone Customer step, in the Branching Condition column, click **Click for Rule**.
- In the Branching Condition Rules dialog, click **Add Rule**, and then click **Expand**.
- Delete the value **Rule 1**, and then enter **Condition for invoices valued at more than 50000 dollars**.
- Click **Left Value**.
- In the Condition Browser dialog, expand **DooSeededOrchestrationRules**, expand **DOOFLine**, click **extendedAmount**, and then click **OK**.
- Click **Is**, and then click **More Than**.
- Click **Right Value**.
- In the Condition Browser dialog, enter **50000**, and then click **OK**. Do not include commas.

## Create the Then Statement

Create the Then statement:

- In the Then area, click **Add Action**, and then click **Assert New**.
- Click **Select a Target**, and then click **DooSeededOrchestrationRules.Result**.
- Click **Edit Properties**.
- In the Properties dialog, in the ResultObj row, click **Value**.
- In the Condition Browser dialog, expand **DooSeededOrchestrationRules**, expand **Boolean**, click **True**, click **OK**, and then click **OK** again.
- In the Branching Condition Rules dialog, click **Save**.
- On the Edit Orchestration Process Definition page, click **Save**.

## Test Your Work

Test your work:

- Verify that you correctly defined the orchestration process steps and flow:
  - In the header of the Edit Orchestration Process Definition page, click **Actions**, and then click **Generate Process Diagram**.
  - Make sure the diagram that this page displays includes the same steps and logic that the diagram at the beginning of this topic displays.
- Test the nonbranching flow:
  - In the Navigator, click **Order Management**.
  - In the Order Management work area, create a sales order that is valued at less than \$50,000.00.
  - Verify that Order Management Cloud ships the product without requesting that the user call the customer.

3. Test the branching flow:
  - o In the Order Management work area, create a sales order that is valued at more than \$50,000.00.
  - o Verify that Order Management does not ship the product until the activity that the Wait for Phone Customer step references reaches a Completed state.

#### *Related Topics*

- [Creating Business Rules: Demo](#)

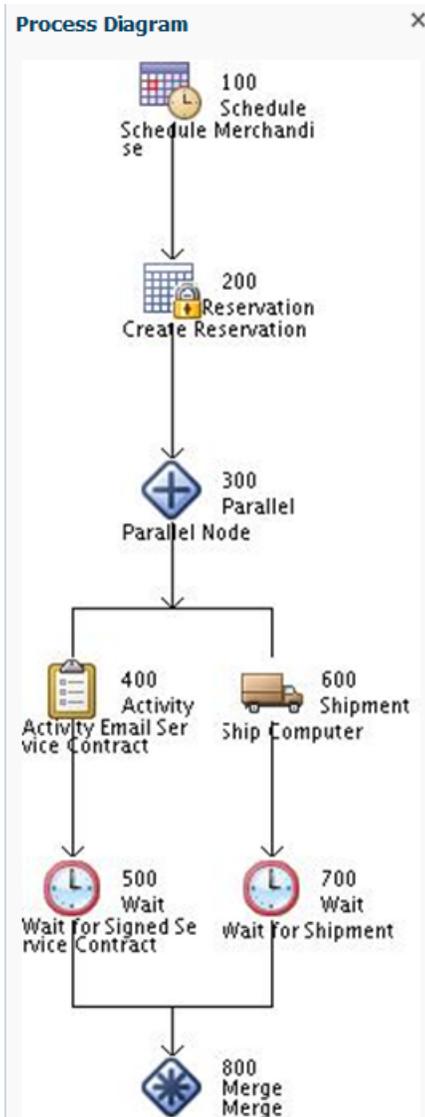
## Parallel Processing: Explained

Order Management Cloud can simultaneously process more than one task. You can use this parallel processing to process different sets of fulfillment lines that each require a different set of processing tasks.

For example, assume a customer purchases a laptop computer that includes a service contract, and that your business flow requires that the fulfillment system ship the laptop at the same time that it sends the service contract through e-mail. You can use parallel processing to ship the laptop and send the e-mail at the same time.

The following figure illustrates part of an orchestration process that includes branches that run in parallel with one another. Step 300 begins the parallel branching. Order Management runs steps 400 and 500 in one branch at the same time that it

runs steps 600 and 700 in a parallel branch. The Create Invoice and the Wait for Billing steps occur after step 800. This figure does not include these final steps.



Order Management plans the orchestration process after it finishes each task. However, if a task requires several days to finish, then planning data might not be current because Order Management cannot replan the branch while it waits for a long-running task to finish. For details about how to use a scheduled process named Plan Orchestration Processes to schedule an update that occurs at regular intervals, see *Defining Orchestration Processes: Explained*.

## Guidelines for Using Parallel Processing

You set up an orchestration process so that it uses parallel processing in the same way that you set up an orchestration process to use a conditional branch, with the following differences:

- When you create a branch, you select the Parallel step type.
- You do not create branching conditions, set an evaluation sequence, or select Otherwise.

Note the following:

- It is recommended that you use parallel processing when multiple, long-running tasks exist that must run at the same time. For example, to send simultaneous requests to two different fulfillment systems that typically require elapsed time to reply.
- It is recommended that you do not use parallel processing when the reply from a task request occurs almost immediately.
- You can set up parallel processing so that more than one task can run at the same time.
- You can filter the fulfillment lines of a group so that the tasks for some lines in this group run at the same time as tasks for the other lines.
- You can create a parallel process that does not filter lines in a branch so that more than one task is active for each line.
- Order Management does not allow you or a fulfillment system user to split a task that resides in a parallel branch.

## Assigning Orchestration Processes

### Assigning Orchestration Processes: Explained

Order Management Cloud creates a sales order, and then assigns an orchestration process to fulfillment lines according to assignment rules. Order Management creates assignment rules according to orchestration groups and order attributes.

You do not need to specify versions or effectivity dates in the assignment rules because the orchestration process controls these versions and dates. You must define orchestration processes or at least know the names that you will use for them before you create an assignment rule. You add the orchestration process name to bucket sets that allow you to select them when you create an assignment rule.

### Orchestration Groups

A fulfillment line resides in an orchestration group. Order Management uses the following predefined orchestration groups.

Orchestration Group	Description
Shipment Set	Order Management assigns the fulfillment lines that reside in a shipment set to the same orchestration process.
Model or Kit	Order Management assigns the fulfillment lines that reside in a configuration model or kit to the same orchestration process.
Standard	Use for standard items or finished items.

Note the following:

- You can assign an orchestration process according to each set of unique conditions.
- You can use Otherwise logic to set up a default orchestration process for each orchestration group.

## Assigning Orchestration Processes: Examples

The following scenarios describe examples of assigning an orchestration process.

Assign an Orchestration Process According to . . .	Description
Product. Each sales order for ceramic tile must use the same processing steps.	You write a process assignment rule that assigns an orchestration process named Tile Processing to each order line that includes tile as the product.
Customer. Customer A requires an inspection step for each sales order.	You write a process assignment rule that assigns an orchestration process named Customer A Process to each order line that includes Customer A in the order header.
Destination. Each sales order that the fulfillment system must ship to a country that resides outside of your current location requires different handling, such as completing customs forms.	You write a process assignment rule that assigns an orchestration process named International Orders to each order line that includes a foreign country in the ship-to address in the header.

## Assigning Orchestration Processes: Procedure

This topic describes how create a rule that assigns an orchestration process to a fulfillment line. For details about how to use a simplified rule builder, see [Using Visual Information Builder to Create Rules: Explained](#).

Assume you are an order administrator for a company that receives a large quantity of sales orders each day for laptop computers. You must create an assignment rule that assigns an orchestration process for each bulk order according to the ordered quantity. You create the following rule:

- If the ordered quantity is large, then assign the orchestration process named CallCustomerWhenLargeInvoice to the fulfillment line.

### Summary of the Work

Assign an orchestration process:

1. Add a rule to a decision table.
2. Add an assignment condition.
3. Add the buckets.
4. Add the bucket set to the assignment condition.
5. Configure the action to take.

This topic includes example values. You might use different values, depending on your business requirements.

### Add a Rule to a Decision Table

In this section you create a business rule. For a demonstration, see [Creating Business Rules: Demo](#).

Add a rule to a decision table:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, search for, and then open Manage Orchestration Process Assignment Rules.
4. On the Manage Orchestration Process Assignment Rules page, in the View list, click **AssignProcess**.

AssignProcess is a predefined decision table that you customize in this topic. It uses Advanced Mode, which allows you to do the following work:

- o Set a variable and an alias for each object.
- o Define a variable so that you can use a simple name to access it.
- o Define a hierarchy, which can improve performance.
- o Arrange fulfillment lines in a group.

AssignProcess comes predefined with the following aliases. An alias simplifies the text that Order Management displays. For example, the Group alias references the assignLaunchRules.ResponseGL object. So, you can specify Group instead of assignLaunchRules.ResponseGL in some fields when you define an assignment rule. You can also add custom aliases. Take a moment to examine the following predefined aliases, and then click **Collapse Pane** so that you can more easily view the decision table.

Alias	Object That the Alias References
Group	assignLaunchRules. ResponseGL
fline	assignLaunchRules. FulfillLineVO
Header	assignLaunchRules. HeaderVO

5. In the decision table, immediately above the Conditions section, click **Add**, and then click **Add Rule**.

Notice that the decision table includes a new rule in the R1 column.

## Add an Assignment Condition

In this section, you create the following part of the If statement:

- If ordered quantity is equal to quantity

Add an assignment condition:

1. In the decision table, immediately above the Conditions section, click **Add Rule**, and then click **Add Condition**.

Notice that the Condition Browser includes the Group, fline, and Header aliases that you noticed earlier.

2. In the Condition Browser dialog, expand **fline**, click **OrderedQty**, and then click **OK**.

Notice that the Conditions section of the decision table includes a new fline.OrderedQty row.

## Add Buckets

In this section, you create the buckets that you can select to complete the If statement. You create three buckets. Add the buckets:

1. In the decision table, in the Conditions area, click **fline.OrderedQty**.

When you click fline.OrderedQty, notice that the Local List of Ranges list and a pencil icon display immediately above the Condition table.

2. Click **Edit Bucket list** immediately to the right of Local List of Ranges.
3. In the Bucket set Editor, specify the following values for the bucket set.

Field	Value
Name	Bulk Ranges You can enter any text.
Description	Set of ranges to use for bulk orders.
Data Type	int

4. Create the small bucket. Click **Add Bucket**, and then set the following value in the row that contains an End Point value of 1.

Field	Value
Description	Small

5. Create the medium bucket. Click **Add Bucket**, and then set the following values in the row that contains an End Point value of 2. You modify this End Point value to 100. Notice that the Bucket set Editor updates the values in the Range and Alias fields when you modify the End Point.

Field	Value
End Point	100
Description	Medium

6. Create the large bucket. Click **Add Bucket**, and then set the following values in the row that contains an End Point value of 101. You modify this End Point value to 1000.

Field	Value
End Point	1000
Description	Large

7. Click **OK**, and then click **Save**.

## Add Bucket Set to Assignment Condition

Add the bucket set to the assignment condition:

1. In the Conditions section of the decision table, in the R1 column, set the following values. Double-click each cell to add a value.

Row	Value in R1 Column
Group. groupType	Standard
fline. CategoryCode	Order
Header. <b>value</b>	True
<p>The header might contain more information, depending on your configuration. For example, it might contain <b>Header.Orig.Sys.Document.Ref==null</b>.</p>	
fline. OrderedQty	>=1000

2. Click **Save**.

## Configure the Action to Take

Configure the action to take if the condition is true:

1. In the decision table, scroll down to the Actions area.
2. In the R1 column, set the following values. Double-click each cell to add a value.

Row	Value
Modify Group	Contains a check mark.
processName: String	<p>AssignLaunchRules.ProcessName.<b>process_name</b>,</p> <p>where:</p> <ul style="list-style-type: none"> <li>o <b>process_name</b> identifies the name of the orchestration process where Order Management applies this assignment rule.</li> </ul> <p>For this example, use the following value:</p> <p><b>AssignLaunchRules.CallCustomerWhenLargeInvoice</b></p> <p>This code references an orchestration process named CallCustomerWhenLargeInvoice. For details about how to create this process, see Branching Orchestration Processes: Procedure.</p>
Call Print	Contains a check mark.

Row	Value
-----	-------

3. Click **Save**.

*Related Topics*

- [Using Visual Information Builder to Create Rules: Explained](#)
- [Using Visual Information Builder to Manage Process Assignment Rules: Procedure](#)
- [Creating Business Rules: Demo](#)

## Getting Data from Different Product Models When Using Oracle Business Rules: Explained

To optimize performance, data from the Oracle Product Model is not part of the data model that Order Management Cloud uses. If you encounter a situation where you must use item data from this product model in an Oracle Business Rule, then you can call the `getItemData` function to get this data. For example, if you require a rule that assigns an orchestration process to a fulfillment line if it contains a specific item category, then you can call the `getItemData` function to get this item category data.

### Using the `getItemData` Function

To use the `getItemData` function, you do the following work:

1. Navigate to the page where you write the business rule, such as the Manage Orchestration Process Assignment Rules page.
2. Add a rule that calls the `getItemData` function.
3. Select Highest priority for this rule. At run time, Order Management will run this rule before it runs any other rule that uses the data that `getItemData` returns.
4. Click **Edit Properties** to specify the arguments of the `getItemData` function. You use these arguments to specify the data that you require.
5. Add another rule that uses the data that `getItemData` returns.

### Arguments of the `getItemData` Function

You can use the following arguments of the `getItemData` function.

Argument Name	Description
<code>inventoryItemId</code>	Identifies the inventory item.
<code>inventoryOrganizationId</code>	Identifies the organization that includes the inventory item.
<code>fetchItemDefinition</code>	Gets the item definition. Use one of the following values: <ul style="list-style-type: none"> <li>• "Yes". Get the definition.</li> <li>• "No". Do not get the definition.</li> </ul> <p>You must include the double quotation marks.</p>

Argument Name	Description
fetchItemCategory	<p>Gets the item category. Use one of the following values:</p> <ul style="list-style-type: none"> <li>• "Yes". Get the category.</li> <li>• "No". Do not get the category.</li> </ul> <p>You must include the double quotation marks.</p>
fetchRelatedItem	<p>Gets related items. Use one of the following values:</p> <ul style="list-style-type: none"> <li>• "Yes". Get related items.</li> <li>• "No". Do not get related items.</li> </ul> <p>You must include the double quotation marks.</p>
customKey	<p>Makes sure duplicate data does not exist. It is recommended that you use the following value for this argument:</p> <p>FulfillLineId</p>
viewRowImpl	<p>Use this argument with one of the following items:</p> <ul style="list-style-type: none"> <li>• External interface routing rule</li> <li>• Assignment rule for an orchestration process</li> <li>• Transformation rule</li> </ul> <p>You use the following value for this argument:</p> <p><b>FulfillLineVO.viewRowImpl</b></p>

You can use the DOOFLine argument rules for an orchestration process. You use the following value for this argument:

**DoSeededOrchestrationRules.DOOFLine**

## Using a Custom Template to Get More Data

Only a subset of the Item Definition is available. If you require data that is not available in this subset, then you can use the getFusionAppsData function to get the data that you require. You can use the following arguments. The first five arguments are required. You can use the last five arguments to populate the attributes that you require. If the Value column includes double quotation marks, then you must include them.

