



Oracle CPQ – Oracle Order Management Integration Guide

Standard Process Update 23B and Later

April 2024

Copyright © 2024, Oracle and/or its affiliates

Table of Contents

Introduction	4
Purpose	5
Audience	5
Prerequisites	5
Acronym List	5
Terminology	6
Order Flow Overview	6
Create Order Process Flow	6
Cancel Order Process Flow	7
Renewal Subscription Process Flow	7
OIC Integration Installation and Setup	8
Installing Business Accelerator (Update 24A or later)	8
Import Integration Package (Update 23B, 23C, or 23D)	15
OIC Mapping Details (Reference Material)	19
Oracle CPQ Package Installation and Setup	26
CPQ Integration Center Setup	26
Create an OIC Integration	26
Create a Get Sales Order Status from FOM Integration	27
Create a Cancel a Sales Order from CPQ Integration	28
Import CPQ-Oracle Order Management Package	30
Enable Subscription Ordering for Simple Products (Optional)	31
CPQ Standard Process Elements	31
Commerce Integrations	32
Commerce Attributes	34
Commerce Actions	34
Library Functions	35
Validation Rules	36
Hiding Rules	37
Order Management Setup	38
Enable Business Events	38
Manage Trigger Points for Business Events	38
Send Status Updates for Fulfillment Lines	38
Send Status Updates for Subscription Lines	40
Setting Up Order Management Event Subscriptions	41
Get the CSF Key	41
Register the CSF Key in Order Management	42
Oracle CPQ Field Setup	43
Business Unit ID Field	43
Account Fields	43
Part Subscription Product Type Field	43
Layout Fields	44
Demo Product Setup (Optional)	45
Import the Demo Migration Package	45
Deploy Product Families	45
Deploy the Home Page	46
Install the Parts Package	47
Import BOM Data Tables	47
Activate and Map Data Tables	48
Verify BOM Products	48
Order Management Pricing Integration	49
Enable Pricing	49
Pricing Related Utility BML	49
Prices Array Set	49
Oracle CPQ Account Integration	51
Library Functions	51
Manual Data Table Changes	52
Add Template Dependencies to File Manager	53

Appendix A: Workflows	54
Appendix A1: Create Order – Standard Item Workflow.....	54
Appendix A2: Create Order – Configurable Item Workflow	55
Appendix A3: Auto Sync Status Workflow	56
Appendix A4: Full Order Cancellation.....	57
Appendix A5 Partial Order Cancellation.....	58
Appendix A6: Terminate Subscription Product Workflow	58
Appendix A7: Renewal Subscription Workflow	59
Appendix B: BML	61
BML – Process Create Order Response.....	61
BML - Get Account Details	61
Appendix C: Payload Template Files	65
findOrganizationPayload.txt	65
customerAccountPayload.txt.....	65
Appendix D: Miscellaneous Commerce Library Functions	66
String getTemplateLocation(String system, String operation).....	66
String invokeWebService(String system, String soapReq)	66
Appendix E: CPQ-OM Status Mapping	67
Map CPQ Line Status to Order Management Line Status.....	67
Appendix E1: OIC Integration – UpdateSOSStatusFromFOM/ FOM CPQ SalesOrderStatusSync	69
Appendix E2: OIC Integration – UpdateSubStatusFromOSS/ OSS CPQ AssetUpsertFlow	70
Appendix F: Retrieve OIC Integration Endpoint URL	71
Appendix G: Limitations and Troubleshooting	74
Appendix G1: Limitations.....	74
Appendix G2: Troubleshooting	74
Resolve Issues with Create Order Action.....	74
Assets Not Created in CPQ Assets Repository.....	74
Blank Line Status on click of 'Get SO Status from FOM' button.....	74

Revision History

This document will continue to evolve as existing sections change and new information is added.

All updates appear in the following table:

Date	What's Changed	Notes
MAR 2024	Introduction Renewal Subscription Process Flow Activate the OSS CPQ CreateRenewalQuote integration OIC Mapping Details Appendix A7: Renewal Subscription Workflow	Oracle CPQ 24B updates.
JAN 2024	Introduction OIC Integration Installation and Setup Create a Get Sales Order Status from FOM Integration Create a Cancel a Sales Order from CPQ Integration Appendix E: CPQ-OM Status Mapping Appendix F: Retrieve OIC Integration Endpoint URL	Oracle CPQ 24A updates.
SEPT 2023	Introduction Import Integration Package OIC Mapping Details (Reference Material) Commerce Integrations Appendix B: BMLIntroduction	Oracle CPQ 23D updates including support for rate plans.
AUG 2023	Commerce Integrations	Oracle CPQ 23C updates to Process Create Order Response (BML) section.
JUN 2023	Oracle CPQ 23C Updates	Charge Entity Support for CPQ Standard Process - Order Management Integrations
APR 12 2023	Initial Release	Oracle CPQ 23B content.

Introduction

This integration includes:

- REST-based Integration with Oracle Order Management using Oracle Integration Cloud (OIC)
- Support for pre-priced order creation and cancellation
- Asynchronous update of quote order and line status from Oracle Order Management
- The CPQ Standard Commerce Process Integration includes artifacts for CPQ and Oracle Order Management such as, Commerce Integration instance, Attributes, Actions, and associated logic.

Oracle CPQ Update 23B introduced the Commerce Standard Process with pre-defined quoting, ordering, integration, and other workflow entities. This provides a standard quote object for customers as a well-defined and functional starting point for their implementation.

- When you create a new Standard Process, it is automatically populated with standardized logic for documents, attributes, actions, rules, library functions, layouts, data columns, Commerce integrations, and workflow steps.
- The new Standard Process simplifies administrator set up for not just the Commerce Process in CPQ, it also has prebuilt artifacts for the unified Oracle end-to-end integrations flows with Oracle CX Sales and Oracle Order Management
- Beginning in Oracle CPQ 23B, the Standard Process is delivered for new Oracle CX Sales integrated site installations and is the default process when creating a new Commerce process.
- The Standard Quote Process Commerce process, included with Oracle CPQ RefApp from Update 23B onwards, is based on the new Commerce Standard Process.

Customers currently using an Oracle CPQ and Oracle Order Management integration from an update prior to Oracle CPQ 23B that want to upgrade to the Oracle CPQ 23B or later platform release, can continue to use the CPQ-Oracle Order Management Package (CPQ_FOM_Package_21B.zip) and OIC integration package (CPQFOMIntegrations_21B.par).

Oracle CPQ delivers a new CPQ-Oracle Order Management Package (CPQ_FOM_Package_<release_version>.zip) in every release from Oracle CPQ 23B onwards to support the new Commerce Standard Process released in that specific Oracle CPQ platform release.

For customers already implementing the new Commerce Standard Process (available from Oracle CPQ 23B platform release onwards) and upgrading to the latest Oracle CPQ platform release, import of the release specific CPQ-Oracle Order Management Package (CPQ_FOM_Package_<release_version>.zip) is optional. However post-upgrade if there is a need to import the CPQ-Oracle Order Management Package again, then it is recommended to use the release specific CPQ-Oracle Order Management Package that is consistent with your platform release.

Customers implementing the new Commerce Standard Process for the first time, **must** import the release specific CPQ-Oracle Order Management Package (CPQ_FOM_Package_<release_version>.zip) that is consistent with your platform release and then optionally merge their customization manually after careful planning and testing during implementation.

From Oracle CPQ 24A onwards, customers must install the new business accelerator (*Oracle CPQ - Oracle Order Management | Quote and Order Management*) available in the integration store.

Purpose

This installation guide describes how to implement the reference integration between Oracle CPQ and Oracle Order Management using order management REST APIs and OIC.

- This guide is only relevant to customers implementing the new Commerce Standard Process in Oracle CPQ 23B platform or later release. Customers can either use the *Oracle Quote to Order* Commerce process, included with the Oracle CPQ 23B RefApp or later release, or they can create a new standard process to get all the benefits of the new Commerce Standard Process.
- Customers that upgrade to the Oracle CPQ 23B platform or later release who want to integrate CPQ and Oracle Order Management using their existing Commerce process from a release prior to Oracle CPQ 23B, should refer to the [21B Oracle CPQ – Oracle Order Management Integration Guide](#).

Audience

This installation guide is intended for administrators responsible for setting up and configuring the Oracle CPQ - Order Management solution. This guide assumes administrators have prior Oracle CPQ, Oracle Order Management, and OIC administration experience.

Prerequisites

Administrators must integrate the Oracle CPQ 23B or later Standard Commerce Process with the following:

- ABO Implementation Package 23B or later
- Oracle Order Management 23B or later using OIC 20.3.3.0.0 or later middleware, which is used to establish an integration between Oracle CPQ and Oracle Order Management.
- Customer Data Management (CDM) Integration which supports account integration.
- Users can create products in the Oracle CX Sales Product Model and synchronize the products into CPQ. Likewise, they can create parts in Oracle CPQ and synchronize them into the Oracle CX Sales Product Model.

Notes:

- Administrators performing the integration installation must have administrator privileges on the Oracle CPQ, Oracle Order Management, and OIC sites.
- For information about obtaining any of the above prerequisites, contact [My Oracle Support](#).

Acronym List

Definitions of the acronyms used within this document are provided in the following table. For additional information, refer to the Oracle CPQ Administration Online Help.

- **BML** (Big Machines Extensible Language) - A scripting tool used to capture a company's complex business logic within Oracle CPQ configuration and commerce.
- **BOM** (Bill of Material) - Fulfillment systems often maintain BOMs containing complex, multi-level part structures that differ from the configuration attributes used in Oracle CPQ when sales users configure products. BOM mapping provides a data-driven mechanism for mapping these differing product views. To use the Order Management solution to create an order from an Oracle CPQ transaction, the products must be modeled as a BOM.
- **CPQ** (Configure, Price, Quote) - This Oracle solution enables companies to streamline their entire opportunity-to-quote-to-order process, including product selection, configuration, pricing, quoting, ordering, and approval workflows.
- **OM** (Order Management) - Oracle Order Management, previously known as Fusion Order management, is designed to improve order capture and fulfillment execution across the quote to cash process by providing a central order hub for multi-channel environments. The application provides the ability to capture, price and configure orders through direct order entry.
- **OIC** (Oracle Integration Cloud) - Oracle Integration Cloud Service is a cloud-based integration application designed to perform integrations between cloud-based applications – but also has capabilities that extend beyond that, to performing integrations with your on-premises applications.