Argument Name	Data Type	Value
inputAttributeName1	String	"InventoryItemId"
inputAttributeValue1	String	RuleType.FulfillLineVO. AttributeYouWanttoPopulate
inputAttributeName2	String	"InventoryOrganizationId"

Argument Name	Data Type	Value
inputAttributeValue2	String	RuleType. FulfillLineVO. AttributeYouWanttoPopulate
viewObjectName	String	"ItemPVO"
outputAttributeName1	String	"AttributeYouWanttoPopulate"
outputAttributeName2	String	"AttributeYouWanttoPopulate"
outputAttributeName3	String	"AttributeYouWanttoPopulate"
outputAttributeName4	String	"AttributeYouWanttoPopulate"
outputAttributeName5	String	"AttributeYouWanttoPopulate"

The Condition Browser displays the data type for each attribute at run time. You must specify the correct data type for this attribute when you write your rule.

## Getting Data from Different Product Models When Using Oracle Business Rules: Procedure

You can get data from a different product model when you use Oracle Business Rules.

For example, you can create the following assignment rule:

- If the item category on the fulfillment line equals a value, then assign an orchestration process to a group of one or more fulfillment lines that are associated with a customer

For this example, you create the following rules:

- If the fulfillment lines are associated with a specific customer, then get the item data.
- If the item category ID is 3574, then assign an orchestration process named **ManualShipOrderGeneric123** to a group.

### Summary of the Work

Get data from a different product model:

1. Create the If statement of the rule that gets the data.
2. Create the Then statement of the rule that gets the data.
3. Create the If statement of the rule that assigns the orchestration process.
4. Create the Then statement of the rule that assigns the orchestration process.

This example assumes the following:

- You already created an orchestration process named **ManualShipOrderGeneric123**.
- You already configured a product model for the organization and this model includes an item category named 3574.

This topic includes example values. You might use different values, depending on your business requirements.

## Create the If Statement of the Rule That Gets Data

In this section, you create the following condition:

- If the ship-to customer ID is 1304523

For a demonstration, see [Creating Business Rules: Demo](#).

Create the If statement of the rule that gets the data:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, search for, and then open Manage Orchestration Process Assignment Rules.
4. On the Manage Orchestration Process Assignment Rules page, click **Add Rule**.
5. Replace **Rule 1** with any descriptive text. For this example, enter **GetItemData**.
6. Make sure Order Management Cloud runs this rule first:
  - Next to GetItemData, click **Expand**, and then click **Show Advanced Settings**.
  - Set the following value, and then click **Collapse Pane**.

Field	Value
Priority	Highest

7. Specify the attribute:
  - Click **Left Value**.
  - In the Condition Browser, expand **AssignLaunchRules**, expand **FulfillLineVO**, click **ShipToCustomerId**, and then click **OK**.
8. Specify the value for the attribute. In the Right Value field, enter **1304523**.
9. Click **Save**.

## Create the Then Statement of the Rule That Gets Data

In this section, you create the Then statement of the rule that gets data:

- Call the getItemData function to get the data for the category.

Create the Then statement of the rule that gets the data:

1. Call the getItemData function:
  - In the Then area, click **Add Action**, and then click **Call**.
  - In the Select a Target list, enter **AssignLaunchRules.getItemData**. If a small list displays after you enter this value, then click **AssignLaunchRules.getItemData** in this list.
2. Specify the arguments of the getItemData function:
  - Click **Edit Properties**.
  - In the Arguments dialog, specify the following values. Do not include a space before or after each period (.). You must include the double quotation marks (").

Name	Value
inventoryItemid	AssignLaunchRules.FulfillLineVO.inventoryItemid

Name	Value
inventoryOrganizationId	AssignLaunchRules. FulfillLineVO. inventoryOrganizationId
fetchItemDefinition	"No"
fetchItemCategory	"Yes"
fetchRelatedItem	"No"
customKey	AssignLaunchRules. FulfillLineVO. FulfillLineId. toString()
viewRowImpl	AssignLaunchRules. FulfillLineVO. ViewRowImpl

3. Click **OK**, and then click **Save**.
4. Click **Collapse** next to GetItemData.

## Create the If Statement of the Rule That Assigns the Orchestration Process

In this section, you create the following condition:

- If the ID of the item category that the getItemData function returns is 3574

When you complete this section, the If statement that you create must resemble the following configuration.

The screenshot shows the configuration of an IF statement in the Oracle Orchestration Rule Editor. The rule is titled "AssignShipOrderGenericIf3524Cate". The IF statement is composed of three main conditions:

- Group** is a **AssignLaunchRules,ResponseRL**.
- FLine** is a **AssignLaunchRules,FulfillLineVO** and **FLine.ShipToCustomerId** is **1304523**.
- ItemCategory** is a **AssignLaunchRules.ItemCategory** and **ItemCategory** isn't **null**, **ItemCategory.categoryId** is **3524L**, and **ItemCategory.customKey** is **FLine.FulfillLineId.toString()**.

Create the If statement of the rule that assigns the orchestration process:

1. Click **Add Rule**.
2. Replace **Rule 1** with any descriptive text. For this example, enter **AssignShipOrderGenericIf3524Category**.
3. Add a check mark to **Advanced Mode**.

4. Create the condition for the Group alias:
  - o In the If area, in the field to the left of **Is A**, enter **Group**.
  - o In the If area, in the field to the right of **Is A**, click **AssignLaunchRules.ResponseRL**.
  - o In the If area, next to the Right Value field, click the **down arrow**, and then click **Delete Test**.
5. Create the condition for the FLine alias:
  - o In the If area, click **Add Pattern**.
  - o In the field below **And**, enter **FLine**.
  - o To the right of **Is A**, click **AssignLaunchRules.FulfillLineVO**.
6. Add a test for the FLine alias:
  - o Below the field that contains FLine, click **Add Test**.
  - o Click **Left Value**.
  - o In the Condition Browser dialog, expand **FLine**, click **ShipToCustomerId**, and then click OK.
  - o In the Right Value field, enter **1304523**.
7. Create the condition for the ItemCategory alias:
  - o To the right of the field that contains AssignLaunchRules.FulfillLineVO, click **Add Pattern**.
  - o In the left value field below **And**, enter an alias, such as **ItemCategory**.
  - o To the right of **Is A**, click **AssignLaunchRules.ItemCategory**.
  - o Click **Save**.
8. Add a test for the ItemCategory alias:
  - o Below the field that contains ItemCategory, click **Add Test**.
  - o In the field below the field that contains ItemCategory, enter **ItemCategory**.
  - o In the list of operators to the right of the field that contains ItemCategory, click **Isn't**.
  - o Click **Right Value**.
  - o In the Condition Browser, click **null**, and then click **OK**.
9. Add another test for the ItemCategory alias:
  - o Click the arrow next to Right Value, and then click **Simple Test**.
  - o Below the field that contains ItemCategory for the isn't condition, click **Left Value**.
  - o In the Condition Browser, expand **ItemCategory**, click **categoryId**, and then click **OK**.
  - o In the Right Value field, enter **3524L**.
10. Add another test for the ItemCategory alias:
  - o Click the **down arrow** next to Right Value, and then click **Simple Test**.
  - o Below the field that contains ItemCategory.categoryId, click **Left Value**.
  - o In the Condition Browser, expand **ItemCategory**, click **customKey**, and then click **OK**.
  - o Click **Right Value**.
  - o In the Condition Browser, expand **FLine**, expand **FulfillLineId**, click **toString**, and then click **OK**.

## Create the Then Statement of the Rule That Assigns the Orchestration Process

Create the Then statement of the rule that assigns the orchestration process:

1. In the Then area, click **Add Action**, and then click **Modify**.
2. Click **Select a Target**, and then click **Group**.
3. Click **Edit Properties**.
4. In the Properties dialog, set the following value, and then click **OK**. You must include the double quotation marks.

Name	Value
processName	"ManualShipOrderGeneric123"

5. Click **Save**.

### Related Topics

- [Creating Business Rules: Demo](#)

## Pausing Orchestration Processes

### Pausing Orchestration Processes: Explained

A pause task temporarily stops an orchestration process from running so that it can wait for a condition to be met. When the condition is met, the orchestration process releases this pause task, and then proceeds to the next orchestration process step. You can use a pause task to wait between tasks, and to specify when to release the pause and begin the next orchestration process step.

You can use a pause task in the following business scenarios.

Scenario	Description
Pause an orchestration process until an event occurs.	<p>Assume your company provides a layaway program that allows a customer to reserve an item, and then pay for it in installments. You can use a pause task to pause the orchestration process. When the customer pays in full, the orchestration process calls a public service named Release Paused Tasks, releases the pause task, and then ships the item to the customer.</p> <p>In another example, assume your company must release orders for shipping only if inventory is available at the fulfillment location. You can use a pause task that pauses the orchestration process until the inventory is available. When the inventory is available, the orchestration process calls a public service named Release Paused Tasks, releases the pause task, and then ships the item to the customer.</p>
Pause an orchestration process until time elapses.	<p>Assume a customer cannot receive the goods that they order too early. You could use the requested ship date to prevent early delivery, but the warehouse might not recognize this date. You can set up a pause task that pauses the orchestration process until two days before the scheduled ship date occurs.</p> <p>In another example, assume your company sells a video game up to seven days before the release date of the game. This lead time allows your company to ship the game as soon as it is available. Your company allows seven days because it estimates that it requires this number of days to schedule and process the sales order before it ships this order. You can set up a pause task that pauses the orchestration process before the schedule step occurs, and continues this pause</p>

Scenario	Description
Pause an orchestration process until a dependency resolves.	<p>until the publisher releases the game. In this situation, you can use a rule that instructs Order Management Cloud to release the pause task according to a flexfield that includes a value of the release date minus seven days. Order Management releases the pause task, schedules the sales order, and then resumes the orchestration process.</p> <hr/> <p>Consider the following examples:</p> <ul style="list-style-type: none"> <li>• A customer requests to receive a shipment that includes coffee, paper cups, sugar, and creamer at the same time. You can use a pause task that pauses each order line until the orchestration process finishes scheduling each of these lines. The orchestration process periodically evaluates the sales order. It finishes scheduling all lines, and then ships them to the customer at the same time.</li> <li>• A customer must receive invoices for all order lines of a sales order at the same time, regardless of when the orchestration process ships each of these lines. You can use a pause task that pauses the orchestration process until it ships all items. It then sends the invoices for all lines to the customer at the same time.</li> <li>• A shipping system cannot handle an early request. It can only handle a request for a sales order that Order Management schedules to ship within two days. You can pause the orchestration process so that it does not send a shipping request to the shipping system until two days before the ship date occurs.</li> <li>• Assume your company policy requires manual review and approval of any sales order where the scheduled date occurs after the requested date. You can use a pause task to pause the orchestration process until the review and approval finishes.</li> </ul>

## Resuming a Paused Orchestration Process

You can do the following work to release the pause task that is pausing an orchestration process:

- **Release automatically.** Create a scheduled process that releases the pause task.
- **Release manually.** Navigate to the Orchestration Process details page, and then click **Release Pause Task**. You must use the Order Orchestration Error Recovery Manager role.

## Pausing Orchestration Processes Until an Event Occurs: Procedure

You can define an orchestration process that uses a pause task to temporarily stop the orchestration process from running so that it can wait for an event to occur.

Assume you work for a publisher who will release a new book at some point in the future, and that you need to provide your Gold customers an opportunity to order the book before your company releases it to the general public. You can define an orchestration process that pauses processing so that it does not attempt to ship these books before the release date occurs.

You create the following condition:

- If the customer demand class is Gold, then use a scheduled process named Release Pause Tasks. This scheduled process searches for the condition that releases the pause task.

You create the following rules:

1. If the conditions are true, then pause the task.
2. If the conditions of the first rule are not true, then skip the pause task.

If you use an If, Then, or Else condition in a single rule instead of creating two rules, then you must create the Else condition before you add an action in the Then condition.

## Summary of the Work

Pause an orchestration process until an event occurs:

1. Define the orchestration process.
2. Define the orchestration process steps.
3. Create the If statement for the first rule.
4. Create the Then statement for the first rule.
5. Create the If statement for the second rule.
6. Create the Then statement for the second rule.
7. Deploy the orchestration process. For more information, see *Deploying Orchestration Processes: Procedure*.

This topic includes example values. You might use different values, depending on your business requirements.

## Define the Orchestration Process

Define the orchestration process:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, search for, and then open *Manage Orchestration Process Definitions*.
4. On the *Manage Orchestration Process Definitions* page, click **Actions**, and then click **Create**.
5. On the *Create Orchestration Process Definition* page, set the following fields, and then click **Save**.

Field	Description
Process Name	Enter an orchestration process name that does not include any spaces, such as <i>OrchestrationProcess1</i> .
Process Display Name	Enter any value. Do not include spaces.
Process Class	For this example, select <b>Ship Order Class</b> .
Set	For this example, select <b>Common Set</b> .

## Define the Orchestration Process Steps

Define the orchestration process steps:

1. On the Step Definition tab, click **Actions**, and then click **Add Row**.
2. Set the following fields, and then click **Save**.

Field	Value
Step Name	Schedule Goods
Step Type	Service
Task Type	Schedule
Task	Schedule

Field	Value
Service	Create Scheduling

3. Add more steps. Repeat steps 1 and 2, using the following values.

- o Create Reservation
- o Create Shipment
- o Wait for Shipment Advice
- o Create Invoice
- o Wait for Invoice

Note the following:

- o Set the Step Type for all steps to Service.
  - o Set the other fields so that they reflect the step behavior. For example, for the Create Reservation step, set the Task Type to Reservation, the Task to Reserve, and the Service to Create Inventory Reservation.
  - o Make sure you set the Exit Criteria field for each wait step. For example, set the Exit Criteria for the Wait for Shipment Advice step to Shipped.
4. Add the pause step. Click the Create Reservation step, click **Add Row**, set the following values in the new row, and then click **Save**.