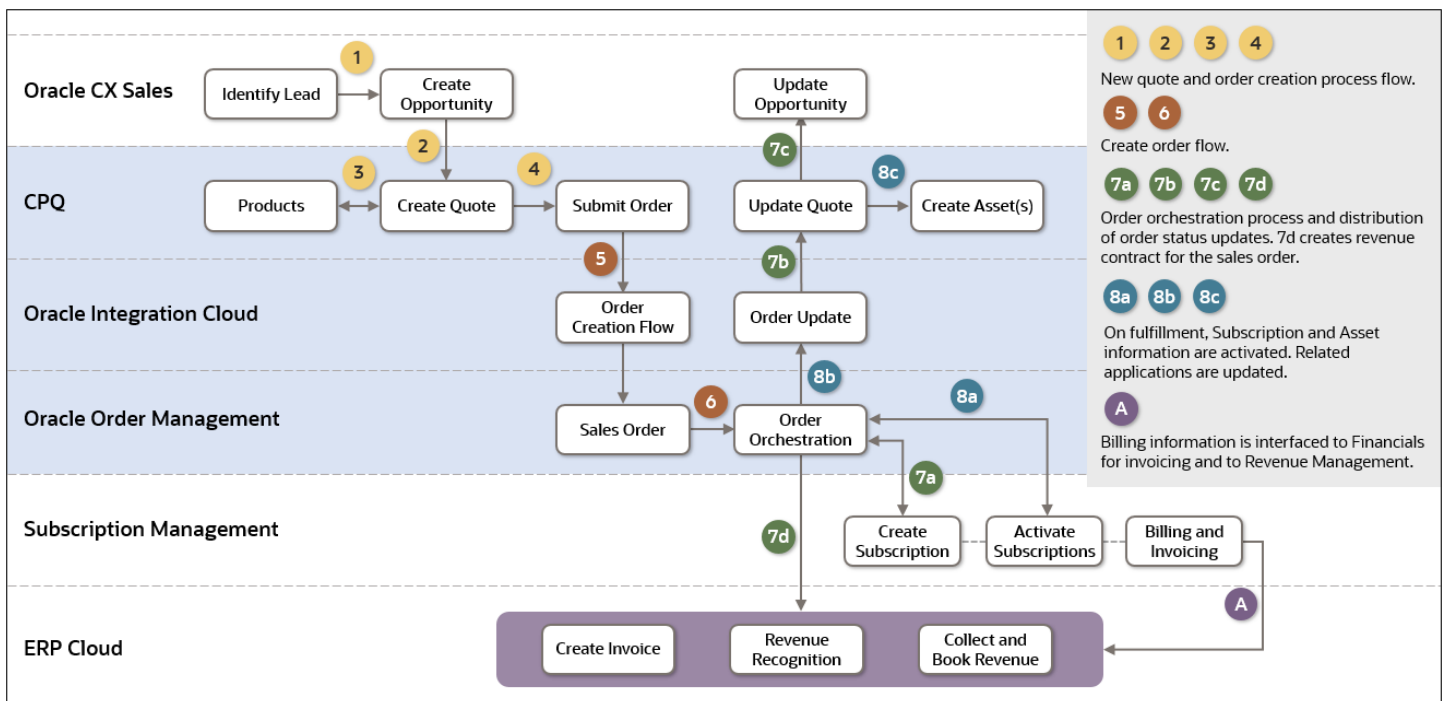
Terminology

- **Commerce Process** – an administrative object to create templates for selling processes.
- **Sales Transaction** - a runtime instance set up for quoting, ordering, approval, and other workflow processes.
- **Quote** – General term referring to a sales transaction
- **Standard Commerce Process / Standard Process / Standard Quote Process** - All refer to the standardized seeded Commerce Process that is available to all CPQ environments beginning with the Oracle CPQ Update 23B when creating a new Commerce Process.

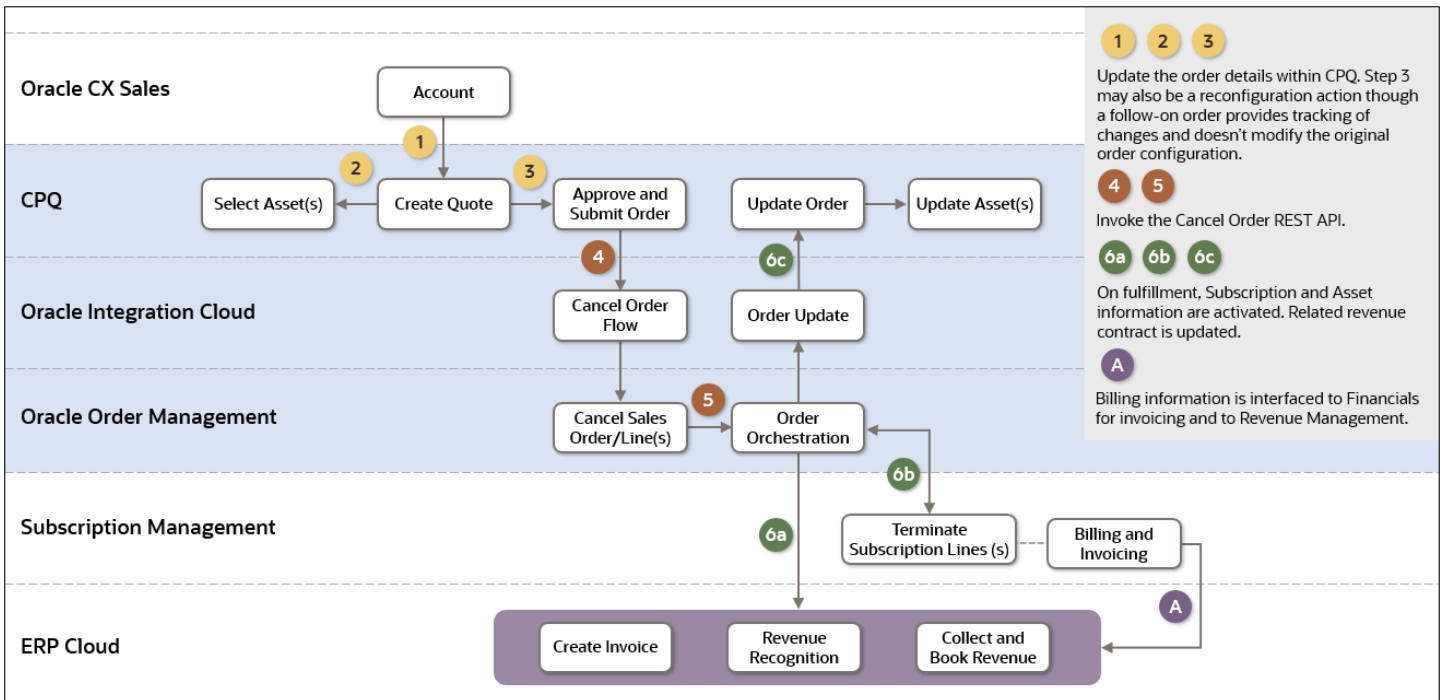
Order Flow Overview

The following images show Oracle end-to-end solution for order process flows. This integration focuses on the CPQ, Oracle Integration Cloud, and Oracle Order Management elements of this solution.

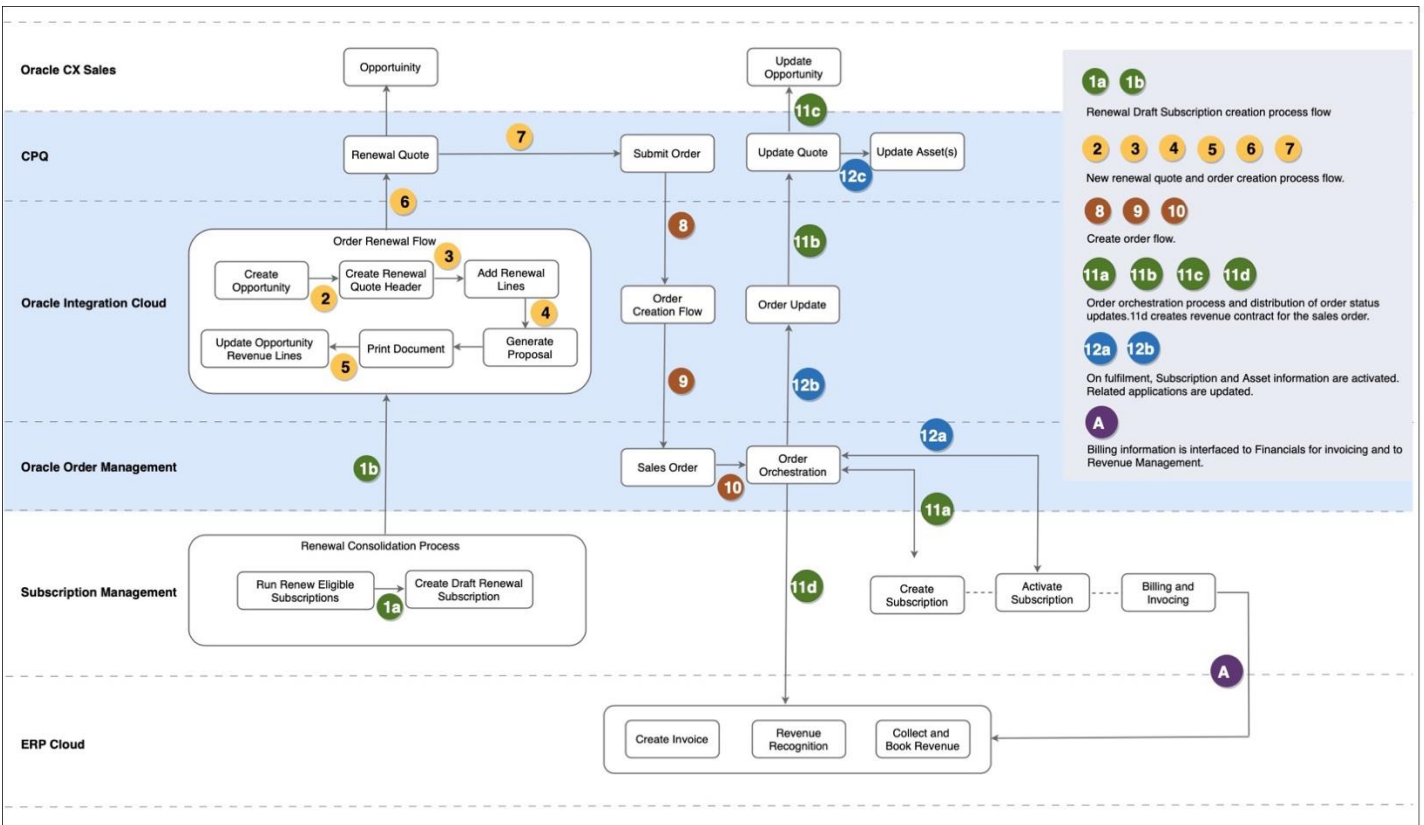
Create Order Process Flow



Cancel Order Process Flow



Renewal Subscription Process Flow



OIC Integration Installation and Setup

This section contains information about installing the new OIC business accelerator from integration store into your OIC environment and creating web service connections between Oracle CPQ and Oracle Order Management in Update 24B.

This section also contains information about importing the OIC Integration package into your OIC environment and creating web service connections between Oracle CPQ and Oracle Order Management in Update 23B, 23C, or 23D.

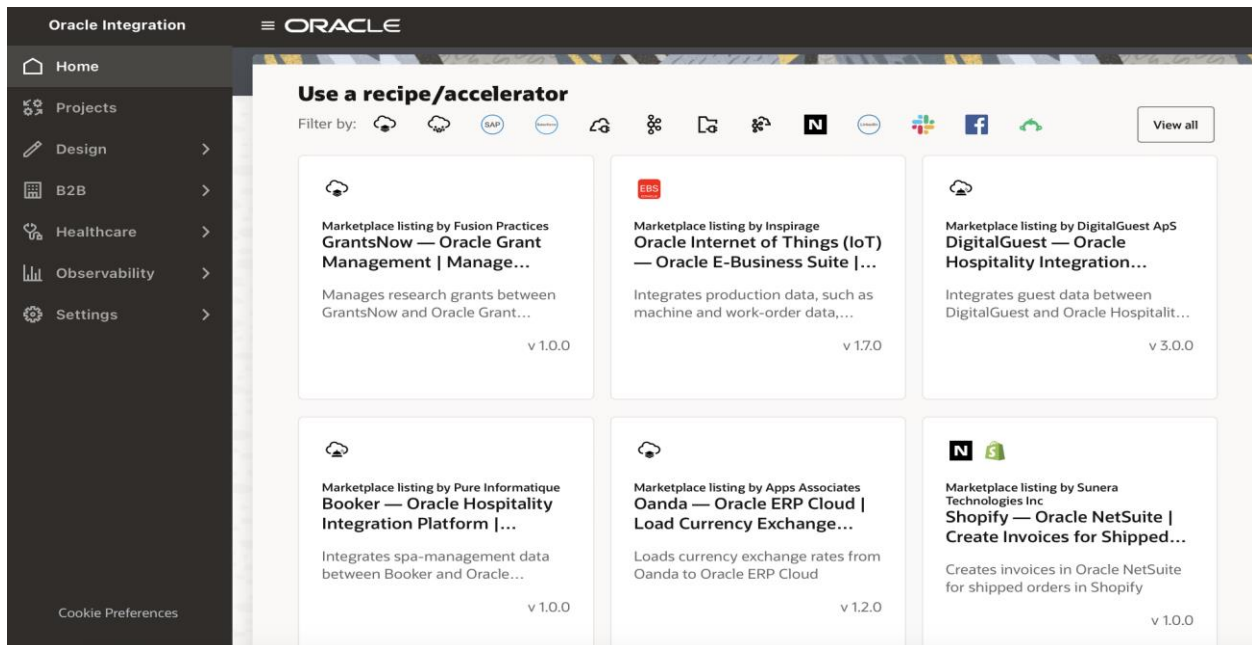
Prerequisites

- The SSL certificate of OIC environment is installed in Oracle Order Management for ERP event subscription.
- All required SSL certificates (CPQ and Order Management SSL certificates) are in the OIC environment.

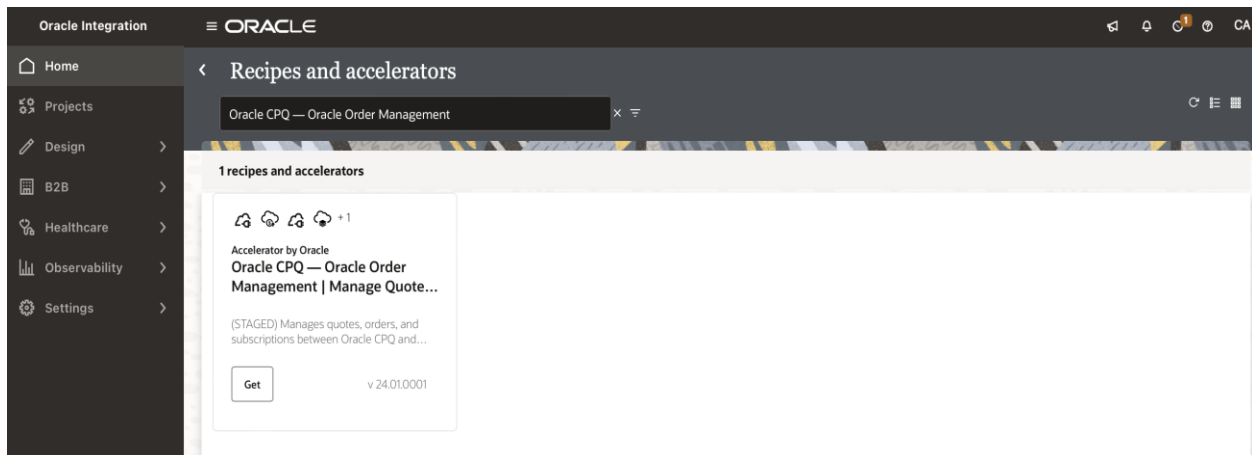
Installing Business Accelerator (Update 24A or later)

Perform the following steps to install the business accelerator into OIC.

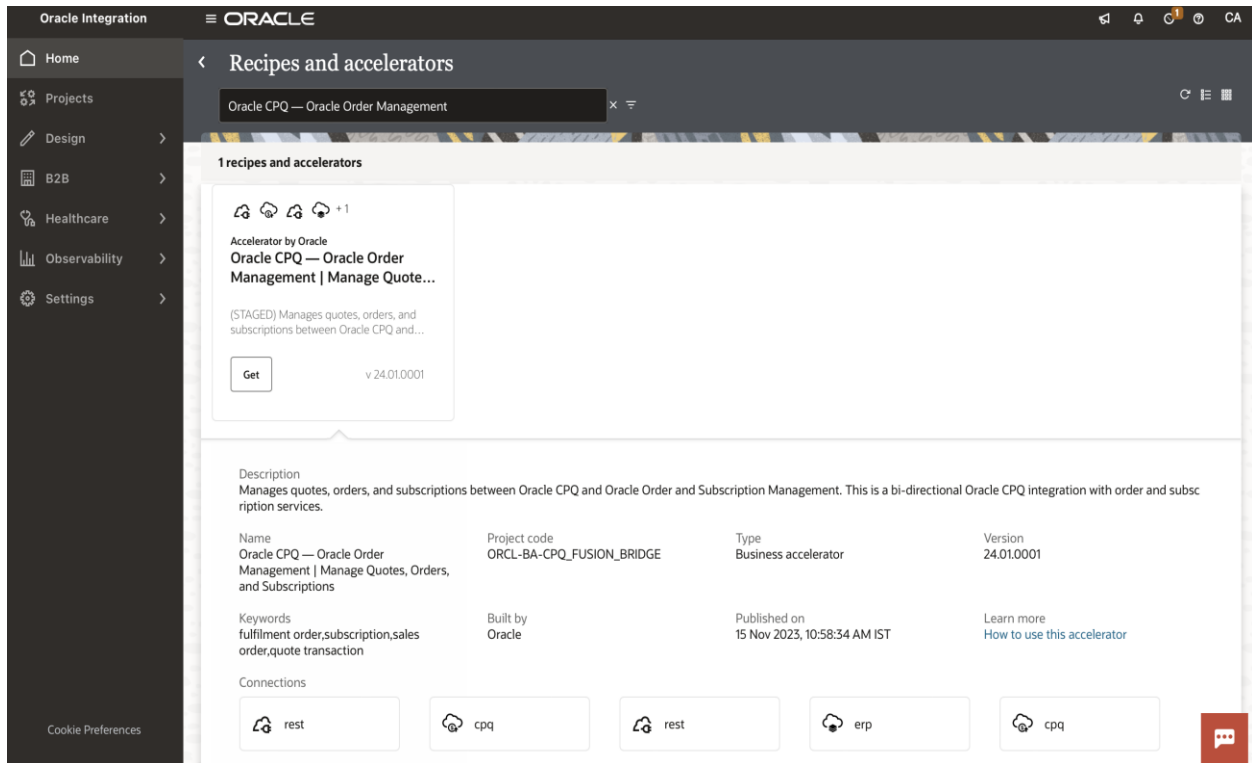
1. Log in to the OIC site as an administration user.
2. Select Home in the left side navigation panel.
3. On the Home page, in the **Use a recipe/accelerator** section, click **View all**.



4. Use the **Search** option and enter “Oracle CPQ — Oracle Order Management”.

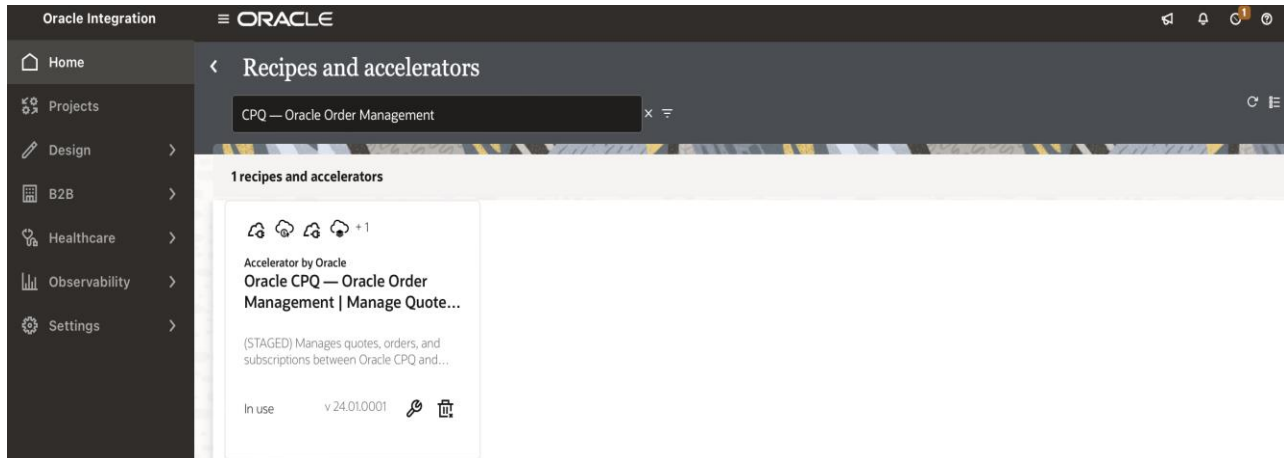


5. Hover over the card and click to verify the following business accelerator name and project code is displayed:
 - Name: Oracle CPQ —Oracle Order Management | Manage Quotes, Orders, and Subscriptions
 - Project Code: ORCL-BA-CPQ_FUSION_BRIDGE



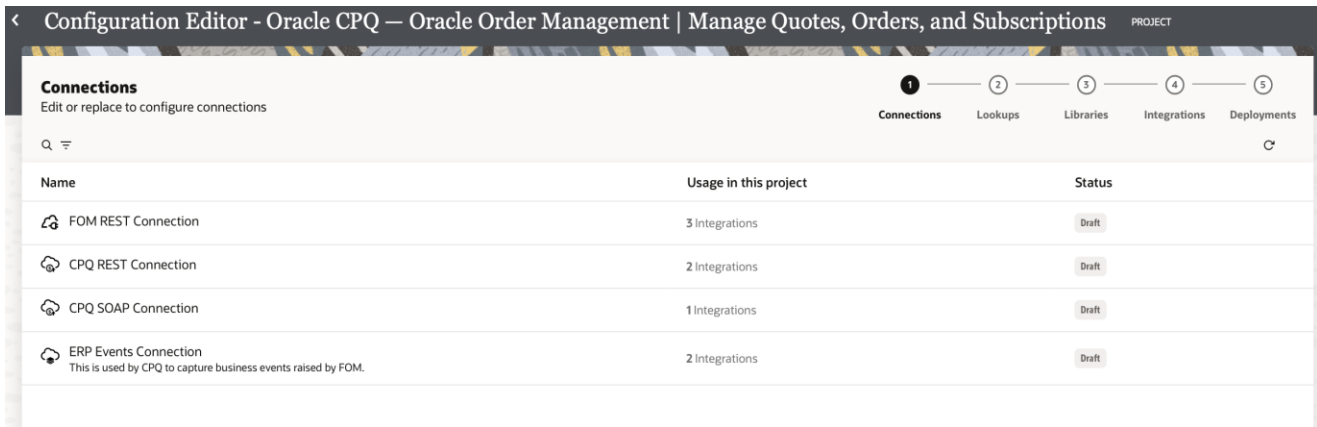
6. Hover over the card and click on **Get** to install the business accelerator.

A message displays “Getting Oracle CPQ — Oracle Order Management | Manage Quotes, Orders, and Subscriptions.” Wait for few seconds and then click Refresh. The accelerator will be installed and the card displays **In use**, **Configure** and **Uninstall** options.