Field	Value
Step Name	Pause
Step Type	Service
Task Type	Pause
Task	Pause
Service	Pause Process

## Create the If Statement for the First Rule

In this section you create the start-after condition. You create a business rule. For a demonstration, see Creating Business Rules: Demo.

Create the If statement for the first rule:

1. On the Pause step, in the Start-After Condition column, click **Click for Rule**.
2. In the Start After Condition dialog, click **Add Rule**, and then click **Expand**.
3. Click **Left Value**.
4. In the Condition Browser dialog, expand **DooSeededOrchestrationRules**, expand **DOOFLine**, click **demandClassCode**, and then click **OK**.
5. Click **Right Value**.

6. In the Condition Browser dialog, enter "**Gold**", and then click **OK**. You must include the double quotation marks (").

## Create the Then Statement for the First Rule

Create the Then statement for the first rule:

1. In the Then area, click **Add Action**, and then click **Assert New**.
2. Click **Select a Target**, and then click **DooSeededOrchestrationRules.SacResult**.
3. Click **Edit Properties**.
4. In the Properties dialog, in the reevaluateFlag row, enter "**N**". You must include double quotation marks.
5. In the sacType row, click **Value**.
6. In the Condition Browser dialog, expand **DooSeededOrchestrationRules**, expand **SacResult**, click **SAC\_TYPE\_EVENT**, and then click **OK**.
7. In the eventName row, in the Value field, enter "**Gold Preorders**". You must include double quotation marks.

Note that a setting on a scheduled process references this event, but you can also use it in a public service. You can also use other parameters to release a pause task. Note that this event is not related to the events that the event framework in Oracle Middleware uses.

8. Click **OK**, click **Save**, and then click **Save** again.

## Create the If Statement for the Second Rule

Create the If statement for the second rule:

1. On the Pause step, in the Start-After Condition column, click **Click for Rule**.
2. In the Start After Condition dialog, click **Add Rule**. Note that the dialog displays a new rule and enters a value of **Rule 2** in a field. Make sure you do the rest of this procedure in this rule.
3. In the THEN area, click **Expand**.
4. Click **Left Value**.
5. In the Condition Browser dialog, expand **DooSeededOrchestrationRules**, expand **DOOFLine**, click **demandClassCode**, and then click **OK**.
6. Click **Is**, and then click **Isn't**.
7. Click **Right Value**.
8. In the Condition Browser dialog, enter "**Gold**", and then click **OK**. You must include the double quotation marks.

## Create the Then Statement for the Second Rule

Create the Then statement for the second rule:

1. In the Then area, click **Add Action**, and then click **Assert New**.
2. Click **Select a Target**, and then click **DooSeededOrchestrationRules.SacResult**.
3. Click **Edit Properties**.
4. In the Properties dialog, in the reevaluateFlag row, enter "**N**". You must include double quotation marks.
5. In the sacType row, click **Value**.
6. In the Condition Browser dialog, expand **DooSeededOrchestrationRules**, expand **SacResult**, click **SAC\_TYPE\_IMMEDIATE**, and then click **OK**.
7. Click **Save**.
8. On the Edit Orchestration Process Definition page, click **Save**.

### Related Topics

- [Creating Business Rules: Demo](#)

## Pausing Orchestration Processes Until Time Elapses: Procedure

You can define an orchestration process that uses a pause task to temporarily stop the orchestration process from running so that it can wait for time to elapse.

In this example, you define an orchestration process that pauses before it ships preordered items. You create the following condition:

- If the source system is Legacy, then release the pause on ShipDate minus two.

You create the following rules:

1. If the condition is met, then release the pause on a date.
2. If the condition is not met, then release the pause immediately.

### Summary of the Work

Pause an orchestration process until an event occurs:

1. Define the orchestration process and orchestration process steps. To define the orchestration process and to create the pause task for this example, use the instructions described in Pausing Orchestration Processes Until an Event Occurs: Procedure.
2. Create the If statement for the first rule.
3. Create the Then statement for the first rule.
4. Create the If statement for the second rule.
5. Create the Then statement for the second rule.
6. Deploy the orchestration process. For details, see Deploying Orchestration Processes: Procedure.

This topic includes example values. You might use different values, depending on your business requirements.

### Create the If Statement for the First Rule

In this section you create a business rule. For a demonstration, see Creating Business Rules: Demo. This rule uses a pause task to temporarily stop processing if the source system is Legacy.

Create the If statement for the first rule.

1. On the Pause step, in the Start-After Condition column, click **Click for Rule**.
2. In the Start After Condition dialog, click **Add Rule**.
3. Replace Rule 1 with any value that does not include spaces. For this example, enter `ResumeShippingMinus2`.
4. Click **Expand**, and then click **Show Advanced Settings**.
5. Add a check mark to **Advanced Mode**, and then add a check mark to **Tree Mode**. The Tree Mode maintains the order hierarchy and the orchestration process hierarchy when Order Management Cloud evaluates the rule.
6. Set the **Root** to `DooSeededOrchestrationRules.DOOHeader`.
7. In the IF area, in the left field, delete `DooSeededOrchestrationRules.DOOHeader`, and then enter `Header`.
8. In the IF area, click the **down arrow** that is located next to Right Value, and then click **Delete Test**.
9. Click **Add Pattern**.
10. In the left field, under Add Test, enter `Line`.
11. Set the field that is located to the right of **Is A** to `Header/childFLines`.
12. Click **Add Test**, and then click **Simple Test**. Make sure you use Add Test that is located immediately below the field that you edited in step 7.
13. Click **Left Value**.
14. In the Condition Browser, expand **Header**, click `sourceOrderSystem`, and then click **OK**.

15. In the Right Value field, enter "LEG". LEG is an abbreviation for the term legacy. You can use any string. You must include the double quotation marks when you use a string.
16. In the Condition Browser dialog, enter "Gold", and then click **OK**. You must include the double quotation marks.

## Create the Then Statement for the First Rule

Create the Then statement for the first rule. This statement releases the pause task two days before the scheduled ship date occurs:

1. In the Then area, click **Add Action**, and then click **Assign New**.
2. Click **Select a Target**, and then click **DooSeededOrchestrationRules.Timestamp**.
3. In the empty field immediately to the right of the Select a Target field, enter `DateTime`. Do not include quotation marks. `DateTime` is the name of a variable that this example uses for the date calculation.
4. Click **Expression Value**.
5. In the Condition Browser dialog, expand **Line**, click **scheduleShipDate**, and then click **OK**.
6. Click **Add Action**, and then click **Modify**.
7. Click **Select a Target**, and then click **DateTime**.
8. Click **Edit Properties**.
9. In the Properties dialog, in the Time row, click **Value**.
10. In the Condition Browser dialog, expand **DateTime**, and then click **Time**. The Condition Browser adds a value of `DateTime.time` to the Expression field.
11. In the Expression field, append `-(2*24*60*60*1000)` to the end of `DateTime.time`. This value calculates the number of milliseconds so that the pause task ends two days before Order Management schedules the order lines for shipping.
12. Make sure the Expression field includes the following value, and then click **OK**: `DateTime.time-(2*24*60*60*1000)`.
13. In the Properties dialog, click **OK**.
14. Click **Add Action**, and then click **Assert New**.
15. Click **Select a Target**, and then click **DooSeededOrchestrationRules.SacResult**.
16. Click **Edit Properties**.
17. In the Properties dialog, in the reevaluateFlag row, in the Value field, enter "N". You must include the double quotation marks.
18. In the sacType row, click **Value**.
19. In the Condition Browser dialog, expand **DooSeededOrchestrationRules**, expand **SacResult**, click **SAC\_TYPE\_TIMER**, and then click **OK**.
20. In the Properties dialog, in the waitDateTime row, click **Value**.
21. In the Condition Browser, click **DateTime**, click **OK**, and then click **OK** again.

## Create the If Statement for the Second Rule

Create the If statement for the second rule. This rule releases the pause if the source system is not Legacy:

1. In the Start After Condition dialog, click **Add Rule**.
2. Replace Rule 1 with any value that does not include spaces. For this example, enter `ResumeShippingMinus2`.
3. Click **Expand**, and then click **Show Advanced Settings**.
4. Add a check mark to **Advanced Mode**, and then add a check mark to **Tree Mode**.
5. Set the **Root** to `DooSeededOrchestrationRules.DOOHeader`.
6. In the IF area, in the left field, delete `DooSeededOrchestrationRules.DOOHeader`, and then enter `Header`.
7. In the IF area, click the **down arrow** that is located next to Right Value, and then click **Delete Test**.
8. Click **Add Pattern**.
9. In the left field, under Add Test, enter `Line`.
10. Set the field that is located to the right of **Is A** to `Header/childFLines`.
11. Click **Add Test**, and then click **Simple Test**. Make sure you use Add Test that is located immediately below the field that you edited in step 6.

12. Click **Left Value**.
13. In the Condition Browser, expand **Header**, click **sourceOrderSystem**, and then click **OK**.
14. In the Right Value field, enter "**LEG**". You must include the double quotation marks.

## Create the Then Statement for the Second Rule

Create the Then statement for the second rule:

1. In the Then area, click **Add Action**, and then click **Assert New**.
2. Click **Select a Target**, and then click **DooSeededOrchestrationRules.SacResult**.
3. Click **Edit Properties**.
4. In the Properties dialog, in the reevaluateFlag row, in the Value field, enter "**N**". You must include the double quotation marks. This example uses reevaluateFlag to specify whether or not to evaluate the rule after the calculated date occurs.
5. In the sacType row, click **Value**.
6. In the Condition Browser dialog, expand **DooSeededOrchestrationRules**, expand **SacResult**, click **SAC\_TYPE\_IMMEDIATE**, click **OK**, and then click **OK** again.
7. In the Start-After Condition dialog, click **Save**.
8. On the Edit Orchestration Process Definitions page, click **Save**.

### Related Topics

- [Creating Business Rules: Demo](#)

## Pausing Orchestration Processes Until Items Are Ready to Ship: Procedure

This example describes how to coordinate orchestration processes to make sure that Order Management Cloud invoices all the order lines that a sales order contains at the same time, even if the shipment dates vary. The orchestration process uses the same sequence of steps to process each fulfillment line, but these lines might not be synchronized.

Assume a school district places a sales order for 600 new history books and requires that they receive the invoice for this sales order only after Order Management ships all the books.

Assume that the orchestration process must get the books from the following warehouses:

- Fulfillment line 1 gets 80 books from the Seattle warehouse
- Fulfillment line 2 gets 400 books from the Denver warehouse
- Fulfillment line 3 gets 120 books from the Chicago warehouse

You define an orchestration process that processes the order lines, pauses them until all of these lines are shipped, and then sends them to billing.

## Summary of the Work

Pause an orchestration process until items are ready to ship:

1. Define the orchestration process.
2. Create the default rule.
3. Create the rule that pauses the orchestration process.

This topic includes example values. You might use different values, depending on your business requirements.

## Define the Orchestration Process

Define the orchestration process. For an example that includes details about how to do this work, see [Pausing Orchestration Processes Until an Event Occurs: Procedure](#):

1. Create a new orchestration process and define the header.
2. Add the following orchestration process steps:
  - o Schedule
  - o Reserve
  - o Ship
  - o Wait for Shipment
  - o Pause
  - o Invoice
  - o Wait for Billing

## Create the Default Rule

In this section you create a business rule. For a demonstration, see [Creating Business Rules: Demo](#).

Create the default rule:

1. Click **Add Rule**.  
In the Start After Condition dialog, click **Add Rule**, click **Expand**, and then click **Show Advanced Settings**.
2. Set Priority to Highest.
3. Add a check mark to **Advanced Mode**.
4. In the field below If, enter an alias, such as `Header`.
5. In the field to the right of Is A, select **DooSeededOrchestrationRules.DOOHeader**.
6. Click the **down arrow** next to Right Value, and then click **Delete Test**. If you remove the test, then the rule always applies the actions that the Then area contains.
7. In the Then area, click **Add Action**, and then click **Assign New**.
8. Click **Select a Target**, and then click **DooSeededOrchestrationRules.SacResult**.
9. In the field to the left of the equal (=) sign, enter `sac`.
10. In the field to the right of the equal sign (=), click **Expression Value**.
11. In the Condition Browser dialog, expand **DooSeededOrchestrationRules**, expand **SacResult**, click **New**, and then click **OK**.
12. Click the **down arrow** next to Add Action, and then click **Assign**.
13. Click **Select a Target**, and then click **Header.sacResult**.
14. In the field to the right of the equal sign (=), enter `sac`.
15. Click the **down arrow** next to Add Action, and then click **Assign**.
16. Click **Select a Target**, and then click **Header.sacResult.sacType**.
17. In the field to the right of the equal sign (=), click **Expression Value**.
18. In the Condition Browser dialog, expand **DooSeededOrchestrationRules**, expand **SacResult**, click **SAC\_TYPE\_IMMEDIATE**, and then click **OK**.
19. Click **Validate**, and then click **Save**.

## Create the Rule That Pauses the Orchestration Process

Create the following rule:

- If the actual ship date is not null on all fulfillment lines, then release the pause task.

Create the rule that pauses the orchestration process:

1. On the Pause step, in the Start-After Condition column, click **Click for Rule**.
2. In the Start After Condition dialog, click **Add Rule**, click **Expand**, and then click **Show Advanced Settings**.
3. Add a check mark to **Advanced Mode**, and then add a check mark to **Tree Mode**.
4. Set Root to **DooSeededOrchestrationRules.DOOHeader**.
5. In the If area, in the field to the left of Is A, replace the default value with an alias, such as **OrderHeader**.
6. Click the **down arrow** to the right of Right Value, and then click **Delete Test**.
7. Click **Add Pattern** to the right of the If statement.
8. To the right of the empty fields, click **Surround Pattern with Parenthesis**.
9. In the list of values under And, click **There is a Case Where**. This selection means that even if the fulfillment system shipped all lines except one line, then pause all shipped lines.
10. In the field on the left, under There is a Case Where, enter **FLine**.
11. In the field to the right of Is A, select **Header/allFLinesIntheOrder**. This selection allows Order Management to coordinate across multiple orchestration processes.
12. Click the **down arrow** next to Add Test, and then click **Simple Test**. This is the test for the reason to pause.
13. Click **Left Value**.
14. In the Condition Browser, expand **FLine**, click **actualShipDate**, and then click **OK**.
15. Click **Right Value**.
16. In the Condition Browser, click **null**, and then click **OK**.
17. In the Then area, click the **down arrow** next to Add Action, and then click **Assign**.
18. Click **Select a Target**, and then click **OrderHeader.sacResult.eventName**.
19. Click **Expression Value**.
20. In the Condition Browser, expand **DooSeededOrchestrationRules**, expand **SacResult**, select **SAC\_SYSTEM\_EVENT\_IPC\_PAUSE**, and then click **OK**. This step configures Order Management to evaluate the order lines until it ships all of them, and then to release them.
21. Click the **down arrow** next to Add Action, and then click **Assign**.
22. Click **Select a Target**, and then click **OrderHeader.sacResult.reevaluateFlag**.
23. In the field to the right of the equal sign (=), enter **"Y"**. You must include the double quotation marks.
24. Click the **down arrow** next to Add Action, and then click **Assign**.
25. Click **OrderHeader.sacResult.sacType**.
26. Click **Expression Value** on the same line.
27. In the Condition Browser, expand **DooSeededOrchestrationRules**, expand **SacResult**, click **SAC\_TYPE\_EVENT**, and then click **OK**.
28. Click **Validate**, and then click **Save**.

#### Related Topics

- [Creating Business Rules: Demo](#)

## Automatically Resuming Paused Orchestration Processes: Explained

You can use a scheduled process or a public service to release a pause task so that an orchestration process can automatically resume processing sales orders.

### Using a Scheduled Process to Release Pause Tasks

You can use the Scheduled Processes page to run a scheduled process named Release Pause Tasks. This scheduled process can release one or more pause tasks.

You can use any of the following settings to identify the pause tasks to release:

- Event Name
- Inventory Item
- Sold-to Customer
- Task
- Order Number
- Fulfillment Line Number
- Warehouse
- From Order Date
- To Order Date
- From Scheduled Ship Date
- To Scheduled Ship Date
- From Scheduled Release Date
- To Scheduled Release Date
- Source Order System
- Source Order Number

For an example that uses this scheduled process, see [Pausing Orchestration Processes Until an Event Occurs: Procedure](#).

## Using a Public Service to Release Paused Tasks

An external application can call a public service named Release Paused Tasks to release a pause task. You can use the settings of this service to identify the pause tasks to release. The event handler calls the service to release the pause tasks according to the settings that you specify.

To get the attributes that the sales order references, you can use the order capture service that requests the order details.

## Managing Hold Codes: Procedure

You can define a custom hold code that allows your users to place a hold on a sales order. This hold pauses an action that is currently running on a business object or business service that an orchestration process references.

You can define a hold code only for a hold that originates in Order Management. You cannot use it for a hold that originates in some other application.

### Summary of the Work

For this example, assume you are an order administrator for Vision Corporation, and your sales engineers have informed you that a problem might exist with an item. You must create a hold code that your users can use to temporarily hold all sales orders that reference this item while your sales engineers investigate the problem. Do the following work:

1. Create a hold code.
2. Create a release reason.
3. Test your work.

This topic includes example values. You might use different values, depending on your business requirements.

## Create a Hold Code

Create a hold code:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, search for, and then open Manage Hold Codes.
4. On the Manage Hold Definitions page, click **Actions**, click **Add Row**, and then set the following values.

Field	Value
Code	HOLD_ PRODUCT_ISSUE
Name	Hold to Investigate Product
Description	This hold allows us to investigate a problem that occurred with the item.
Hold All Services	Contains a check mark.  This value specifies to apply a hold on all services. Task layer services determine whether or not Order Management attaches a hold code to the fulfillment line, or to the sales order for one or more fulfillment tasks that the orchestration process references.
Set	COMMON  You can use Set to specify the set that Order Management uses with a hold code. You can assign a hold code to a single business unit, or you can assign multiple hold codes to sets of business units. These sets can share hold codes.

5. Optional. Specify each service where Order Management must apply the hold. For example, do the following work to apply the hold only for the Shipment service:
  - o Make sure Hold All Services does not contain a check mark.
  - o In the Hold to Investigate Product: Services area, click **Actions**, and then click **Select and Add**.
  - o the Select and Add: Services dialog, leave all fields empty, and then click **Search**.
  - o Click a **row** that displays Shipment in the Task Type column.
  - o For this example, you must apply the hold to all services, so click **Cancel**, and then add a check mark to Hold All Services.
6. Click **Save and Close**.

## Create a Release Reason

Create a reason that your users can choose to indicate why they released the hold:

1. On the Setup page, search for, and then open Manage Standard Lookups.
2. On the Manage Standard Lookups page, enter the following value, and then click **Search**.

Field	Value
Lookup Type	DOO

3. In the Search Results, click the **row** that displays DOO\_HLD\_RELEASE\_REASON in the Lookup Type column.
4. In the DOO\_HLD\_RELEASE\_REASON: Lookup Codes area, click **Actions**, click **New**, and then set the following values.

Field	Value
Lookup Code	DOO_HLDREL_ PRODUCT_OK
Display Sequence	1
Start Date	The current date.
End Date	One week from today.
Meaning	Fixed the Product Problem
Description	Fixed the possible problem with item.

5. Click **Save and Close**.

## Test Your Work

Test your work:

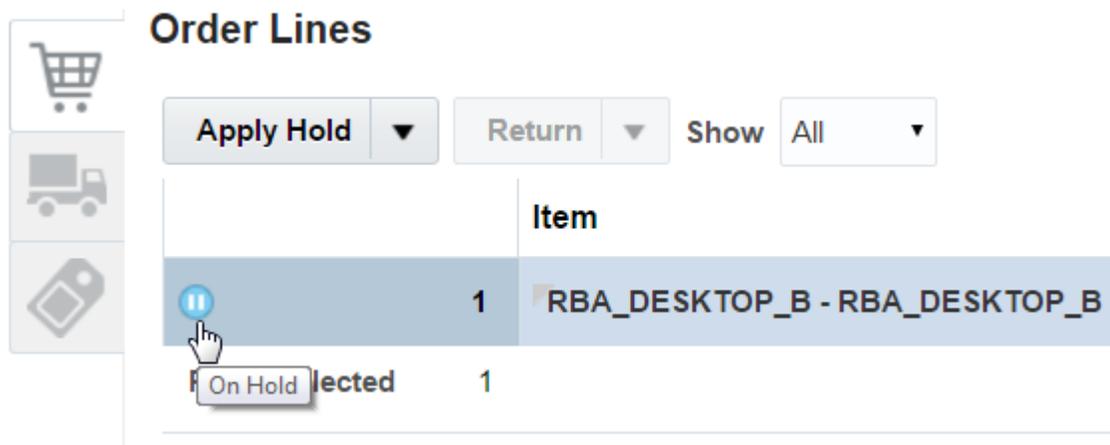
1. In the Navigator, click **Order Management**.
2. On the Overview page, click **Actions**, and then click **Manage Orders**.
3. On the Manage Orders page, enter the following value, and then click **Search**.

Field	Value
Status	Equals Processing

4. In the Search Results, click a **link** in the Order column.
5. On the Order page, in the Order Lines area, click **Apply Hold**.
6. In the Apply Hold dialog, set the following value, and then click **Save and Close**.

Field	Value
Hold Name	Hold to Investigate Product This is the name of the hold that you created earlier in this topic.

- In the Order Lines area, to verify that Order Management placed a hold on the order line, make sure it displays the following On Hold icon.



- Click the **arrow** next to Apply Hold, and then click **Release Hold**.
- In the Release Hold dialog, set the following values, and then click **Save and Close**. Note that you defined these values earlier in this topic.

Field	Value
Hold Name	Hold to Investigate Product
Release Reason	Fixed the Product Problem

*Related Topics*

- Using Holds to Temporarily Stop Processing: Explained



# 8 Using Visual Information Builder to Create Rules

## Using Visual Information Builder to Create Rules: Explained

Prior to Release 11, you use Oracle Business Rules to create rules in Order Management Cloud. Starting with release 12, you can use Visual Information Builder, which is a rule editor that supports a simplified drag-and-drop interface. It helps you to visualize data, visualize your business processes, customize your business logic, and customize your business rule sets.

You can use either editor. To access them, use the following pages.

Editor for Oracle Business Rules	Editor for Visual Information Builder
Manage External Interface Routing Rules	Manage External Integration Routing Rules for Sales Orders
Manage Pretransformation Defaulting Rules	Manage Pretransformation Rules for Sales Orders
Manage Orchestration Process Assignment Rules	Manage Process Assignment Rules for Sales Orders

Note the following limitations when using Visual Information Builder:

- You cannot use a decision table. If you created a rule in an earlier version of Order Management that uses a decision table, and if you prefer to use Visual Information Builder to replace this rule, then you can use the IF-THEN-ELSE statement in Visual Information Builder to replace the decision table logic. For an example that uses the IF-THEN-ELSE statement instead of a decision table, see *Using Visual Information Builder to Manage Process Assignment Rules: Procedure*.
- It is recommended that you use Visual Information Builder or Oracle Business Rules in your implementation. It is recommended that you do not use Visual Information Builder and Oracle Business Rules at the same time in the same implementation.
- You cannot define a global variable for use in a rule.
- You cannot define collections of objects in conditions.

### Examine the Behavior of a Predefined Rule

In this section, you observe how the rule editor establishes the following condition:

- If task type is shipment, then set the connector to Connect to Oracle Fusion Shipping System

Examine the behavior of a predefined rule:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, search for, and then open *Manage External Integration Routing Rules for Sales Orders*.
4. On the *Manage External Interface Routing Rules* page, notice the predefined routing rules.

You can use this page to do the following work:

- o Activate or inactivate a rule.
- o Create a new rule.
- o Change the rule priority.
- o View, modify, duplicate, or delete a rule.
- o Publish active rules.

5. Right-click **Shipping Task Routing**, click **Actions**, and then click **Edit**.
6. Click **IF**, and then notice the phrase **Task Type is Equal to Shipment**.
7. Click **DO**, and then notice the phrase **Set Connector Name, Connector Name: Connect to Oracle Fusion Shipping System**.
8. Click **Close**.

## Using Visual Information Builder to Manage External Interface Routing Rules: Procedure

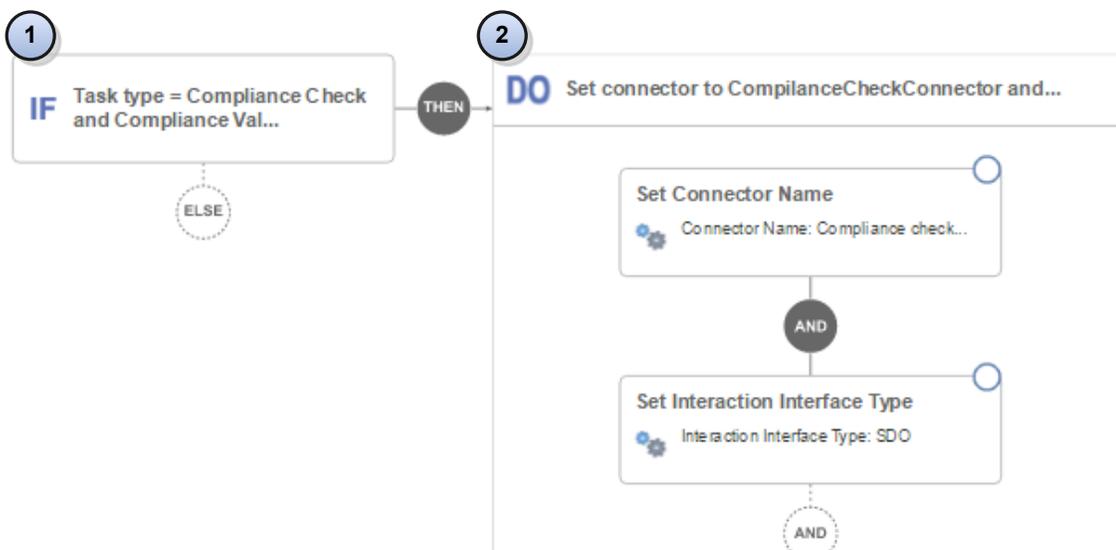
You can use Visual Information Builder to create routing rules.

Assume you must create a routing rule that implements the following condition:

- If the task type is Compliance Check, and if the compliance value is less than 100, then set the connector to ComplianceCheckConnector, and set the interface type to SDO

### Summary of the Work

In this example, you create the following rule:



Use Visual Information Builder to create external interface routing rules:

1. Create the routing rule and the If condition.
2. Create the Then statement.
3. Activate and publish your rule.

This topic includes example values. You might use different values, depending on your business requirements.

## Create the Routing Rule and IF Condition

Create the routing rule and If condition:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, search for, and then open Manage External Integration Routing Rules for Sales Orders.
4. On the Manage External Interface Routing Rules page, click **Create New Rule**.
5. Change the rule name.

Current Name	New Name
New Rule	Routing Rule for Compliance Check

6. Name the If statement so that it reflects the meaning of the condition:
  - o Click **Enter Description**.
  - o In the Enter Description dialog, enter **Task Type = Compliance Check and Compliance Value Is Less Than 100**, and then click **OK**.

7. Click **New Condition** (the dashed circle in the IF area).

8. In the Create Condition dialog, enter **Task**, wait a moment, and then click **Task Type (Order Header)**.

Notice that the dialog displays suggestions when you wait a moment after you finish typing.

9. In the Create Condition dialog, click **Search**.

10. In the Search dialog, enter **Compliance**, and then click **Search**.

11. Choose **Compliance Check**, click **OK**, and then click **OK** again.

Notice that the IF area now displays the following condition:

- o **Task Type is Equal to Compliance Check**

12. Click **AND**.

13. In the Create Condition dialog, enter **Compli**, and then wait a moment.

Notice that the dialog displays a number attributes, such as **Compliance Date (Header Compliance Details)**. Extensible flexfields that are defined on the order header provide the values for these attributes. The order header displays them as typical attributes in the Order Management work area.

14. Click **Compliance Value (Header Compliance Details)**.

15. Change the operator from **=** (equal), to **<** (less than).

16. Change the value from **0.0**, to **100**, and then click **OK**.

Notice that the IF area now displays the following condition:

- o **Task Type is equal to Compliance Check AND Compliance Value < 100**

## Create the THEN Statement

Create the Then statement:

1. Click **THEN**, and then click **DO**.
2. Name the DO so that it reflects the meaning of the action:
  - o Click **Enter Description**.
  - o In the Enter Description dialog, enter **Set connector to ComplianceCheckConnector and interface type to SDO**, and then click **OK**.
3. Click **New Action** (the dashed circle in the DO area), and then click **Perform an Action**.
4. In the Create Action dialog, set the action to **Set Connector Name**, and then click **Search**.
5. In the Search dialog, enter **%Compli%**, and then click **Search**.

The percentage symbols (%) are wildcards that search for all values before and after the text **Compli**. This search returns the services that are defined on the Web Services Detail page that contain the text **Compli**.