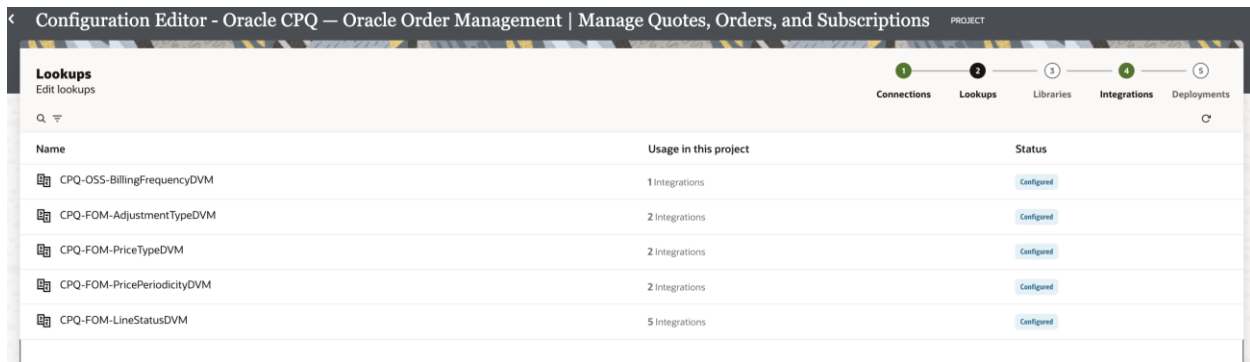


Note: The installation of accelerator installs the project **Oracle CPQ Fusion Bridge**. This is a single pane canvas to manage Integrations, Connections, Lookups, and Libraries.

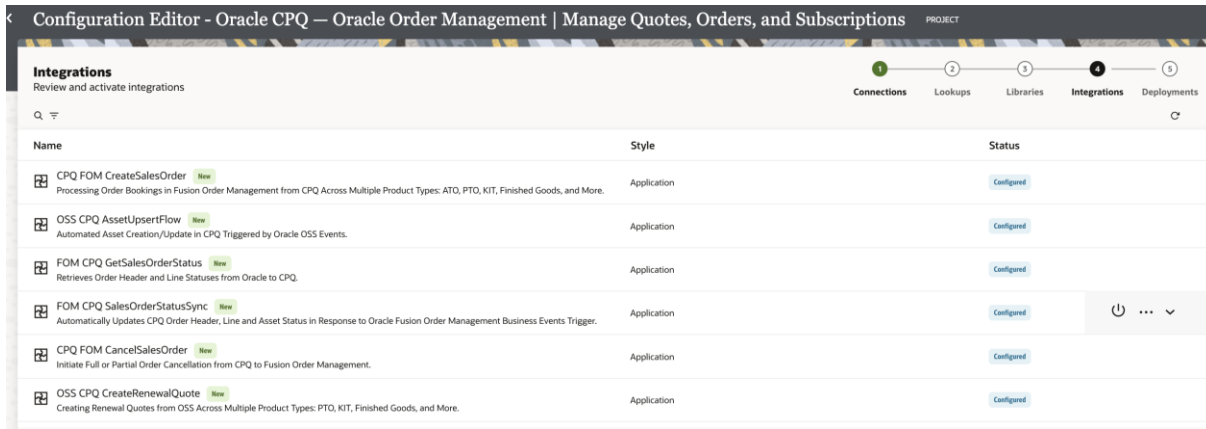
7. Click on **Configure**, a Configuration Editor opens with five train stops:
 - Connections
 - Lookups
 - Libraries
 - Integrations
 - Deployments
8. Select **Connections** train stop and verify the following connections are imported:
 - FOM REST Connection
 - CPQ REST Connection
 - CPQ SOAP Connection
 - ERP Events Connection



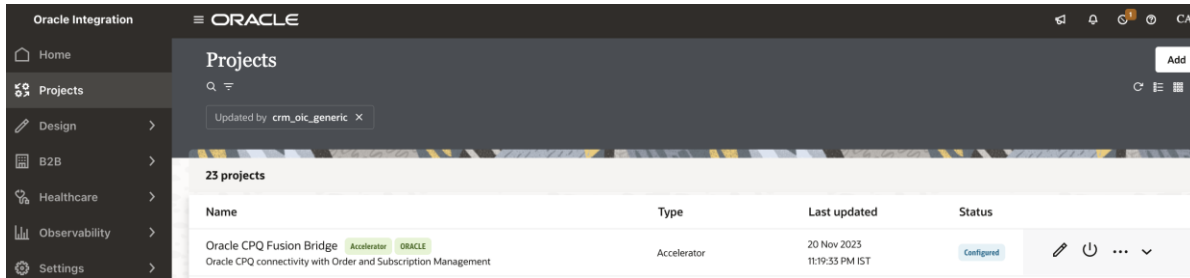
9. Select **Lookups** train stop and verify the following lookups are imported:
 - CPQ-FOM-AdjustmentTypeDVM
 - CPQ-FOM-LineStatusDVM
 - CPQ-FOM-PricePeriodicityDVM
 - CPQ-FOM-PriceTypeDVM
 - CPQ-OSS-BillingFrequencyDVM (Update 24B or later)



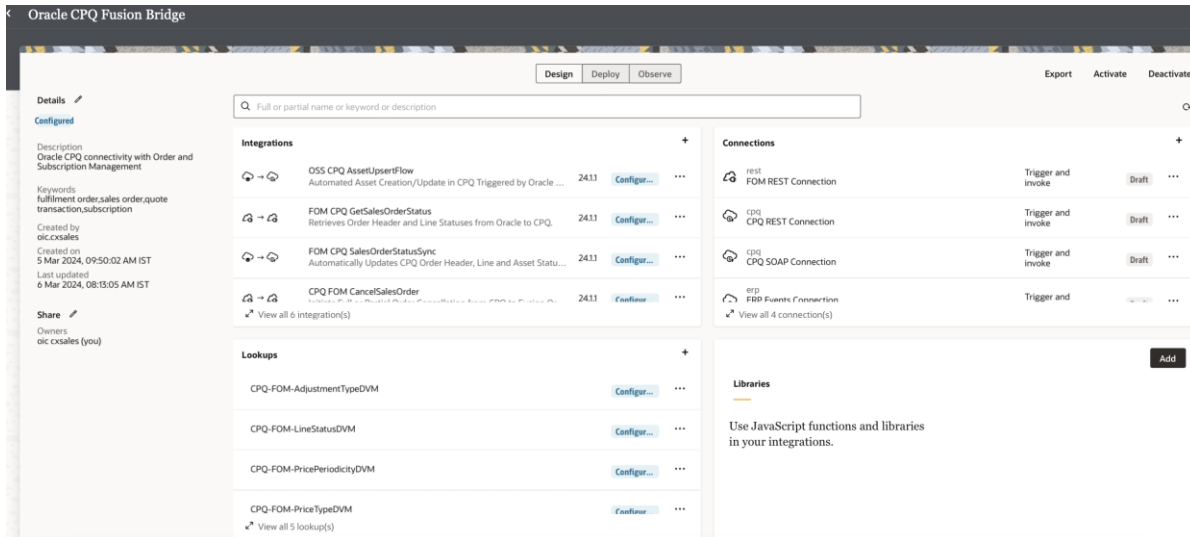
10. Select **Integrations** and verify the following integrations are imported:
 - CPQ FOM CancelSalesOrder
 - CPQ FOM CreateSalesOrder
 - OSS CPQ AssetUpsertFlow
 - FOM CPQ SalesOrderStatusSync
 - FOM CPQ GetSalesOrderStatus
 - OSS CPQ CreateRenewalQuote (Update 24B or later)



11. Select **Projects** in the left side navigation panel and verify the following accelerator project is imported:
 - Oracle CPQ Fusion Bridge



12. Click **Oracle CPQ Fusion Bridge** Project and verify all the above verified Integrations, Connections and Lookups are available.





13. Configure the CPQ REST Connection.


- a. In the Connections Panel, click on 3 dots and **Edit** option for the CPQ REST Connection.
- b. Enter the connection URL in the following format:



`https://<CPQ host name>/rest/<latest rest version>/metadata-catalog`

- c. Enter your CPQ administrator username and password in the corresponding fields.
- d. Click **Test** to verify the connection.
- e. Click **Save**.


- f. **Save changes?** popup with two Inactive integrations (OSS CPQ AssetUpsertFlow and FOM CPQ SalesOrderStatusSync) display. Click **Save**.
- g. Click the Back icon. 
- h. Verify CPQ REST Connection is **Configured**.
- i. Click on 3 dots  again and click on the **Refresh Metadata** option for the CPQ REST Connection.



14. Configure the CPQ SOAP Connection

- a. In the Connections Panel, click on 3 dots  and Edit option for the CPQ SOAP Connection.
- b. Select “SOAP WSDL URL” for the Connection Type.
- c. Enter the connection URL in the following format:


```
https:// <CPQ host name>/v2_0/receiver/commerce/oraclecpq?wsdl
```
- d. Enter your CPQ administrator username and password in the corresponding fields.
- e. Click **Test** to verify the connection.
- f. **Save changes?** popup with one Inactive integration (CPQ FOM CreateSalesOrder) displays. Click **Save**.
- g. Click **Save**.
- h. Click the Back icon. 
- i. Verify CPQ SOAP Connection is **Configured**.
- j. Click on 3 dots  again and click on the **Refresh Metadata** option for the CPQ SOAP Connection.


15. Configure the Order Management REST Connection.

- a. In the Connections Panel, click on 3 dots  and **Edit** option for the **FOM REST Connection**.
- b. Select “restUrl” for the **Connection Type**.
- c. Select “TLSv1.2TLS” for the **Version**.
- d. Enter the connection URL in the following format:




```
https://<Order Management host name>/fscmRestApi/resources/<latest rest version>/
```
- e. Select security policy as Basic Authentication.
- f. Enter your Order Management username and password in the corresponding fields.
- g. Click **Test** to verify the connection.
- h. Click **Save**.
- i. **Save changes?** popup with three Inactive integrations (CPQ FOM CancelSalesOrder, CPQ FOM CreateSalesOrder and FOM CPQ GetSalesOrderStatus) display. Click **Save**.
- j. Click the Back icon. 
- k. Verify FOM REST Connection is **Configured**.
- l. Click on 3 dots  again and click on the **Refresh Metadata** option for the FOM REST Connection.

Note: The integration Order Management user must have appropriate roles to Create Orders using the Order Management REST APIs.



16. Configure the ERP Events Connection.

- a. In the Connections Panel, click on 3 dots  and the **Edit** option for the **FOM REST Connection**.
- b. Enter ERP Cloud Host in the following format:

`https://<Order Management host name>`





- c. Enter your Fusion username and password for Fusion in the corresponding fields.
- d. Click **Test** to verify the connection.
- e. **Save changes?** popup with two Inactive integrations (OSS CPQ AssetUpsertFlow and FOM CPQ SalesOrderStatusSync) display. Click **Save**.
- f. Click **Save**.
- g. Click the Back icon. 
- h. Verify ERP Events Connection is **Configured**.
- i. Click on 3 dots  again and click on the **Refresh Metadata** option for the ERP Events Connection.

17. Activate the CPQ FOM CancelSalesOrder integration.

- a. In the Integrations Panel, click on 3 dots  and **Activate** option for the CPQ FOM CancelSalesOrder integration.
- b. Select **tracing level**.
- c. Click on **Activate**.
- d. Click on the **Refresh**  icon and verify CPQ FOM CancelSalesOrder integration is **Active**.





Note: Oracle recommends that the **Enable Tracing** when activating an integration. This captures valuable troubleshooting information that may be useful in case of a failed integration run or instance.

18. Activate the CPQ FOM CreateSalesOrder integration.

- a. In Integrations Panel, Click on 3 dots  and **Refresh endpoints** option for the CPQ FOM CreateSalesOrder integration.
- b. Select CPQ SOAP Connection and click on Refresh button.
- c. Click **Close**.
- d. Click on **Refresh**  icon and verify CPQ FOM CancelSalesOrder integration is **Locked**.
- e. Wait approximately one minute to verify CPQ FOM CancelSalesOrder integration is **Refresh successful**.
- f. In the Integrations Panel, click on 3 dots  and **Activate** option for the CPQ FOM CreateSalesOrder integration.
- g. Select **tracing level**.
- h. Click on **Activate**.
- i. Click on **Refresh**  icon and verify CPQ FOM CreateSalesOrder integration is **Active**.





Note: Oracle recommends that the **Enable Tracing** when activating an integration. This captures valuable troubleshooting information that may be useful in case of a failed integration run or instance.

19. Activate the OSS CPQ AssetUpsertFlow integration.

- a. In the Integrations Panel, click on 3 dots  and the **Refresh endpoints** option for the OSS CPQ AssetUpsertFlow integration.
- b. Select ERP Events Connection and click the Refresh button.
- c. Click Close.
- d. Click on the **Refresh**  icon and verify OSS CPQ AssetUpsertFlow is **Locked**.
- e. Wait for approximately one minute to verify OSS CPQ AssetUpsertFlow integration is **Refresh successful**.
- f. In the Integrations Panel, click on 3 dots  and the **Activate** option for the OSS CPQ AssetUpsertFlow integration.
- g. Select **tracing level**.
- h. Click on **Activate**.
- i. Click on the **Refresh**  icon and verify OSS CPQ AssetUpsertFlow integration is **Active**.

Note: Oracle recommends that the **Enable Tracing** when activating an integration. This captures valuable troubleshooting information that may be useful in case of a failed integration run or instance.

20. Activate the FOM CPQ SalesOrderStatusSync integration.

- a. In the Integrations Panel, click on 3 dots  and the **Refresh endpoints** option for the FOM CPQ SalesOrderStatusSync integration.
- b. Select ERP Events Connection and click the Refresh button.
- c. Click **Close**.
- d. Click on the **Refresh**  icon and FOM CPQ SalesOrderStatusSync is **Locked**.
- e. Wait approximately one minute to verify the FOM CPQ SalesOrderStatusSync integration is **Refresh successful**.
- f. In the Integrations Panel, click on 3 dots  and the **Activate** option for the FOM CPQ SalesOrderStatusSync integration.
- g. Select **tracing level**.
- h. Click **Activate**.
- i. Click on the **Refresh**  icon and verify FOM CPQ SalesOrderStatusSync integration is **Active**.





Note: Oracle recommends that the **Enable Tracing** when activating an integration. This captures valuable troubleshooting information that may be useful in case of a failed integration run or instance.

21. Activate the FOM CPQ GetSalesOrderStatus integration.

- a. In the Integrations Panel, click on 3 dots  and the **Activate** option for the FOM CPQ GetSalesOrderStatus integration.
- b. Select **tracing level**.
- c. Click **Activate**.
- d. Click on the **Refresh**  icon and verify FOM CPQ GetSalesOrderStatus integration is **Active**.

Note: Oracle recommends that the **Enable Tracing** when activating an integration. This captures valuable troubleshooting information that may be useful in case of a failed integration run or instance.

22. Activate the OSS CPQ CreateRenewalQuote integration.

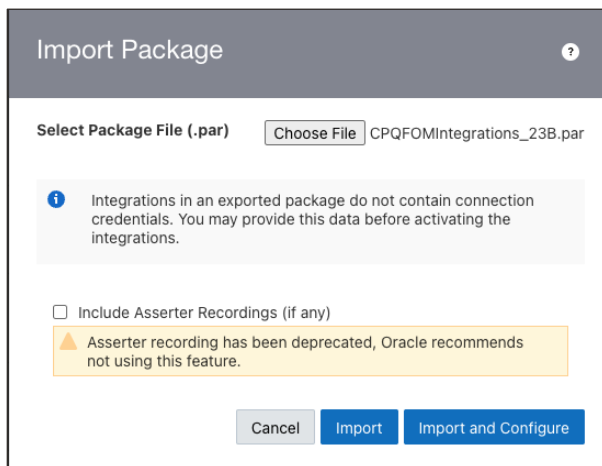
- a. In the Integrations Panel, click on 3 dots  and the **Refresh endpoints** option for the OSS CPQ CreateRenewalQuote integration.
- b. Select ERP Events Connection and click the Refresh button.
- c. Click Close.
- d. Click on the **Refresh**  icon and verify OSS CPQ CreateRenewalQuote is **Locked**.
- e. Wait for approximately one minute to verify OSS CPQ CreateRenewalQuote integration is **Refresh successful**.
- f. In the Integrations Panel, click on 3 dots  and the **Activate** option for the OSS CPQ CreateRenewalQuote integration.
- g. Select tracing level.
- h. Click on **Activate**.
- i. Click on the **Refresh**  icon and verify OSS CPQ CreateRenewalQuote integration is **Active**.

Note: Oracle recommends that the **Enable Tracing** when activating an integration. This captures valuable troubleshooting information that may be useful in case of a failed integration run or instance.

Import Integration Package (Update 23B, 23C, or 23D)

Perform the following steps to import the OIC integration package into OIC.

1. Log in to the OIC site as an administration user.
2. Select **Integrations** in the left side navigation panel, and then select **Packages**.
3. Click **Import**.
4. Click **Choose File**, and then select the CPQFOMIntegrations_<release_version>.par package, where <release_version> refers to the release being implemented.



5. Click **Import and Configure**.
6. Select **Integrations** in the left side navigation panel, verify the following integrations have been imported:
 - CancelSOFromCPQ
 - CreateSOFromCPQ
 - GetSOStatusFromFOM
 - UpdateSOStatusFromFOM
 - UpdateSubStatusFromOSS

Oracle Integration - Integrations

16 of 73 Integrations

Updated By: oic.cpq

Name	Version	Style	Last Updated	Status
GetSOStatusFromFOM Gets Order Header Status and Order Lines Status and update it in CPQ.	2.0.0	App Driven Orchestration	Mar 15th, 2023 03:54:07 PM PDT	Configured
CreateSOFromCPQ	2.0.0	App Driven Orchestration	Mar 15th, 2023 03:54:07 PM PDT	Configured
CancelSOFromCPQ Place cancellation request for whole order or selected line item from CPQ to F...	2.0.0	App Driven Orchestration	Mar 15th, 2023 03:54:07 PM PDT	Configured
UpdateSOStatusFromFOM Updates line status in CPQ when triggered by FOM Business Events.	2.0.0	App Driven Orchestration	Mar 15th, 2023 03:54:06 PM PDT	Configured
UpdateSubStatusFromOSS Updates subscription line status in CPQ when triggered by OSS Business Even...	2.0.0	App Driven Orchestration	Mar 15th, 2023 03:54:06 PM PDT	Configured

7. Select **Connections** in the left side navigation panel, verify the following connections have been imported:

- CPQ REST Connection
- CPQ SOAP Connection
- ERP Events Connection
- FOM REST Connection

Oracle Integration - Connections

16 of 17 Connections

Name	Type	Usage	Last Updated	Status
CPQ SOAP Connection	Oracle CPQ	1 Integrations	Feb 2nd, 2021 07:14:52 AM CST	Draft
FOM REST Connection	REST	3 Integrations	Feb 2nd, 2021 07:14:51 AM CST	Draft
CPQ REST Connection CPQ REST Connection	Oracle CPQ	1 Integrations	Feb 2nd, 2021 07:14:49 AM CST	Draft
ERP Events Connection	Oracle ERP Cloud	1 Integrations	Feb 2nd, 2021 07:14:49 AM CST	Draft

8. Select **Lookups** in the left side navigation panel, verify the following lookups have been imported:

- CPQ-FOM-AdjustmentTypeDVM
- CPQ-FOM-ChargeDefinitionDVM (Oracle CPQ 23C and earlier only)
- CPQ-FOM-LineStatusDVM
- CPQ-FOM-PricePeriodicityDVM
- CPQ-FOM-PriceTypeDVM

Name	Usage	Domains	Values Mapped	Last Updated	Status
CPQ-FOM-PriceTypeDVM This lookup is used to map between CPQ - FOM mappings for Price Type codes	1 Integrations	2	3	Feb 23rd, 2023 05:34:40 PM PST	
CPQ-FOM-PricePeriodicityDVM This lookup is used to map between CPQ - FOM mappings for Price Periodicity cod...	1 Integrations	2	2	Feb 23rd, 2023 05:34:40 PM PST	● Configured
CPQ-FOM-ChargeDefinitionDVM Charge definition names to charge codes	1 Integrations	2	5	Sep 13th, 2022 02:27:52 PM PDT	● Configured
CPQ-FOM-LineStatusDVM	5 Integrations	2	14	Sep 13th, 2022 02:27:52 PM PDT	● Configured

9. Configure the CPQ REST Connection.

- a. Click the Edit icon for the **CPQ REST Connection**.
- b. Enter the connection URL in the following format:

`https://<CPQ host name>/rest/<latest rest version>/metadata-catalog`
- c. Enter your CPQ administrator username and password in the corresponding fields.
- d. Click **Test** to verify the connection.
- e. Click **Save**.
- f. Click the Back icon.

10. Configure the CPQ SOAP Connection

- a. Click on the Edit icon for the **CPQ SOAP Connection**.
- b. Select “SOAP WSDL URL” for the **Connection Type**.
- c. Enter the connection URL in the following format:

`https:// <CPQ host name>/v2_0/receiver/commerce/oraclecpq?wsdl`
- d. Enter your CPQ administrator username and password in the corresponding fields.
- e. Click **Test** to verify the connection.
- f. Click **Save**.
- g. Click the Back icon.


11. Configure the Order Management REST Connection.

- a. Click the Edit icon for the **FOM REST Connection**.
- b. Select “restUrl” for the **Connection Type**.
- c. Select “TLSv1.2TLS” for the **Version**.
- d. Enter the connection URL in the following format.

`https://<Order Management host name>/fscmRestApi/resources/<latest rest version>/`
- e. Select security policy as Basic Authentication.
- f. Enter your Order Management username and password in the corresponding fields.
- g. Click **Test** to verify the connection.
- h. Click **Save**.
- i. Click the Back icon.

Note: The integration Order Management user should have appropriate roles to Create Orders using the Order Management REST APIs.

12. Configure the ERP Events Connection.

a. Click the Edit icon  for the **ERP Events Connection**.

b. Enter ERP Cloud Host in the following format:

`https://<Order Management host name>`

c. Enter your Fusion username and password for Fusion in the corresponding fields.

d. Click **Test** to verify the connection.

e. Click **Save**.

f. Click the Back icon. 

13. On the configuration editor page, click activate for each integration to activate all integrations.

Notes:

- If an integration with the same name already exists in the OIC environment, deactivate the integration before replacing it.
- Oracle recommends that the **Enable Tracing** and **Include Payload** options are selected when activating an integration. This captures valuable troubleshooting information that may be useful in case of a failed integration run or instance.