6. In the Description area, click **Compliance check system conn. . .**, and then click **OK**.
7. On the Create Action dialog, click **OK**.
8. In the DO area, click **AND**, and then click **Perform an Action**.
9. In the Create Action dialog, set the action to **Set Interaction Interface Type**.
10. Set Interaction Interface Type to **SDO**.

If you are a new customer, then you must set Interaction Interface Type to SDO for any interaction interface that you define. Use EBM only for backward compatibility to a prior release.

11. Click **Save and Close**.

## Activate and Publish Your Rule

Activate and publish your rule:

1. On the Manage External Interface Routing Rules page, notice that the Active indicator for Routing Rule for Compliance Check is grey.
2. Right-click **Routing Rule for Compliance Check**, click **Actions**, and then click **Edit**.
3. In the Routing Rule for Compliance Check dialog, add a check mark to the **Activate Rule** option, and then click **Save and Close**.

Notice that the Active indicator for Routing Rule for Compliance Check is green.

4. Click **Publish**.

Order Management publishes each of the active rules that display on the Manage External Interface Routing Rules page.

### Related Topics

- [Routing Requests to Fulfillment Systems: Explained](#)

## Using Visual Information Builder to Manage Pretransformation Rules: Procedure

You can use Visual Information Builder to manage a pretransformation rule that sets the default value of an attribute according to the value of another attribute.

In the example in this topic, you create the following rule:

- If the order type is equal to Government, then set the warehouse to Denver Manufacturing

### Summary of the Work

In this example, you create the following rule:



Use Visual Information Builder to create a pretransformation rule:

1. Create the routing rule and the If condition.
2. Create the Then statement.
3. Activate and publish your rule.

This topic includes example values. You might use different values, depending on your business requirements.

### Create the Routing Rule and IF Condition

Create the routing rule and If condition:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, search for, and then open Manage Pretransformation Rules for Sales Orders.
4. On the Manage Pretransformation Defaulting Rules page, click **Create New Rule**.
5. Change the rule name.

Current Name	New Name
New Rule	Pretransformation Default Rule for Warehouse

6. Name the If statement so that it reflects the meaning of the condition:
  - o Click **Enter Description**.
  - o In the Enter Description dialog, enter **Order Type = Government**, and then click **OK**.
7. Click **New Condition** (the dashed circle in the IF area).
8. In the Create Condition dialog, enter **order type**, wait a moment, and then click **Order Type (Order Header)**. Notice that the dialog displays suggestions when you wait a moment after you finish typing.
9. Accept **is equal to** as the condition, set the order type to Government, and then click **OK**. Notice that the IF area now displays the following condition:
  - o **Order Type is equal to Government**

## Create the THEN Statement

Create the Then statement:

1. Click **THEN**, and then click **DO**.
2. Name the DO so that it reflects the meaning of the action:
  - o Click **Enter Description**.
  - o In the Enter Description dialog, enter **Set warehouse to Denver**, and then click **OK**.
3. Click **New Action** (the dashed circle in the DO area).
4. In the Create Action dialog, enter **ware**, wait a moment, and then choose **Warehouse (Order Fulfill Line)**.
5. In the Create Action dialog, click **Search**.
6. In the Search dialog, enter **Denver**, and then click **Search**.
7. Under Warehouse Name, click **Denver Manufacturing**, and then click **OK**.
8. On the Create Action dialog, click **OK**. Notice that the DO area now displays the following condition:
  - o **Warehouse is set to Denver Manufacturing**
9. Click **Save and Close**.

## Activate and Publish Your Rule

Activate and publish your rule:

1. On the Manage External Interface Routing Rules page, notice that the Active indicator for Pretransformation Default Rule for Warehouse is grey.
2. Right-click **Pretransformation Default Rule for Warehouse**, click **Actions**, and then click **Edit**.
3. In the Pretransformation Default Rule for Warehouse dialog, add a check mark to the **Activate Rule** option, and then click **Save and Close**. Notice that the Active indicator for Pretransformation Default Rule for Warehouse is green.
4. Click **Publish**.

Order Management publishes each of the active rules that display on the Manage Pretransformation Defaulting Rules page.

#### *Related Topics*

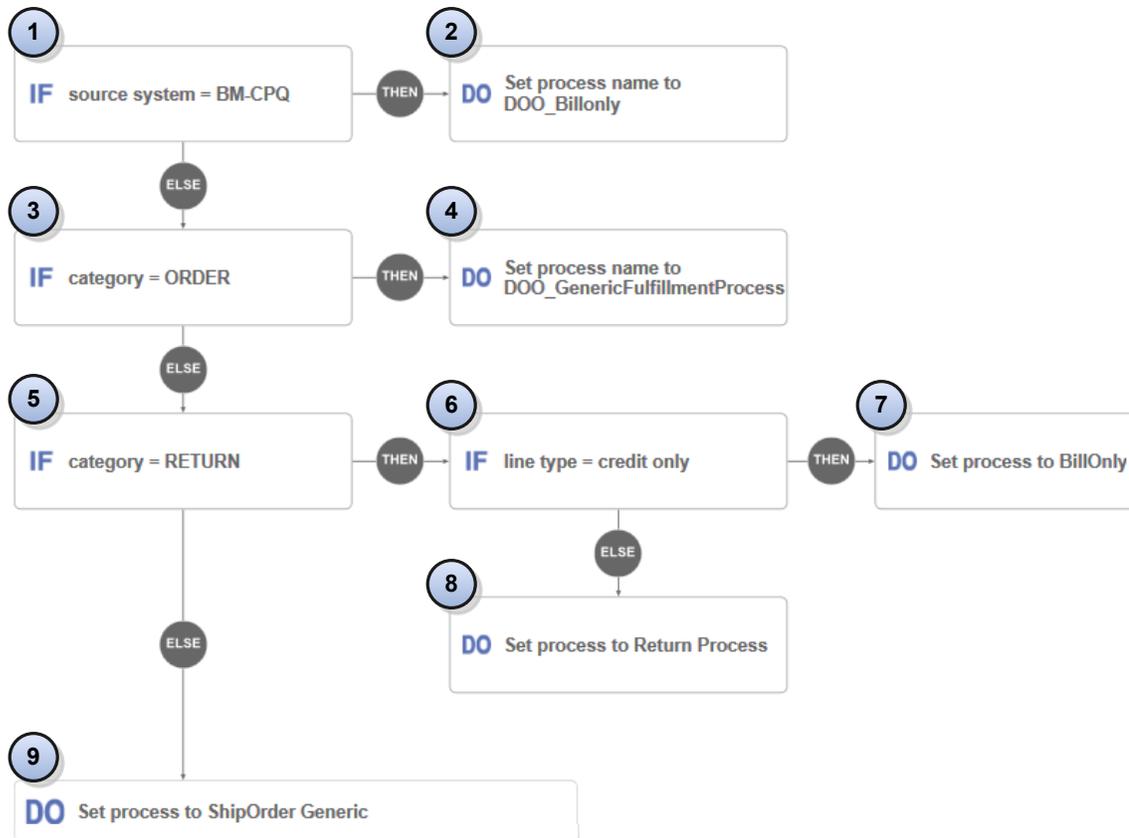
- [Transformation Rules: Explained](#)
- [Transforming Source Orders to Sales Orders: How It Works](#)
- [Transforming Order Lines to Fulfillment Lines: How it Works](#)

## Using Visual Information Builder to Manage Process Assignment Rules: Procedure

You can use Visual Information Builder to assign an orchestration process according to various conditions. For example, you can assign different orchestration processes to handle source orders that Order Management Cloud receives from different source systems, assign them differently depending on whether the order is a new sales order or a return, or assign them depending on the value of an attribute, such as the line type is a credit.

## Summary of the Work

In this example, you create the following rule:



Use Visual Information Builder to manage process assignment rules:

1. Create the routing rule and the first If condition.
2. Create the Then for the first If.
3. Create the second If.
4. Create the Then for the Second If.
5. Create the third If.
6. Create the Then for the Third If.
7. Create the Do that assigns the orchestration process when receiving returns for credit.
8. Create the Do that assigns the orchestration process for all other returns.
9. Create the Do that assigns the orchestration process for all other sales orders.
10. Activate and publish your rule.

This topic includes example values that demonstrate how you can use Visual Information Builder to create a complex process assignment rule. You might use different values, depending on your business requirements.

## Part One Work Instructions

This section includes the detailed steps to perform for the first part of this example.

## Create the Routing Rule and First IF Condition

Create the routing rule and first If condition:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, search for, and then open Manage Process Assignment Rules for Sales Orders.
4. On the Manage Orchestration Process Assignment Rules page, click **Create New Rule**.
5. Change the rule name.

Current Name	New Name
New Rule	Default Process Assignment Rule

6. Name the If statement so that it reflects the meaning of the condition:
  - o Click **Enter Description**.
  - o In the Enter Description dialog, enter `source system = BM-CPQ`, and then click **OK**.  
This IF statement looks for source orders that are coming in from one channel in an integration, named **BM-CPQ** (Big Machines - Configure, Price, and Quote).
7. In the Attributes tab, in the Search field, enter `source order`, and then wait a moment.  
Notice that the dialog displays suggestions when you wait a moment after you finish typing.
8. Click **Source Order System (Order Header)**.  
Notice that the object hierarchy in the Attributes tab highlights the attribute that the search results returned, such as Source Order System.
9. Drag the **Source Order System** attribute from the Attributes tab, and then drop it into the **IF** area.
10. In the Create Condition dialog that displays after you release the mouse button, accept **is equal to** as the condition, and then click the **down arrow** in the list that displays immediately below **is equal to**.  
Notice that the list displays the source systems that are configured for this deployment.
11. Set the source system to **BM\_CPQ**, and then click **OK**.  
Notice that the IF area now displays the following condition:
  - o `Source Order System is equal to BM_CPQ`

## Create the THEN for the First IF

Create the Then statement for the first If:

1. Click **THEN**, and then click **DO**.
2. Name the DO so that it reflects the meaning of the action:
  - o Click **Enter Description**.
  - o In the Enter Description dialog, enter `Set process name to DOO_Billonly`, and then click **OK**.
3. Click **New Action** (the dashed circle in the DO area), and then click **Perform an Action**.
4. In the Create Action dialog, enter `process name`, wait a moment, click **Process Name (Order Fulfill Line)**, and then click **Search**.
5. In the Search dialog, enter `DOO_Bill`, and then click **Search**.
6. In the ProcessName area, click **DOO\_BillonlyGenericProcess**, and then click **OK**.

7. In the Create Action dialog, click **OK**.

Notice that the DO area now displays the following value:

- o `Process Name is set to DOO_BillonlyGenericProcess`

8. Click **Save**.

### Create the Second IF

Create the second If statement:

1. Below the IF area, click **ELSE**, and then click **IF**.
2. Name the If statement so that it reflects the meaning of the condition:
  - o Click **Enter Description**.
  - o In the Enter Description dialog, enter `category = ORDER`, and then click **OK**.
3. In the Attributes tab, notice that the object hierarchy displays as a navigation tree.
4. In the Attributes tab, browse through the object hierarchy:
  - o Click **Attributes**, and then click **Order Header**.
  - o In the Order Header branch, click **Order Line**.
  - o In the Order Line branch, click **Order Fulfill Line**.
  - o In the Order Fulfill Line branch, click **Category**.
  - o Drag the **Category** attribute from the Attributes tab, and then drop it into the **IF** area.
5. In the Create Condition dialog that displays, notice that the condition defaults to **Category is equal to Order**, and then click **OK**.

### Create the THEN for the Second IF

Create the Then statement for the second If:

1. In the area to the right of the **IF category = ORDER** condition, Click **THEN**, and then click **DO**.
2. Name the DO so that it reflects the meaning of the action:
  - o Click **Enter Description**.
  - o In the Enter Description dialog, enter `Set process name to DOO_GenericFulfillmentProcess`, and then click **OK**.
3. Click **New Action** (the dashed circle in the DO area), and then click **Perform an Action**.
4. In the Create Action dialog, enter `process name`, wait a moment, click **Process Name (Order Fulfill Line)**, and then click **Search**.
5. In the Search dialog, enter `doo_`, and then click **Search**.
6. In the ProcessName area, click **DOO\_OrderFulfillmentGenericProcess**, and then click **OK**.
7. In the Create Action dialog, click **OK**.

Notice that the DO area displays the following value:

- o `Process Name is set to DOO_OrderFulfillmentGenericProcess`

8. Click **Save**.

### Create the Third IF

Assign the orchestration process to use when handling a return.

Create the third If statement:

1. Below the **IF category = ORDER** area, click **ELSE**, and then click **IF**.
2. Name the If statement so that it reflects the meaning of the condition:
  - o Click **Enter Description**.
  - o In the Enter Description dialog, enter **category = RETURN**, and then click **OK**.
3. Click **New Condition** (the dashed circle in the IF area).
4. In the Create Condition dialog, enter **category**, wait a moment, and then click **Category (Order Fulfill Line)**.
5. Set the condition to **is equal to Return goods**, and then click **OK**.

Notice that the IF area now displays the following condition:

**Category is equal to Return goods**

### Create the THEN for the Third IF

Returns for different types of fulfillment lines might require Order Management to use different fulfillment processing. In this example, you configure the rule to consider only returns for credit. You create the following statement:

- If Line Type is not empty, and if Line type is equal to Return for credit

Create the THEN statement for the third IF:

1. In the area to the right of the **IF category = RETURN** condition, click **THEN**, and then click **IF**.
2. Name the IF so that it reflects the meaning of the action:
  - o Click **Enter Description**.
  - o In the Enter Description dialog, enter **line type = credit only**, and then click **OK**.
3. Click **New Condition** (the dashed circle in the IF area).
4. In the Create Condition dialog, enter **line type**, wait a moment, and then click **Line Type (Order Fulfill Line)**.
5. Configure the rule to handle an empty line type value:
  - o Set the operator to **is blank**, and then click **OK**.

It is important to include a condition that can handle a situation where the value of the line type is empty. If you do not create this condition, then the rule might fail and cause an error.