OIC Mapping Details (Reference Material)

This section contains the OIC mapping details of the Order Management payload and Oracle CPQ attributes.

Note: The OIC Mapping Details section is provided for reference, no action is required to implement the attribute mapping. Attribute mapping is automatically set up when the CPQFOMIntegrations_<release_version>.par package is imported in OIC.

Header Attribute Mapping

Order Management Attribute	Mapped to CPQ Attribute	Comments
BuyingPartyContactFirstName	_customer_t_first_name	
BuyingPartyContactLastName	_customer_t_last_name	
BuyingPartyName	accountPartyId_t	CPQ Attribute Name has been changed from accountPartyId_t to _s_accountPartyId_t in CPQ release 23D
FreezePriceFlag	freezePriceFlag_t	
FreezeShippingChargeFlag	freezePriceFlag_t	
FreezeTaxFlag	freezePriceFlag_t	
PartialShipAllowedFlag	oRCL_ERP_PartialShipAllowed_t	
PaymentTerms	paymentTerms_t	
RequestingBusinessUnitName	businessUnitName_t	
RequestingBusinessUnitId	businessUnitId_t	
SourceTransactionId	bs_id	
SourceTransactionNumber	transactionID_t	
SourceTransactionSystem	transactionSourceSystem_t	
TransactionOn	orderDate_t	
TransactionalCurrencyCode	currency_t	
billToCustomer - AccountNumber	invoiceToPartyID_t OR accountPartyId_t	If invoiceToPartyID_t is not null then invoiceToPartyID_t else accountPartyId_t CPQ Attribute Name has been changed from accountPartyId_t to _s_accountPartyId_t in CPQ release 23D
shipToCustomer - PartyName	_shipTo_t_company_name	

Line Attribute Mapping

Order Management Attribute	Mapped to CPQ Attribute	Comments
RequestedRatePlanId	_price_applied_ratePlan_integrationId	Applied Rate Plan Integration Id added in CPQ Release 23D
OrchestrationProcessName	CustomDOO_PauseforShipSubscriptionsProcesses	(Optional field) Refer to the Order Management Setup section for Process name
EndCreditMethodCode	changeCode_1	When oRCL_ABO_ActionCode_1 = 'TERMINATE'
EndDate	contractEndDate_1	When oRCL_ABO_ActionCode_1 = 'TERMINATE'
EndReasonCode	changeReason_1	When oRCL_ABO_ActionCode_1 = 'TERMINATE'
associatedProductReferences - SourceLineId	associatedLineId_1	Applicable only for Subscription Lines with Associations
associatedProductReferences - SourceOrderId	associatedOrderId_1	Applicable only for Subscription Lines with Associations
associatedProductReferences - SourceOrderSystem	transactionSourceSystem_t	Applicable only for Subscription Lines with Associations
SubscriptionProfileId	subscriptionProfileId_t	Applicable only for Subscription Lines
externalAssetReference - ExternalAssetKey	itemInstanceId_1	Applicable only for Subscription Lines
renewSubscriptionReference - ExternalAssetKey ExternalRootParentAssetKey	itemInstanceId_1 rootAssetKey_1	Applicable only for renew Subscription Lines
ActionTypeCode	'ORA_END' OR 'ORA_RENEW' OR 'ORA_ADD'	When oRCL_ABO_ActionCode_1 = 'TERMINATE' then 'ORA_END' When oRCL_ABO_ActionCode_1 = 'RENEW' then 'ORA_RENEW' Else 'ORA_ADD'
CancelReasonCode		
ContractEndDate	contractEndDate_1	

Order Management Attribute	Mapped to CPQ Attribute	Comments
ContractStartDate	contractStartDate_1	
OrderedQuantity	_price_quantity	
OrderedUOMCode	requestedUnitOfMeasure_1	In 23C, if requestedUnitOfMeasure_1 is null, then default value is set to "Ea"
ParentSourceTransactionLineId	parentDocNumber_1	
PartialShipAllowedFlag	oRCL_ERP_PartialShipAllowed_t	
ProductNumber	_part_number OR _model_name	In 23D, this mapping has been changed as follows: "_line_bom_part_number" will be mapped if it is not empty OR "_model_variable_name"
RequestedFulfillmentOrganizationCode	fulfillmentOrganizationCode_1	
RequestedShipDate	requestShipDate_1	
SourceScheduleNumber	_document_number	
SourceTransactionLineId	_document_number	
SourceTransactionLineNumber	_document_number	
SourceTransactionScheduleId	_document_number	
TransactionCategoryCode	'RETURN'	When oRCL_ABO_ActionCode_1 = 'TERMINATE'
coveredProductReferences - SourceLineNumber	_s_coverageLineNumber	Added in 23D but is for future use Applicable only for Coverage Lines
coveredProductReferences - SourceOrderNumber	_s_coverageOrderNumber	Added in 23D but is for future use Applicable only for Coverage Lines

Charge Attribute Mapping

Order Management Attribute	Mapped to CPQ Attribute	Comments
ApplyTo	oRCL_applyTo	
ChargeCurrencyCode	currency_t	
ChargeDefinitionCode	oRCL_chargeName	Uses OIC lookup CPQ-FOM-ChargeDefinitionDVM for domain value map. The value should match with the Order Management Pricing Charge Definition code. In 23D, mapping changed to “_s_oRCL_chargeDefIntegrationId”
ChargeTypeCode	oRCL_chargeType	CPQ Attribute Name has been changed from oRCL_chargeType to _s_oRCL_chargeType in CPQ release 23C
PricePeriodicityCode	pricePeriod_1	In CPQ release 23C, the CPQ attribute mapping has been changed to _s_oRCL_pricePeriod Applicable only when priceType_1 = ‘Recurring’ or ‘Usage’. Uses OIC lookup CPQ-FOM- PricePeriodicityDVM for domain value map.
PriceTypeCode	priceType_1	In CPQ release 23C, the CPQ attribute mapping has been changed to _s_oRCL_priceType Uses OIC lookup CPQ-FOM- PriceTypeDVM for domain value map. Default value is “One Time”
PricedQuantity	_price_quantity	
PrimaryFlag	oRCL_primaryCharge	
RollupFlag	rollupFlag_1	
SequenceNumber	oRCL_chargeSequenceNumber	
SourceChargeId	oRCL_chargeName	

Charge Attribute Mapping for List and Net Prices

In Fusion Order Management, Create Order payload, List Price and Net Price are wrapped in chargeComponents.

Order Management Attribute	Mapped to CPQ Attribute 23B	Comments
ChargeCurrencyCode	currency_t	
ChargeCurrencyUnitPrice	oRCL_listPrice or oRCL_netPrice	
HeaderCurrencyCode	currency_t	
HeaderCurrencyDurationExtendedAmount	oRCL_contractListValue or oRCL_contractNetValue	
HeaderCurrencyExtendedAmount	oRCL_listAmount or oRCL_netAmount	
HeaderCurrencyUnitPrice	oRCL_listPrice or oRCL_netPrice	
PriceElementCode	QP_LIST_PRICE or QP_NET_PRICE	
PriceElementUsageCode	LIST_PRICE or NET_PRICE	
RollupFlag	False	
SequenceNumber	'1' for List Price, '2' for Net Price	
SourceChargeComponentId	oRCL_chargeName-C1 for List Price, oRCL_chargeName-C2 for Net Price	Unique charge component Id generated using charge name.

Charge Attribute Mapping for Adjustments

In Fusion Order Management, Create Order payload, Discounts (Adjustments) are wrapped in chargeComponents.

Order Management Attribute	Mapped to CPQ Attribute	Comments
ChargeCurrencyCode	currency_t	
ChargeCurrencyUnitPrice	oRCL_customDiscountAmount	
HeaderCurrencyCode	currency_t	
HeaderCurrencyDurationExtendedAmount	oRCL_contractDiscountAmount	
HeaderCurrencyExtendedAmount	oRCL_discountAmount	
HeaderCurrencyUnitPrice	oRCL_customDiscountAmount	
PriceElementCode	'QP_CUSTOM_ADJUSTMENT'	
PriceElementUsageCode	PRICE_ADJUSTMENT	
RollupFlag	False	
SequenceNumber	3	
SourceChargeComponentId	oRCL_chargeName-C3	Unique charge component Id generated using charge name.
Explanation	Auto adjustment	

Charge Attribute Mappings for Manual Price Adjustments

In Fusion Order Management, Create Order payload, Discounts (Adjustments) are also mapped as Manual Price Adjustments.

This mapping is added only if `_s_oRCL_customDiscountType` is not null

Order Management Attribute	Mapped to CPQ Attribute 23B	Comments
AdjustmentAmount	oRCL_customDiscountAmount	
AdjustmentElementBasisName	'Your Price'	
AdjustmentTypeCode	_s_oRCL_customDiscountType	Uses OIC lookup CPQ-FOM-AdjustmentTypeDVM for domain value map.
ChargeDefinitionCode	oRCL_chargeName	Uses OIC lookup CPQ-FOM-ChargeDefinitionDVM for domain value map. The value should match with the Order Management Pricing Charge Definition code. In 23D, mapping changed to "_s_oRCL_chargeDefIntegrationId"
ChargeRollupFlag	'false'	
Comments	'Discount requested'	
ReasonCode	'ORA_PRICE_MATCH'	
SequenceNumber	oRCL_chargeSequenceNumber	
SourceManualPriceAdjustmentId	oRCL_chargeName + oRCL_chargeSequenceNumber	Concatenate oRCL_chargeName and oRCL_chargeSequenceNumber

Oracle CPQ Package Installation and Setup

Oracle creates implementation packages to distribute elements needed by customers to implement new Oracle CPQ features. This section contains information about importing the CPQ – Order Management Package into your CPQ environment.

The CPQ – Order Management Package (CPQ_FOM_Package_<release_version>.zip) is a granular migration CPQ package containing CPQ artifacts to support the Order Management Integration solution.

Note: This package also contains some of the artifacts related to subscription products handling in CPQ and Order Management Integration for a future release.

CPQ Integration Center Setup

Before importing the CPQ – Order Management Package, administrators must create an OIC and generic integrations.

Create an OIC Integration

Creating an OIC integration enables Oracle CPQ to connect to back-office systems, on premise environments, and other Oracle products in a consistent, enhanced manner.

Notes:

- Oracle Integration Cloud (OIC) is also known as Integration Cloud Service (ICS). When creating the OIC integration in the Integration Center, select Integration Cloud Service as the integration type.
- Use the same username and password that was used to import the integration package in OIC.

Perform the following steps to create an OIC integration.

1. Navigate to the Admin Home page.
2. Select **Integration Center** in the Integration Platform section.
3. Click **Create Integration**.
4. Select **Integration Cloud Service** from the **Type** drop-down.
5. Select **Gen 3** from the **Generation** drop-down.

6. Enter **OSSICS** in the Name field.
7. Enter **oSSICS** in the Variable Name field.
8. If you are using Gen 2 Generation, enter the **Discovery URL**:
For example, <https://<OIC hostname>:7002/icsapis/v1/integrations>.
9. Enter the username and password for the OIC environment.
10. If you are using Gen 2 Generation, click **Test** to verify the connection. The status must return “Test Connection Passed” before proceeding.
11. Select the **Enable Integration** check box.