Notice that the IF area displays the following value:

**Line Type is blank**

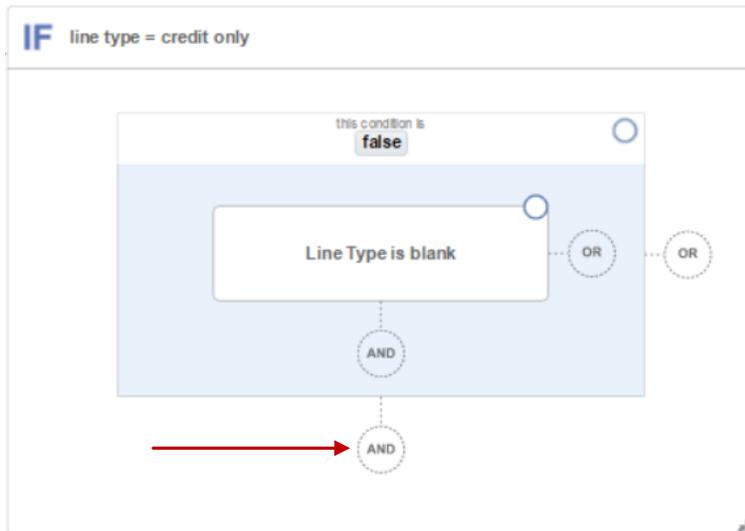
- o In the upper right corner of the **Line Type is blank** area, click **Select** (the circle with the solid line), and then wait for it to display a blue check mark.
- o In the upper left corner of the page, next to the trash can, click **Group**.

Notice that the **IF line type = credit only** area now displays a **this condition is true** area around the **Line Type is blank** area. This grouping is equivalent to adding parenthesis inside your statement. In this example, you group **Line Type is blank** to test whether this statement is true or false. To group a set of statements, you select each statement, and then click **Group**.

- o Click **true**.

Notice that **true** changes to **false**. In this example, when you run the rule, you must make sure the condition **line type is blank** is false because you want Order Management to assign this orchestration process only if the line type contains a value.

6. In the **IF line type = credit only** area, click the **AND** that resides outside of the **this condition is false** area, as indicated by the following arrow:



7. In the Create Condition dialog, enter `line type`, wait a moment, and then click **Line Type (Order Fulfill Line)**.
8. Set the condition to **is equal to Return for credit**, and then click **OK**.

Notice that the IF area now displays the following condition:

- o `Line type is equal to Return for credit`

9. Click **Save**.

## Part Two Work Instructions

This section includes the detailed steps to perform for the second part of this example.

### Create the DO That Assigns the Orchestration Process When Receiving Returns for Credit

Create the Do statement that assigns the orchestration process to use when receiving a return for credit:

1. In the area to the right of the **IF line type = credit only** condition, click **THEN**, and then click **DO**.
2. Name the DO so that it reflects the meaning of the action:
  - o Click **Enter Description**.
  - o In the Enter Description dialog, enter `Set process to BillOnly`, and then click **OK**.
3. Click **New Action** (the dashed circle in the DO area).
4. In the Create Action dialog, enter `process name`, wait a moment, click **Process Name (Order Fulfill Line)**, and then click **Search**.
5. In the Search dialog, enter `%Bill%`, and then click **Search**.

The percentage symbols (%) are wildcards that search for all values before and after the text **Bill**. This search returns the orchestration process names that contain the text **Bill**.

6. In the ProcessName area, click **DOO\_BillOnlyGenericProcess**, and then click **OK**.
7. In the Create Action dialog, click **OK**.

Notice that the DO area displays the following value:

- o `Process Name is set to DOO_BillOnlyGenericProcess`

### Create the DO That Assigns the Orchestration Process for All Other Returns

Create the Do statement that assigns the orchestration process for all other returns:

1. Immediately below the **IF line type = credit only** condition, click **ELSE**, and then click **DO**.
2. Name the DO so that it reflects the meaning of the action:
  - o Click **Enter Description**.
  - o In the Enter Description dialog, enter `Set process to Return Process`, and then click **OK**.
3. Click **New Action** (the dashed circle in the DO area).
4. In the Create Action dialog, enter `process name`, wait a moment, click **Process Name (Order Fulfill Line)**, and then click **Search**.
5. In the Search dialog, enter `%Return%`, and then click **Search**.
6. In the ProcessName area, click **ReturnOrderGenericProcess**, and then click **OK**.
7. In the Create Action dialog, click **OK**.

Notice that the DO area displays the following value:

- o `Process Name is set to ReturnOrderGenericProcess`

8. Click **Save**.

### Create the DO That Assigns the Orchestration Process for All Other Sales Orders

In this section, you assign the orchestration process to use for the sales orders that do not meet the other IF conditions.

Create the Do that assigns the orchestration process for all other sales orders:

1. Immediately below the **IF category = RETURN** condition, click **ELSE**, and then click **DO**.
2. Name the DO so that it reflects the meaning of the action:
  - o Click **Enter Description**.
  - o In the Enter Description dialog, enter `Set process to ShipOrder Generic`, and then click **OK**.
3. Click **New Action** (the dashed circle in the DO area).
4. In the Create Action dialog, enter `process name`, wait a moment, click **Process Name (Order Fulfill Line)**, and then click **Search**.
5. In the Search dialog, enter `ship`, and then click **Search**.
6. In the ProcessName area, click **ShipOrderGenericProcess**, and then click **OK**.
7. In the Create Action dialog, click **OK**.

Notice that the DO area displays the following value:

- o `Process Name is set to ShipOrderGenericProcess`

8. Click **Save**.

### Activate and Publish Your Rule

Activate and publish your rule:

1. On the Manage Orchestration Process Assignment Rules page, notice that the Active indicator for Default Process Assignment Rule is grey.

2. Right-click **Default Process Assignment Rule**, click **Actions**, and then click **Edit**.
3. In the Default Process Assignment Rule dialog, add a check mark to the **Activate Rule** option, and then click **Save and Close**.

Notice that the Active indicator for Default Process Assignment Rule is green.

4. Click **Publish**.

Order Management publishes each of the active rules that display on the Manage Orchestration Process Assignment Rules page.

#### *Related Topics*

- [Assigning Orchestration Processes: Explained](#)
- [Assigning Orchestration Processes: Examples](#)

# 9 Other Implementation Work

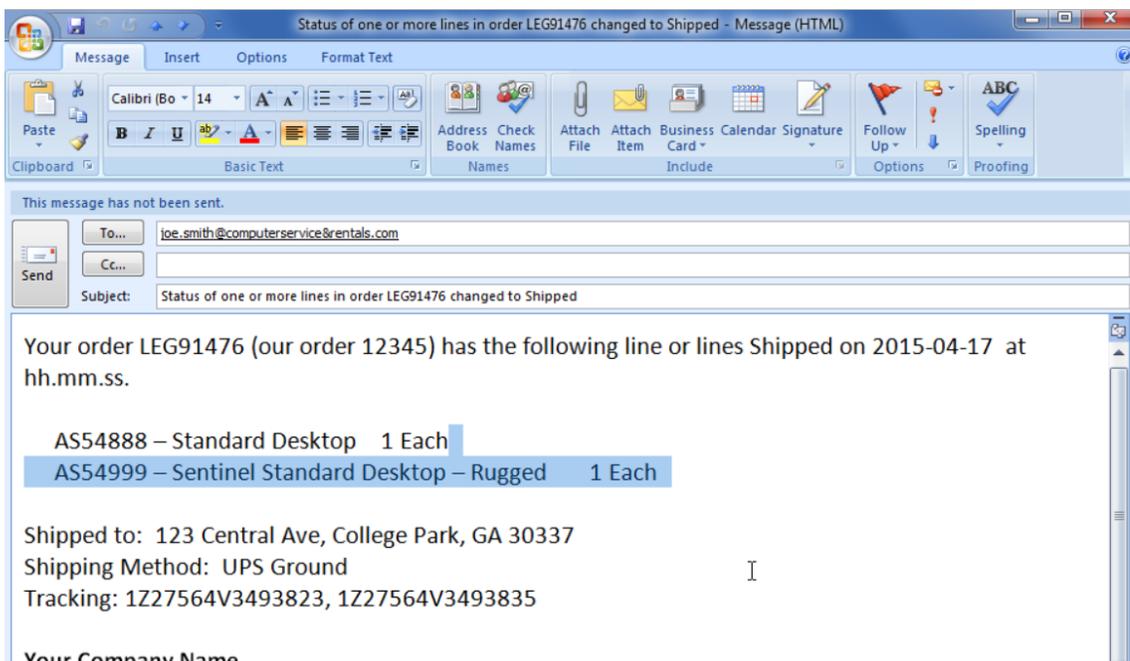
## Administering Messaging and Social Networking

### Administering Email

#### Administering E-mail Notification in Order Management: Explained

You can configure Order Management Cloud to send an e-mail message when a business event occurs, such as when the sales order status changes to Shipped.

The following example illustrates an e-mail format that your deployment can send when the sales order status changes to Shipped. You can modify this format, such as changing the From address, adding your company name and logo, or changing the template message text. For details, see Administering E-mail Formats in Order Management: Explained.



Administer e-mail notification:

1. In the Navigator, click **Setup and Maintenance**.
2. On the Setup and Maintenance page, click **Order Management**, and then click **Setup**.
3. On the Setup page, search for, and then open the Manage Business Event Trigger Points page.
4. Optional. Send an e-mail notification when the sales order goes into hold:
  - o On the Manage Business Event Trigger Points page, click the **Hold** row, and then make sure the **Active** option in this row contains a check mark.

- o In the Details area, add a check mark to the **Send E-mail Notification** option.

**5.** Optional. Send an e-mail notification when the order header status updates:

- o On the Manage Business Event Trigger Points page, click the **Order Header Status Update** row, and then make sure the **Active** option in this row contains a check mark.
- o In the Details area, add a check mark to the **Raise Event** option and the **Send E-mail Notification** option for each Status Value, as necessary. For example, to configure Order Management to send an e-mail notification when the order status changes to Closed, add a check mark to the options in the Closed row.

The Send E-mail Notification option depends on the event, so you must make sure you add a check mark to each option.

- o Repeat this step for other status values, as necessary.

## Administer E-mail Notification for Status Updates on Fulfillment Lines

Administering Order Management to send an e-mail notification when the status updates on a fulfillment line requires that you modify the orchestration process definition.

Administer e-mail notification for status updates on fulfillment lines:

1. On the Manage Business Event Trigger Points page, click the **Fulfillment Line Status Update** row, and then make sure the **Active** option in this row contains a check mark.
2. Click **Save and Close**.
3. On the Setup page, search for, and then open Manage Orchestration Process Definitions.
4. On the Manage Orchestration Process Definitions page, search for ShipOrderGenericProcess.

Each orchestration process controls the status value for each fulfillment line, so you must modify the orchestration process that controls the status value. In this example, you modify the orchestration process that controls the shipping status value.

5. In the Search Results, click the **row** that contains ShipOrderGenericProcess, click **Actions**, and then click **Edit**.
6. In the Process Details area, click **Status Conditions**, click **Fulfillment Line Status Values**, and then click **Edit Status Rule Set**.
7. On the Edit Status Rule Set page, add a check mark to the **Notify External System** option and the **Send E-mail Notification** option for each Status Value where you must send a notification.

For example, to send an e-mail notification when the fulfillment line status changes to Shipped, add a check mark to the options in the Shipped row.

The Send E-mail Notification option depends on the Notify External Systems event, so you must make sure you add a check mark to each option.

8. Repeat step 7 for other status value, as necessary.

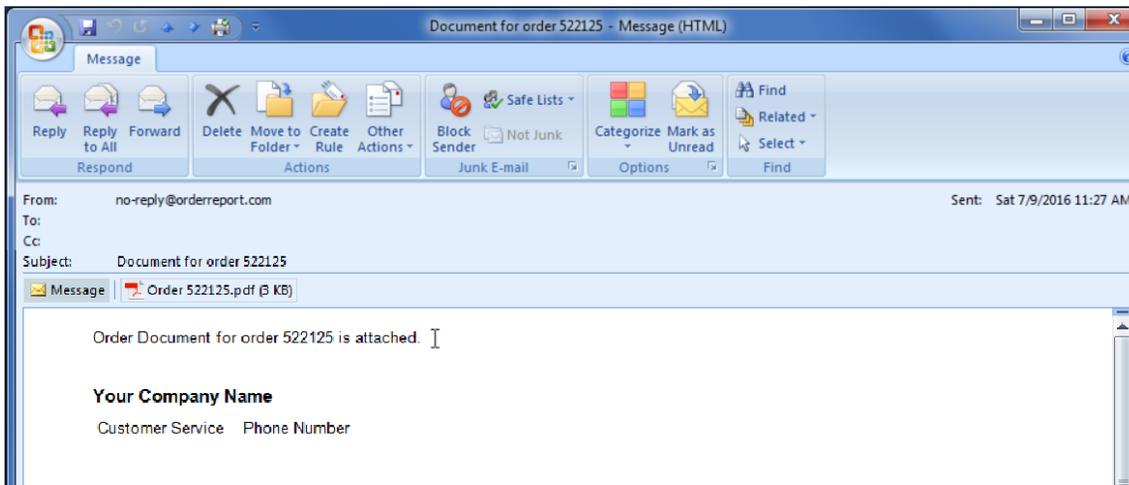
Order management will send an e-mail notification when each fulfillment line that references this orchestration process definition reaches the status that you specify in steps 7 and 8. Order Management sends this e-mail to the customer contact and to the ship-to contact that the sales order references.

9. Repeat steps 4 through 8 for each orchestration process definition in your deployment that updates status values.

## Administering E-mail Formats in Order Management: Explained

The Order Entry Specialist can click Actions, Create Document, and then Send E-mail from the sales order header to send order details through an e-mail message. Order Management Cloud sends order details in a PDF (Portable Document Format) file that it attaches to this e-mail.

Order Management comes predefined to send an e-mail message that resembles the following format. You can use Oracle BI Publisher to modify this format, such as adding your company name and logo, or changing the template message text.



You can use the order management parameter named From Address for E-mail Messages to specify the value that Order Management displays in the e-mail From address. For details, see [Managing Order Management Parameters: Procedure](#).

#### Related Topics

- [Managing Order Management Parameters: Procedure](#)
- [Using Oracle BI Publisher Enterprise to Modify Templates for Use with Formats: Explained](#)

## Administering Collaboration Messaging

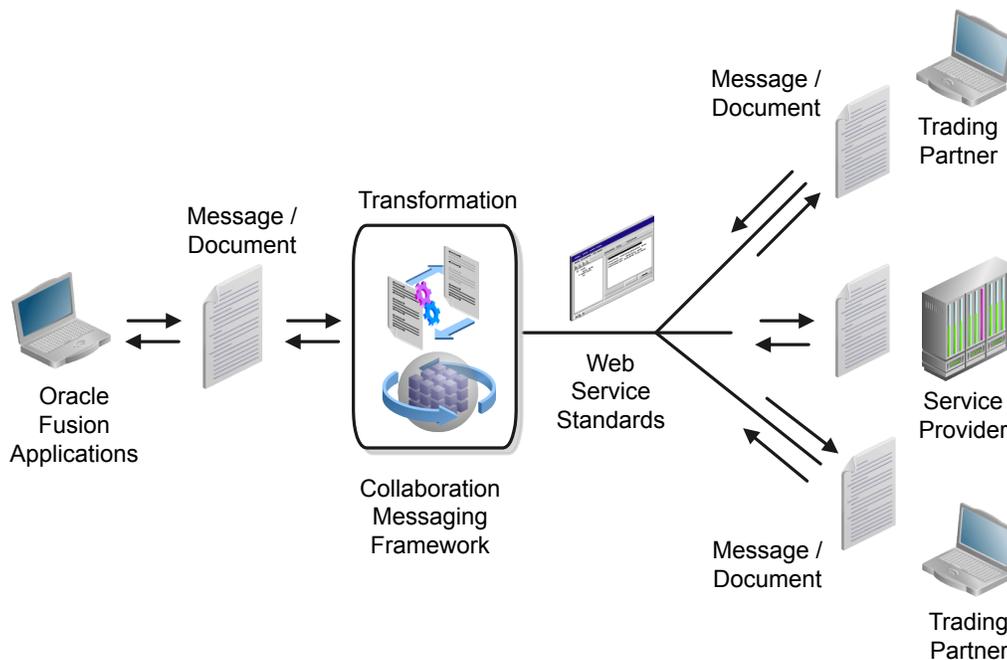
### Using Collaboration Messaging: Overview

Use Oracle Fusion Collaboration Messaging Framework to enable Oracle Fusion applications establish business-to-business (B2B) messaging exchanging capabilities with trading partners.

Using this framework, you can send and receive real-time transactional messages without building new SOA components. You can leverage the existing B2B functionality to exchange messages with collaborators such as suppliers either directly or using an intermediary agency such as a B2B Service Provider.

The framework supports transformation of a B2B document, such as a purchase order, between the Oracle Fusion Applications format and an external message format supported by the trading partner. When you send messages to partners or receive messages from them, the framework performs the required transformation.

The following figure illustrates how Collaboration Messaging Framework delivers a message to the intended recipient.



Using collaboration messaging involves performing the following high-level tasks:

- Setting up external (B2B) trading partners and their messaging capabilities.
- Cross-referencing the Oracle Fusion applications definition of a trading partner (such as a supplier) with the external trading partner definition set up earlier. Also, selecting the messages that must be enabled with the partner.
- Configuring the message delivery method for the partner.

To open the Collaboration Messaging Framework Overview page, click the Navigator menu and select **Collaboration Messaging**.

Use the Collaboration Messaging Framework Overview page to:

- Manage undelivered collaboration messages
- Reprocess failed collaboration messages
- Manage collaboration messaging history
- Validate inbound and outbound collaboration messaging setup

## Reprocessing Undelivered Messages: Procedure

Any inbound or outbound message that isn't processed because of some error, remains undelivered. You can view the undelivered messages on the Collaboration Messaging Framework Overview page. For each undelivered message, you can diagnose the errors, take corrective action, and resubmit a request to deliver it again.

To reprocess an undelivered messages:

1. On the Collaboration Messaging Framework Overview page, click the tasks icon to view the tasks, and select the **Manage Undelivered Collaboration Messages** task.

2. On the Manage Undelivered Collaboration Messages page, search for the undelivered message. The message is listed in a table.
3. Click the message row to view the reason for delivery failure. The details appear under Processing History.

 **Tip:** Click the message ID link to view the setup details of the message.

4. Take the required corrective action and return to the **Collaboration Messaging Framework Overview** page.
5. Select the specific message and click **Reprocess**. If there are no further problems, the message is submitted for delivery.

## Process Inbound Collaboration Messages

### Validating Inbound Collaboration Messaging: Procedure

After you set up an application partner, such as a supplier site, you can send a test inbound message to verify if the setup is appropriate for messaging. Use the Validate Inbound Collaboration Messaging Setup task on the Collaboration Messaging Overview page to validate an inbound message.

All messages that go through the validation process queue up and appear on the Collaboration Messaging History page. There you can examine the details of each processed message to check if it was transformed and processed as intended.

1. On the Overview page, click the tasks icon to view the tasks, and select the **Validate Inbound Collaboration Messaging Setup** task.
2. Select the service provider and the partner ID from whom you expect to receive the collaboration message.
3. Enter the external message ID.
4. Select the details pertaining to the External Message Type. The related details, such as document type and messaging standard, appear.
5. Click **Create Message Payload**. The message payload is generated in XML format and appears in the text box.
6. Update the generated payload or replace it with the XML payload that you want to test.
7. Examine the elements of the message payload. The following table contains an example of the mapping between the elements and actual data.

XML Element	Corresponding Information
<Sender>	The ID of the partner who sent the document.
<Intermediary>	Contains the ID of the service provider.
<Receiver>	Contains the ID of the recipient.
<BODID>	An ID that the sender assigns to the message.

8. Click **Process**. The Processing Confirmation message appears.
9. Click **View Collaboration Message** to view the processed message.
10. Click **Done**.

To view the processed message again, search for it on the Manage Collaboration Messaging History page. In the search results, click the generated message ID to view its details.

 **Tip:** If the message processing fails, you can view the reason for it on the Manage Failed Collaboration Messages page.

## Process Outbound Collaboration Messages

### Validating Outbound Collaboration Messaging: Procedure

After you set up an application partner, such as a supplier site, you can send a test outbound message to verify if the setup is appropriate for messaging. Use the **Validate Outbound Collaboration Messaging Setup** task on the Collaboration Messaging Framework Overview page to validate an outbound collaboration message.

All messages that go through the validation process queue up and appear on the Collaboration Messaging History page. There you can examine the details of each processed message to check if it was transformed and processed as intended.

1. On the Collaboration Messaging Framework Overview page, click the tasks icon to view the tasks, and select the **Validate Outbound Collaboration Messaging Setup** task.
2. On the Validate Outbound Collaboration Messaging Setup page, select the relevant document. The related details appear.
3. Select the supplier. The related details, such as the supplier site and service provider appear.
4. Click **Create Message Payload**. The message payload is generated in XML format and appears in the text box.
5. Update the generated payload or replace it with the XML payload that you want to test.
6. Click **Process**. The generated message ID appears on the page.
7. Click **View Collaboration Message** to view the processed message.
8. Click **Done**.

To view the processed message again, search for it on the **Manage Collaboration Messaging History** page. In the search results, click the message ID to view its details.

 **Tip:** If the message processing fails, you can view the reason for it on the **Manage Failed Collaboration Messages** page.

## Manage Collaboration Messages

### Creating a Trading Partner with a Service Provider: Procedure

A service provider is an intermediary for exchanging messages between Oracle Fusion applications and trading partner. Whenever you set up a trading partner, you can link it with a service provider.

 **Note:** You must be signed in as a supplier and must have access to the Supplier task.

1. On the Supplier Overview page, navigate to the Supplier Business Classifications section and click the supplier.
2. On the Edit Supplier page, switch to the Sites tab and click the required site.
3. On the Edit Site page, select Collaboration Messaging Framework as the channel for B2B Communication.
4. Under Associated Collaboration Documents, click Manage Trading Partners.
5. On the Manage Trading Partners dialog box, click **Create - Create Trading Partner with Service Provider**.
6. On the Create Trading Partner dialog box, fill the required details.
7. Click **Actions - Add Row** and fill the details to associate collaboration documents with the trading partner. You may add multiple documents.

 **Note:** To enable exchange of messages, you must set the collaboration document status to **Active**.

8. Click Save and Close.
9. On the Manage Trading Partners dialog box, click **OK**.
10. On the Edit Site page, click **Save and Close**.
11. On the Edit Supplier page, click **Save and Close**.

To update the details for a trading partner, use the edit option on the Manage Trading Partners dialog box.

## Creating a Trading Partner without a Service Provider: Procedure

You can exchange messages with a trading partner directly, without using a service provider.

 **Note:** You must be signed in as a supplier and must have access to the Supplier task.

1. On the Supplier Overview page, navigate to the Supplier Business Classifications section and click the supplier.
2. On the Edit Supplier page, switch to the Sites tab and click the required site.
3. On the Edit Site page, select Collaboration Messaging Framework as the channel for B2B Communication.
4. Under Associated Collaboration Documents, click Manage Trading Partners.
5. On the Manage Trading Partners dialog box, click **Create - Create Trading Partner without Service Provider**.
6. On the Create Trading Partner dialog box, fill the required details.

 **Note:** By default, the Service Provider is set to None because this setup doesn't involve a service provider.

7. Click **Actions - Add Row** and fill the delivery method, outbound collaboration message, and inbound collaboration message details on the respective tabs.

 **Note:** The following table contains some tips on filling the important information on each tab.

Tab	Details
Delivery Methods	<ul style="list-style-type: none"> <li>○ Provide the Certificate Alias Name associated with the security policy</li> <li>○ Provide the endpoint URL and the associated authentication credentials to initiate the collaboration messaging web service. It must be in the format <code>http://&lt;server&gt;:&lt;port&gt;/&lt;context&gt;</code>, where <b>&lt;context&gt;</b> contains the name of the web service as defined in the application.</li> </ul>
Outbound Collaboration Messages	<ul style="list-style-type: none"> <li>○ Specify a unique name and select the collaboration message. The associated details automatically appear in the row.</li> <li>○ Select a delivery method.</li> <li>○ Set the outbound collaboration document status to <b>Active</b>.</li> </ul>
Inbound Collaboration Messages	<ul style="list-style-type: none"> <li>○ Specify a unique name and select the collaboration message. The associated details automatically appear in the row.</li> <li>○ Select a delivery method.</li> <li>○ Set the inbound collaboration document status to <b>Active</b>.</li> <li>○ Specify an XPath that identifies the application partner in the inbound collaboration document.</li> </ul>

8. Click **Save and Close**.
9. On the Manage Trading Partners dialog box, click **OK**.
10. On the Edit Site page, click **Save and Close**.
11. On the Edit Supplier page, click **Save and Close**.

To update the details for a trading partner, use the edit option on the Manage Trading Partners dialog box.

## Managing Associated Collaboration Documents: Procedure

To set up collaboration messaging, you must associate the supplier site with a trading partner, and select the documents you want to exchange with that partner. The documents that you set up here are associated with trading partners or the service providers of those trading partners.

 **Note:** You must be signed in as a supplier and must have access to the Supplier task.

1. On the Supplier Overview page, navigate to the Supplier Business Classifications section and click the supplier link.
2. On the Edit Supplier page, switch to the Sites tab and click the required site.
3. On the Edit Site page, select Collaboration Messaging Framework as the channel for B2B Communication.
4. Under Associated Collaboration Documents, click **Edit**.
5. On the Edit Associated Collaboration Documents dialog box, click **Add Row** and fill the details required to set up the document. The read-only particulars appear based on the selected details.
6. Click **Save** and repeat the steps to add more documents or click **Save and Close** to return to the previous page.
7. On the Edit Site page, click **Save and Close**.
8. On the Edit Supplier page, click **Save and Close**.

## FAQs for Define Collaboration Messaging

### What are the different undelivered collaboration message error statuses?

The following table describes the main differences among the various error statuses.

Error	B2B Error	Hold
Indicates that the messages haven't been delivered because of a validation, configuration, or processing error in Collaboration Messaging Framework.	Indicates that the B2B component of the SOA suite couldn't deliver the message because of a configuration or processing error.	Indicates that messages haven't been processed because an administrator has put them on hold.

### What are the different undelivered collaboration message error types?

The following table describes the main differences among the message error types.

Document Retrieval Error	Inbound Processing Error	Outbound Processing Error
Occurs when the collaboration messaging framework fails to retrieve the document associated with a collaboration event.	Occurs when the collaboration messaging framework can't process inbound messages because of setup or business rule validation issues.	Occurs when the collaboration messaging framework can't process outbound messages because of setup or business rule validation issues.

## What happens if I don't enable a document type for storage?

If you don't enable a document type for storage, the message processing and delivery details of such documents aren't stored in the log table.

## Why did my message fail?

To know the cause of a message failure, search for the failed message on the Manage Failed Collaboration Messages page.

When you click the message row, the cause of the failure appears under Processing History.

## Configuring Collaboration Messaging for a Customer: Procedure

Using this task, you can associate a customer account with an existing trading partner and select the collaboration messaging documents to be exchanged with the customer.

To configure collaboration messaging for a customer:

1. On the Collaboration Messaging Overview page, click the Tasks icon, and select the **Manage Customer Collaboration Configuration** task.
2. Search for the customer account, select the row, and click **Edit Collaboration Configuration**.
3. Under Associated Service Providers, click **Actions - Add Row** and fill the details of the service provider and the trading partner.
4. Click **Actions - Add Row** and select at least one collaboration document.
5. Set the **Association Status** to Active to enable messaging with the selected service provider.
6. Click **Save and Close**.

## Administering Social Conversations in Order Management: Procedure

You can enable Oracle Social Network in Order Management Cloud so that your user community can engage in social conversations about sales orders.

Administer social conversations for Order Management:

1. Navigate to the Manage Oracle Social Network Objects page.
2. In the Name column, expand **Order Management**, and then click **Sales Order**.
3. Click **Enable Object**.
4. In the Enable Object dialog, choose one of the following options, and then click **OK**.

Option	Description
Manual	Create social conversations only for conversations that the Order Entry Specialist manually initiates for each sales order. It is strongly recommended that you choose Manual.
Automatic	Create a social conversation for every sales order. In most situations, it is not necessary to choose Automatic.

5. In the Attributes area, click **Actions**, and then click **New**.
6. In the Select Attributes dialog, add a check mark to the **Enabled** option for each attribute that Order Management must include in the conversation.

Order Management will stamp each attribute and the value for each attribute that you enable into the social conversation thread. It will also stamp the current attribute value every time the conversation updates.

*Related Topics*

- [Social Networking: Overview](#)
- [Manage Social Networking : Overview](#)

# Glossary

## **action**

The kind of access, such as view or edit, named in a security policy.

## **activity**

An event that occurs outside of Order Management Cloud. For example, an orchestration process might include an activity task type to configure a network router. An activity contains the details needed to complete the task. A user can perform an activity as part of completing order fulfillment. Order Management can assign an activity to a user.

## **advance shipment notice**

A notification that a delivery is pending.

## **application partner**

A B2B trading partner defined in Oracle Applications Cloud that engages in the collaboration messaging activity. For example, a supplier site, or a customer account.

## **assignment rule**

A rule that assigns an orchestration process to fulfillment lines.

## **assignment set**

A group of sourcing rules, bills of distribution, or both, in which each rule or bill is assigned to an assignment level, and to attribute values for the attributes applicable to that assignment level. An assignment set defines a supply chain.