12. Click **Save**.

Create a Get Sales Order Status from FOM Integration

Update 24A or later:

This integration uses the OIC **FOM CPQ GetSalesOrderStatus** integration endpoint URL. Refer to [Appendix F: Retrieve OIC Integration Endpoint URL](#) for instructions to retrieve the endpoint URL of an OIC integration.

Perform the following steps to create a generic integration to get sales order status from Order Management.

1. Navigate to the Admin Home page.
2. Select **Integration Center** in the Integration Platform section.
3. Click **Create Integration**.
4. Select **Generic Integration** from the **Type** drop-down.

The screenshot shows the 'Integration Center' interface. On the left is a sidebar with a list of integration types: eSignature, Integration Cloud Service, OSSICS, Remote Approval, DataCube Integration, Contract Management, and Generic Integration. The main area is titled 'Integration Center' and contains a form for creating a new integration. The 'Type' dropdown is set to 'Generic Integration'. The form fields are: '*Name' (GetSOSStatusfromFOM), 'Request URL' (https://sitename.oracle.com:7003/ic/integration/v1/flows/h), 'Username' (icsadmin), and 'Password' (masked with dots). Below the fields is an 'Enable Integration' checkbox which is checked, followed by a warning icon and the text 'Disabling this integration will update integrations currently using this service.' In the top right corner of the form area, there are three buttons: 'Back', 'Create Integration', and 'Save'.

5. Enter **GetSOSStatusfromFOM** in the Name field.
6. Enter the **Request URL** in the following format:
`http://<OIC hostname>/ic/api/integration/v2/flows/rest/project/ORCL-BA-CPQ_FUSION_BRIDGE/ORCL-BA-GET_SALES_ORDER_STATUS/24.0/GetSOSStatusfromFOM/`
The host name is the OIC environment name.
7. Enter the username and password for the OIC environment.
8. Select the **Enable Integration** check box.
9. Click **Save**.

Update 23B, 23C, or 23D:

This integration uses the OIC **GetSOSStatusfromFOM** integration endpoint URL. Refer to [Appendix F: Retrieve OIC Integration Endpoint URL](#) for instructions to retrieve the endpoint URL of an OIC integration.

Perform the following steps to create a generic integration to get sales order status from Order Management.

1. Navigate to the Admin Home page.
2. Select **Integration Center** in the Integration Platform section.
3. Click **Create Integration**.
4. Select **Generic Integration** from the **Type** drop-down.

5. Enter **GetSOSStatusfromFOM** in the Name field.
6. Enter the **Request URL** in the following format:
`http://<OIC hostname>/ic/api/integration/v1/flows/rest/getsostatusfromfom/2.0/GetSOSStatusfromFOM/`
 The host name is the OIC environment name.
7. Enter the username and password for the OIC environment.
8. Select the **Enable Integration** check box.
9. Click **Save**.

Create a Cancel a Sales Order from CPQ Integration

Update 24A or later:

This integration uses the OIC **CPQ FOM CancelSalesOrder** integration endpoint URL. Refer to [Appendix F: Retrieve OIC Integration Endpoint URL](#) for instructions to retrieve the endpoint URL of an OIC integration.

Perform the following steps to create a generic integration to cancel a sales order from CPQ.

1. Navigate to the Admin Home page.
2. Select **Integration Center** in the Integration Platform section.
3. Click **Create Integration**.
4. Select **Generic Integration** from the **Type** drop-down.

5. Enter **CancelSOFromCPQ** in the Name field.
6. Enter the **Request URL** in the following format:
`https://<OIC hostname>/ic/api/integration/v2/flows/rest/project/ORCL-BA-CPQ_FUSION_BRIDGE/ORCL-BA-CANCEL_SALES_ORDER/24.0/CancelSOfromCPQ`
 The hostname is the OIC environment name.

7. Enter the username and password for the OIC environment.
8. Select the **Enable Integration** check box.
9. Click **Save**.

Update 23B, 23C, or 23D:

This integration uses the OIC **CancelSOfromCPQ** integration endpoint URL. Refer to [Appendix F: Retrieve OIC Integration Endpoint URL](#) for instructions to retrieve the endpoint URL of an OIC integration.

Perform the following steps to create a generic integration to cancel a sales order from CPQ.

1. Navigate to the Admin Home page.
2. Select **Integration Center** in the Integration Platform section.
3. Click **Create Integration**.
4. Select **Generic Integration** from the **Type** drop-down.

The screenshot shows the 'Integration Center' interface. On the left is a sidebar with categories: eSignature, Integration Cloud Service, OSSICS, Remote Approval, DataCube Integration, Contract Management, Generic Integration, and GetSOStatusfromFOM. The main area is titled 'Integration Center' and contains a 'Type' dropdown menu set to 'Generic Integration'. Below this are input fields for:

- *Name: CancelSOFromCPQ
- Request URL: https://sitename.oracle.com:7003/ic/integration/v1/flows/1
- Username: icsadmin
- Password: [masked]

 At the bottom, there is an 'Enable Integration' checkbox which is checked, followed by a warning icon and the text: 'Disabling this integration will update integrations currently using this service.' In the top right corner of the form, there are three buttons: 'Back', 'Create Integration', and 'Save'.

5. Enter **CancelSOFromCPQ** in the Name field.
6. Enter the **Request URL** in the following format:
`https://<OIC hostname>/ic/api/integration/v1/flows/rest/CANCELISOFROMCPQ/2.0/CancelSOfromCPQ`
 The hostname is the OIC environment name.
7. Enter the username and password for the OIC environment.
8. Select the **Enable Integration** check box.
9. Click **Save**.

Import CPQ-Oracle Order Management Package

The CPQ-Oracle Order Management package is a granular migration package containing new elements in support of the Order Management solution. As a granular migration package, administrators can add or remove specific elements from the package or remove specific elements when importing the package.

Note: Please ensure that an OIC integration is setup with variable name “oSSICS”, according to the steps mentioned in the section [Create an OIC Integration](#), before importing the CPQ_FOM_Package_<release_version>.zip

Perform the following steps to install the Order Management package.

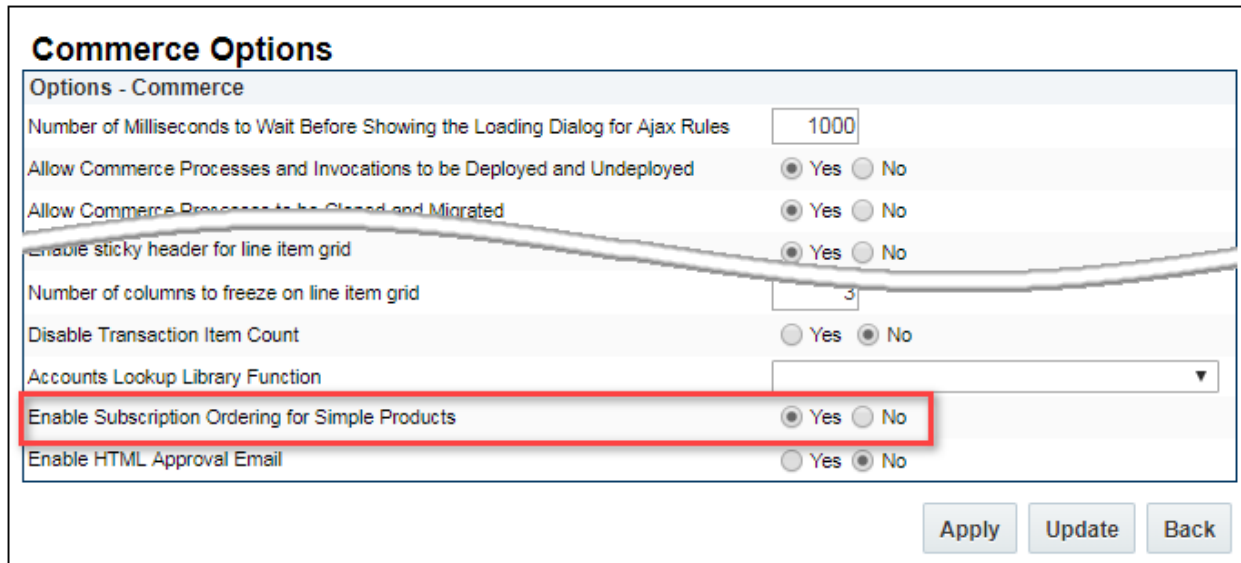
1. Download the Oracle Order Management package (i.e., CPQ_FOM_Package_<release_version>.zip) from [My Oracle Support](#).
2. Navigate to the Admin Home page.
3. Select **Migration** in the Utilities section.
4. Select **Import Package** from the **Select A Mode** drop-down. The Upload Package dialog opens.
5. Click **Browse** and navigate to the CPQ_FOM_Package_<release_version>.zip package.
6. Select the target process from the Choose a target process for Cross Process Migration drop-down.
7. Click **Upload**.
8. Click **Migrate**.
9. When the migration completes, check the migration logs for errors.

Enable Subscription Ordering for Simple Products (Optional)

Customers can enable Subscription Ordering support to directly add simple products to a Commerce Transaction for an asset-based order. A simple product is a product that does not have its part number associated with any of the related configuration models. When enabled, users can use Quick Add to add simple products to a Transaction without navigating away from the Transaction page. They can also add simple products using a parts search.

Perform the following steps to enable Subscription Ordering for Simple Products.

1. Navigate to the Admin Home page.
2. Select **Commerce Settings** in the Commerce and Documents section.



The screenshot shows the 'Commerce Options' configuration page. The 'Enable Subscription Ordering for Simple Products' option is highlighted with a red box and is set to 'Yes'. Other options include 'Number of Milliseconds to Wait Before Showing the Loading Dialog for Ajax Rules' (1000), 'Allow Commerce Processes and Invocations to be Deployed and Undeployed' (Yes), 'Allow Commerce Processes to be Cloned and Migrated' (Yes), 'Enable sticky header for line item grid' (Yes), 'Number of columns to freeze on line item grid' (3), 'Disable Transaction Item Count' (No), 'Accounts Lookup Library Function' (dropdown), and 'Enable HTML Approval Email' (No). Buttons for 'Apply', 'Update', and 'Back' are visible at the bottom right.

3. Set the Enable Subscription Ordering for Simple Products option to **Yes**.
4. Click **Apply** or **Update**.

CPQ Standard Process Elements

The new Commerce Standard Process feature simplifies the implementation of Order Management solution by including the following elements in the *Oracle Quote to Order* Commerce process in the Oracle CPQ RefApp 23B or later.

- Commerce Library Functions
- Commerce Attributes
- Commerce Actions
- Commerce Rules

Notes:

- The new Standard Process now includes several elements that are used for Order Management integration. Refer to the Oracle CPQ Administration Online Help – [Commerce Standard Process](#) topic for more information.
- The remaining elements required for integration can be imported from the release-specific Oracle CPQ-Order Management package.

Commerce Integrations

Note: Some of the Commerce Integrations in this section are provided for reference no action is required to implement Commerce integrations. The following Commerce integrations are included in the Oracle Quote to Order Commerce process (the Commerce Standard Process in the Oracle CPQ RefApp 23B or later) or are imported in the release-specific Oracle CPQ Order Management package, with the exception of [Process Create Order Response](#) which may have to be implemented manually.