## **attribute**

The property of a sales order, order line, fulfillment line, or orchestration process. Customer and Ship-To Address are each an example of an attribute of a sales order. Quantity is an example of an attribute of an order line. Actual Start Date is an example of an attribute of an orchestration process. Jeopardy Score is an example of an attribute of a fulfillment line.

## **blanket purchase agreement**

A long term agreement for the purchase of goods and services from a supplier. It includes terms and conditions, details of the goods or services to be purchased from the supplier, and negotiated amounts. The blanket agreement does not contain delivery dates, individual delivery quantities, or amounts. Complete details necessary to supplier fulfillment are specified subsequently in purchase orders issued against the agreement.

### **bucket**

A container that defines a range of values.

### **bucket set**

A container that holds the overall range of values that a group of buckets defines.

### **business event**

An occurrence in an application or system that might be significant to objects in some other application or system. Applying a hold on a sales order is an example of a business event.

### **business event trigger point**

A condition that causes a business event to occur.

### **business object**

A resource in an enterprise database, such as an invoice or purchase order.

### **business unit**

A unit of an enterprise that performs one or many business functions that can be rolled up in a management hierarchy.

### **client business unit**

A business unit that can requisition and process invoices for a supplier site. One or more procurement business units can service a client business unit.

### **compensation pattern**

A rule that you can set on an orchestration process step that specifies the adjustments to make when an order changes. Undo, Redo, Update, Cancel, and None are each an example of a compensation pattern. For example, assume the compensation pattern for a Create Shipment step is Redo, and that this step calls the Cancel service and the Create service. If Order Management receives a change order that includes a new warehouse for this step, then it runs the Cancel service and the Create service again.

### **configuration model**

A hierarchical structure that specifies how to represent a configured item. A desktop computer that includes a monitor, keyboard, mouse, and external speakers is an example of a configuration model. In this example, the desktop computer is the parent item, and the peripherals are the child items. Oracle Fusion Configurator specifies the structure, which includes referenced models, configurator rules, user interface specifications and so on.

### **Configurator**

A feature that allows you to configure a configured item. The Configurator allows you to choose configuration options for an item from a list of available options. It uses constraints to make sure the items that you choose fit together correctly. It reduces configuration errors, which in turn reduces change order processing and any rework that is required during the order fulfillment process.

### **configured item**

An item that includes one or more options. A desktop computer where you choose the hard drive, monitor, and mouse is an example of a configured item.

### **connector**

Configuration templates that include mapping and protocol settings of the target PLM system to be integrated with Oracle Fusion PLM

### **constraint entity**

The business object or orchestration process that a processing constraint constrains.

### **cost of change**

A numerical value that measures how much a change impacts an orchestration process. For example, the monetary cost to a company, or the difficulty that is associated with incorporating the change.

### **cross-reference**

A reference that relates business data between an order capture system, order fulfillment system, and Order Management Cloud. For example, an item cross-reference can create a relationship between an item that resides in a source system, such as Widget A, and a representation of this same item in Order Management. Cross-references help to manage the representation of data across systems.

### **customer**

A customer is someone with whom you have a selling relationship. The selling relationship can result from the purchase of products and services, or from the negotiation of terms and conditions that provide the basis for future purchases.

### **descriptive flexfield**

Customizable expansion space, such as fields used to capture additional descriptive information or attributes about an entity, such as a customer case. You may configure information collection and storage based on the context.

### **drop shipment**

A type of shipment where the shipper is not the seller. This shipper ships directly to the customer. The shipper for an internal drop shipment is another business unit. The shipper for an external drop shipment is an external supplier.

### **duty role**

A group of function and data privileges representing one duty of a job. Duty roles are specific to applications, stored in the policy store, and shared within an application instance.

### **extensible flexfield**

Customizable expansion space used to capture multiple sets of information within a context or multiple contexts. Some extensible flexfields let you group contexts into categories.

### **external partner**

A B2B partner engaging in the collaboration messaging activity but defined outside Oracle Applications Cloud.

### **external system or external application**

A system or application that is external to and not part of Order Management. An order capture system that resides upstream of Order Management is an example of an external system. A fulfillment application that resides downstream of Order Management is an example of an external application.

### **flexfield**

A flexible data field that you can customize to contain one or more segments or store additional information. Each segment has a value and a meaning.

### **flexfield segment**

An extensible data field that represents an attribute and captures a value corresponding to a predefined, single extension column in the database. A segment appears globally or based on a context of other captured information.

### **fulfillment line**

A request to fulfill an item. A fulfillment line contains information about this request, such as customer, sales order number, item, quantity, scheduled ship date, actual ship date, shipping method, and so on. You can schedule a fulfillment line, reserve product for it, substitute an item for it, change the warehouse, change the shipping method, change the demand class, and so on.

### **fulfillment system**

A system that resides downstream of Order Management Cloud that is responsible for fulfilling a sales order. For example, a fulfillment system is responsible for shipping the physical goods to the customer, such as shipping a laptop computer from a warehouse to a customer location.

### **fulfillment task**

A type of task that Order Management Cloud performs to fulfill a sales order. Schedule and Ship are each an example of a fulfillment task. Order Management might use multiple orchestration process steps to complete a single fulfillment task.

### **hold**

A temporary stoppage of processing an order line or fulfillment line during order fulfillment.

### **hybrid configuration model**

A type of configuration model that includes an assemble-to-order model as the child and a purchase-to-order model as the parent. Order Management Cloud uses one blanket purchase agreement to source the assemble-to-order components, and a different blanket purchase agreement to source the purchase-to-order components.

### **import file**

A flat file that you use as an interface between your source system and Order Management. You copy source orders into this file, and then use a scheduled process to import orders from the import file into Order Management.

### **interface table**

A database table that stores data during data transfer between applications or from an external system or data file.

### **internal identifier**

An identifier that Order Management uses to create a reference between a lookup that you specify for an item, customer, or reference data, to information from the Product Information Manager work area, Trading Community Architecture, or collection data. This data resides in the Order Orchestration and Planning Repository. For example, if your order uses Payment Terms, then you must set up this value in Oracle, and then set up collections processing.

### **inventory organization**

A logical or physical entity in the enterprise that tracks inventory transactions and balances, stores definitions of items, and manufactures or distributes products.

### **item**

A product that resides in the Product master database. Order Management displays items in the Item column when you add an order line to a sales order. For example, items for a company that sells hardware might include nuts, bolts, and screws. Items for a company that sells computers might include hard drives, computer monitors, and desktop computers. Items for a wireless service provider might include service plans, international calling services, and instant messaging services. Items for an insurance company might include an item that applies for a period of time, such as fire insurance coverage.

### **item organization**

Item definition where inventory balances are not stored and movement of inventory is not tracked in the applications. Item attributes that carry financial and accounting information are hidden.

### **item validation organization**

An inventory organization whose primary or secondary unit of measure is used as the costing unit of measure for the item in the cost organization to which that inventory organization belongs. The item master organization can also be designated as the item validation organization.

### **jeopardy**

Measures the level of risk that is associated with the delay of an orchestration process task as low, medium, or high.

### **jeopardy priority**

A value that describes the level of risk that is associated with delaying a fulfillment task. Order Management displays a jeopardy priority value of Low, Medium, or High.

### **jeopardy score**

A numeric ranking that indicates the severity of a delay in completing a fulfillment task

### **jeopardy threshold**

The range of time that a task is delayed.

### **job role**

A role, such as an accounts payable manager or application implementation consultant, that usually identifies and aggregates the duties or responsibilities that make up the job.

### **lead-time**

The amount of time that Order Management requires to complete an orchestration process step, including wait steps and pause steps.

### **line selection rule**

A type of rule that determines whether or not an orchestration process step runs a fulfillment line. For example, a line selection rule can specify not to ship a nonshippable item, such as a contract that a customer completes online.

### **Navigator**

The menu in the global area that you can use to open the work areas and dashboards that you have access to.

### **offering**

A comprehensive grouping of business functions, such as Sales or Product Management, that is delivered as a unit to support one or more business processes.

### **Oracle BI Publisher**

An Oracle application that performs the following formatting tasks for Oracle Fusion Payments: 1) formats extracted data into a message, such as a settlement batch or payment file, that can be understood by the payment system, 2) supports remittance advice formatting and delivery.

### **orchestration planning**

The process of orchestrating and planning the fulfillment of a sales order. For example, planning dates, planning shipments, and so on. Replanning is the process of updating an orchestration plan for a sales order to accommodate a change that occurs to this sales order or in the fulfillment environment. For example, when a customer changes the quantity of an existing sales order.

### **orchestration process**

A process that includes a sequence of steps. These steps process one or more fulfillment lines during order fulfillment. An orchestration process coordinates the orchestration of physical goods and activities in a single order, and it automates order orchestration across fulfillment systems. It contains the instructions that describe how to process an order, such as the steps and services to use, step dependencies, conditional branching, lead-time information, how to handle change orders, and the status values to use. It describes how to schedule, reserve, ship, return, and invoice a sales order.

### **orchestration process branch**

A path in an orchestration process that Order Management runs when a condition is met.

### **orchestration process class**

A set of statuses that you can assign to an orchestration process.

### **orchestration process step**

A step of an orchestration process that specifies the task layer service that Order Management calls or the subprocess to start. Each step references a task type, task, and service. A step might also specify branching. A task includes one or more orchestration process steps.

### **orchestration subprocess**

A set of sequential steps. This set of steps is an orchestration process that another orchestration process references.

### **order compensation**

The process of modifying an order so that it accurately reflects the changes that have occurred to the order. For example, assume Order Management receives a change order that specifies to ship an item from a different warehouse. In this example, it adjusts the order so that this order uses the different warehouse, updates delivery dates so that these dates accurately reflect the time required to ship the item from the different warehouse, and so on.

### **order fulfillment**

The part of the order management lifecycle that fulfills a sales order. Order fulfillment typically begins after the user submits a sales order in Order Management Cloud. Order Management schedules, ships, and bills a sales order during order fulfillment. Order Management communicates with a downstream fulfillment system during order fulfillment.

### **order import template**

A template that you can use to structure your source data so that you can import it into Order Management.

### **order orchestration and planning data repository**

The set of data collected from source systems and stored for use by order orchestration, order promising, and supply chain planning processes. Also known as the planning data repository.

### **order transformation**

The process of transforming a source order into the business objects that define a sales order. These business objects include the order, order lines, fulfillment lines, orchestration processes, and return fulfillment lines.

### **order validation**

The process that Order Management performs to make sure a sales order meets a set of criteria. For example, to make sure the sales order includes information for all required fields and other information according to various predefined configurations and custom configurations that you define.

---

**order-to-cash**

The front-to-back flow of information that Order Management Cloud uses to process a sales order through various systems and applications, from order entry to order fulfillment.

**pause task**

A task that temporarily stops an orchestration process from running so that it can wait for a condition to be met.

**predefined**

An object, integration, or configuration that comes already defined with an Oracle application and is ready to use with little or no modification.

**privilege**

A grant of access to functions and data; a single, real world action on a single business object.

**processing constraint**

A rule that controls who can change a sales order, what can change in this sales order, and when the change can occur.

**requisition**

Document generated for a department or an individual to notify the purchasing department of goods or services it needs to order, their quantity, and the expected delivery time frame.

**requisitioning business unit (BU)**

The business unit that manages and owns the requisitioning transaction.

**role**

Controls access to application functions and data.

**routing rule**

A rule that determines how to route data through a system or application. It typically uses an IF-THEN structure that specifies a condition and an action to take depending on the result of the condition. For example, an external interface routing rule can specify how to route a fulfillment request to a fulfillment system depending on the current state of the fulfillment environment, such as available inventory, transit time, and so on.

**sales order**

A contractual document between a sales organization and their customer to deliver items. It might reference a customer purchase order.

### **scheduled process**

A program that you run to process data and, in some cases, generate output as a report.

### **selling business unit**

A business unit that is a profit center that resides in the selling legal entity. If you specify All Business Units as the selling business unit, then the application applies the buy and sells terms to all the profit center business units that the selling legal entity contains.

### **service provider**

An intermediary that facilitates exchange of messages between Oracle Applications Cloud and external partners.

### **set**

Classified and grouped reference data that organizational entities share.

### **shipment set**

A set of order lines that Order Management ships together as one group. All of these order lines ship and arrive on the same date, although they might be spread across more than one package, depending on packing requirements.

### **source data**

Data that you import from an external system into Order Management.

### **source order**

An order that you create in Order Management before you click Submit, or an order that Order Management receives from a source system. Order Management transforms a source order to a sales order when you click Submit, or when you import it from a source system.

### **source system**

System where the sales order was created. Order Management Cloud and an order capture system are each an example of a source system. A source system provides business application information to an Oracle application. Oracle can use this information to extract fulfillment data and planning data into data files.

### **sourcing rule**

A specification of the means by which organizations can replenish items.

### **split fulfillment line**

A supply chain technique that fulfills an order in a timely and efficient way. For example, assume a fulfillment line includes a quantity of 30 items, but the preferred warehouse, Warehouse A, only has 20 items in stock. Warehouse B has 15 items in stock. To avoid a delay, you can split the fulfillment line so that it gets the items it requires from both warehouses.

### **stamp**

To impress a snapshot of data onto an object. For example, a sales order might contain a set of attribute values that exist on a specific date, such as quantity equals four and color equals blue on October 9, 2016. These values might change at some point, such as during order fulfillment. A stamp impresses the values of this sales order from a specific point in time

onto an object, such as a social conversation, so that it provides an audit trail that you can use to communicate the values of attributes that existed in an historical context.

### **start-after condition**

Rules that govern whether the Pause task establishes a pause and how the pause can be released.

### **status condition**

A rule that determines when a sales order or object reaches a status. The following rule is an example of a status condition: If the Schedule task status equals Scheduled, then set the orchestration process status to Scheduled.

### **subscriber**

A system or application that subscribes to a web service or business event. For example, a source system can subscribe to a business event that sends notifications about changes to sales orders. If a change occurs, then a web service sends information about this event to this source system in the web service payload. This configuration is similar to a subscription service that regularly sends information to subscribers, or that sends information when an event occurs.

### **substitution**

Replacing one item with some other item in a sales order. For example, replacing an 80GB hard drive with a 100GB hard drive because there are no 80GB hard drives in inventory.

### **task type**

A group of services that Order Management uses to perform a fulfillment task. It represents a common business function that Order Management uses so that it can process a sales order from the time that a user creates it to the time that Order Management sends this sales order to an order fulfillment system. Shipment is an example of a task type. It is a set of services that communicates with a shipping fulfillment system to ship the items that a fulfillment line references.

### **transformation rule**

A rule that transforms a source order that resides on your source system to a sales order that resides in Order Management.

### **user credential key**

A user and password combination that authenticates communication between Order Management and an external fulfillment system.