Create Order (Integration Cloud Service)

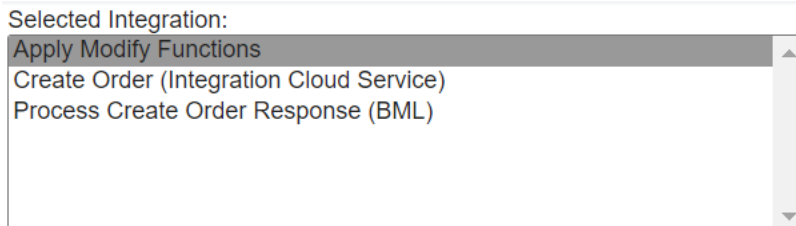
The following Integration Cloud Service type integration invokes OIC to initiate the Create Order workflow.

- **Name:** Create Order
- **Type:** Integration Cloud Service
- **Variable Name:** `createOrder`
- **Timeout:** 500000
- **Endpoint URL:** enter endpoint url from the integration's OIC metadata. Refer to [Appendix L: Retrieve OIC Integration Endpoint URL](#).
- **Associated Trigger-Action Name:** `order_start`

Process Create Order Response (BML)

Note: Customers implementing the new Commerce Standard Process (available beginning in Oracle CPQ 23C platform) or customers implementing new Commerce Standard Process in Oracle CPQ 23B release, who upgrade to Oracle CPQ 23C or later platform, **must manually** create the following [Process Create Order Response](#) BML type integration if it does not exist post-upgrade.

After creating and verifying the Commerce integration, navigate to the transaction action **Create Order** with variable name `order_start` and under the **Integration** tab ensure that the **Selected Integration** list is populated in the following order:



The following commerce BML integration processes REST response from the Create Order ICS Integration and apply it to the transaction.

- **Name:** Process Create Order Response
- **Type:** BML
- **Variable Name:** `processCreateOrderResponse`
- **Run Type:** Always Run
- **BML Function:** Refer to [Appendix B: BML – Process Create Order Response](#)
- **Associated Trigger-Action Name:** `order_start`

Get Sales Order Status from Oracle Order Management (BML)

The following commerce BML integration is used to get sales order status from Order Management.

- **Name:** Get SO Status From FOM
- **Type:** BML
- **Variable Name:** `getSOStatusFromFOM`
- **Run Type:** Always Run
- **Commerce Library Function:** `getSOStatusFromFOM`
- **Associated Trigger-Action Name:** `cancelOrder, getSOStatusFromFOM`

Cancel Sales Order from CPQ (BML)

The following commerce BML integration is used to cancel a sales order status from CPQ.

- **Name:** Cancel SO From CPQ
- **Type:** BML
- **Variable Name:** cancelSOFromCPQ
- **Run Type:** Always Run
- **Commerce Library Function:** cancelOrder
- **Associated Trigger-Action Name:** cancelOrder

Update Fulfillment Line Status (BML)

The following commerce BML integration is used to update fulfillment line status.

- **Name:** Update Fulfillment Line Status
- **Type:** BML
- **Variable Name:** updateFulfillmentLineStatus
- **Run type:** Always Run
- **Commerce Library Function:** updateFulfillmentLineStatus
- **Associated Trigger-Action Name:** updateAsset

Update Asset (BML)

The following commerce BML integration is used to update an asset.

- **Name:** Update Asset
- **Type:** BML
- **Variable Name:** updateAsset
- **Run type:** Always Run
- **Commerce Library Function:** updateAsset
- **Associated Trigger-Action Name:** updateAsset

Commerce Attributes

All the relevant commerce attributes are available in the *Oracle Quote to Order* Commerce process, based on the new Commerce Standard Process in the Oracle CPQ RefApp 23B or later. Refer to Oracle CPQ Administration Online Help - Standard Process for more information.

Commerce Actions

The following are some of the commerce actions that are already included in the Oracle Quote to Order Commerce process, based on the new Commerce Standard Process in the Oracle CPQ RefApp 23B or later.

- **Cancel Order** - This main document action is associated with the Cancel Order button on the Transaction UI and is used to cancel and entire order or the selected line items.
 - **Name:** Cancel Order
 - **Variable Name:** `cancelOrder`
 - **Selected Integrations:** Cancel SO From CPQ (BML), Get SO Status From FOM (BML)

Refer to the following sections for setup and implementation:

- [Generic Integration for Cancel a Sales Order from CPQ](#)
- [Commerce Integration for Cancel Sales Orders from CPQ](#)

- **Create Order** - This main document action is associated with the Create Order button on the Transaction UI. The action is used to create an order in Order Management. This action is hidden if the transaction doesn't contain any line items.
 - **Name:** Create Order
 - **Variable Name:** `order_start`
 - **Selected Integration:** Create Order (Integration Cloud Service), Process Create Order Response (BML)

The integrations need to be in following order for Create Order action:

1. Apply Modify Functions
2. Create Order (ICS) Integration
3. Process Create Order Response (BML) Integration

- **Get Account Details:** - This main document action is used to support account integration. When users click the Customer (or Account) Details tab on the Transaction page, a Customer Company Name field is available. By entering a customer company name and clicking Customer Details, the account fields (Party ID, Invoice to Party id, Bill To Site ID) are populated and mapped to the associated Order Management fields. These field mappings support the creation of a new order in Order Management based on the information provided in the Oracle CPQ Transaction.
 - **Name:** Get Account Details
 - **Variable Name:** `getAccountDetails`

Refer to [BML – Process Create Order Response](#)

This BML is used to process the rest response received from Create Order ICS integration. It is associated with Create Order(`order_start`) action.

BML Input Attributes:

- Transaction: `createOrderRestResponse`
- Transaction Line: `_document_number, _line_bom_parent_id`

```
retStr = "";
if(createOrderRestResponse <> ""){
    for line in transactionLine {
        if (line._line_bom_parent_id == "") {
            retStr = retStr + line._document_number +
            "~fulfillmentStatus_1~BEING_FULFILLED|";
        }
        retStr = retStr + createOrderRestResponse;
    }
}
return retStr;
```

BML - Get Account Details for more information.

- **Get Sales Order Status from Oracle Order Management** - This main document action is associated with the Get SO Status from FOM button and is used to get the order header and line status from FOM on demand.
 - **Name:** Get SO Status From FOM
 - **Variable Name:** `getSOStatusFromFOM`
 - **Selected Integration:** Get SO Status From FOM (BML)

Refer to the following setup and implementation:

- [Generic Integration for Get Sales Order Status from FOM](#)
- [Commerce Integrations for Get Sales Order Status from Oracle Order Management](#)

- **Save** - This main document action is used to save the current state of the transaction and calculate roll up charges for all lines.
 - **Name:** Save
 - **Variable Name:** `cleanSave_t`
- **Update Asset** - This main document action is used to create assets in the CPQ Asset Repository based on the order line status. The code associated with this action is executed via the REST API generated for this action. This REST API is invoked through the orchestration logic in UpdateSOStatusFromOSS OIC integration.
 - **Name:** Update Asset
 - **Variable Name:** `updateAsset`
 - **Selected Integrations:** Update Fulfillment Line Status (BML), Update Asset (BML)

Refer to the following sections for setup and implementation:

- [Commerce Integration for Update Fulfillment Line Status](#)
- [Commerce Integration for Update Asset](#)
- For Update Asset REST API integration usage, refer to [Appendix E1: OIC Integration – UpdateSOStatusFromFOM](#).

- **Save** - This sub-document action used to save the current state of the line item and calculate roll up charges a single line item,
 - **Name:** Save
 - **Variable Name:** `saveChanges_1`

Library Functions

The Oracle Quote to Order Commerce process, based on the new Commerce Standard Process in the Oracle CPQ RefApp 23B or later release includes several commerce library functions.

Validation Rules

Validation Rules are used to validate attribute or field values. They are linked to an action and only run when a specific action is clicked by the user. The Oracle Quote to Order Commerce process, based on the new Commerce Standard Process in the Oracle CPQ RefApp or later includes the following Validation Rules.

- **Validate Line Status Values for Asset Creation** This main document validation rule is used to validate that the 'Trigger Update Asset for Line Status' attribute is not empty when the Update Asset action is invoked.
 - **Name:** Validate Line Status Values For Asset Creation
 - **Variable Name:** `validateLineStatusValuesForAssetCreation`
 - **Status:** Active
 - **Linked Actions:** Save, Update Asset
 - **Condition Type:** Always True
 - **Action Type:** Simple
 - **Components to Validate:** 'Trigger Update Asset for Line Status' main document attribute
 - **Operator:** Equals
 - **Value:** (Empty)

- **Validate Line Status Values for Cancel Order** This main document validation rule is used to validate that 'Cancel Order' is not allowed when status = Awaiting Billing
 - **Name:** Validate Line Status Values for Cancel Order
 - **Variable Name:** `_s_validateLineStatusValuesForCancelOrder`
 - **Status:** Active
 - **Linked Actions:** Cancel Order
 - **Condition Type:** Always True
 - **Action Type:** Simple
 - **Components to Validate:** Status' main document attribute
 - **Operator:** Constrain
 - **Value:** Awaiting Billing

Hiding Rules

Hiding Rules tell Oracle CPQ to hide select attributes or actions when a pre-defined condition is satisfied. They are made up of a condition and an action. The values of the attributes selected as the condition attributes determine the result of the condition, which when True trigger the hiding of the action attributes.

- **Hide Create Order** – This main document hiding rule hides the **Create Order** action when the order is already created or if no lines are present in the transaction.
 - **Name:** Hide Create SO Action
 - **Variable Name:** `hideCreateSOAction`
 - **Status:** Active
 - **Condition Type:** Advanced
 - **Action Type:** Simple
 - **Components to Hide:** 'Create Order' action
- **Hide Cancel Order Button** – This main document hiding rule hides the **Cancel Order** action when the order is not created (i.e., if order Key is empty, hide the Cancel order action).
 - **Name:** Hide Cancel Order Button
 - **Variable Name:** `hideCancelOrderButton`
 - **Status:** Active
 - **Condition Type:** Simple
 - **Condition 1 Attribute:** Order Key
 - **Condition 1 Operator:** Equals
 - **Condition 1 Value:** (Empty)
 - **Action Type:** Simple
 - **Components to Hide:** 'Cancel Order' action and 'Cancel Reason' main document attribute
- **Hide SO Status Action** – This main document hiding rule hides the **Get SO Status From FOM** action when the order is not created (i.e., if order Key is empty, hide the Get SO Status From FOM action).
 - **Name:** Hide SO Status Action
 - **Variable Name:** `hideSOStatusAction`
 - **Status:** Active
 - **Condition Type:** Simple
 - **Attribute:** Order Key
 - **Operator:** Equals
 - **Value:** (Empty)
 - **Action Type:** Simple
 - **Components to Hide:** 'Get SO Status from FOM' action
- **Recurring Attributes (Transaction)**– This main document hiding rule hides all totals attributes related to Recurring Pricing if no such line items exist on the quote.
 - **Name:** Recurring Attributes (Transaction)
 - **Variable Name:** `recurringAttributes_t`
 - **Status:** Active
 - **Condition Type:** Simple
 - **Action Type:** Simple
 - **Components to Hide:**
 - Contracted Periods
 - Annual Value (Net)
 - Annual Discount
 - Annual Value (List)
 - Period

Order Management Setup

Business events must be enabled in Order Management for different Fulfillment Statuses to synchronize the order header and line status and to create assets in the CPQ Asset repository. This section provides steps that are required in Order Management for integration with OIC using business events. For more details about the setup, refer to the [Oracle Order Management Guide](#).

Enable Business Events

Manage Trigger Points for Business Events

OIC integration uses the Sales Order Notification business event raised by the Order Management system based on conditions defined as business event triggers.

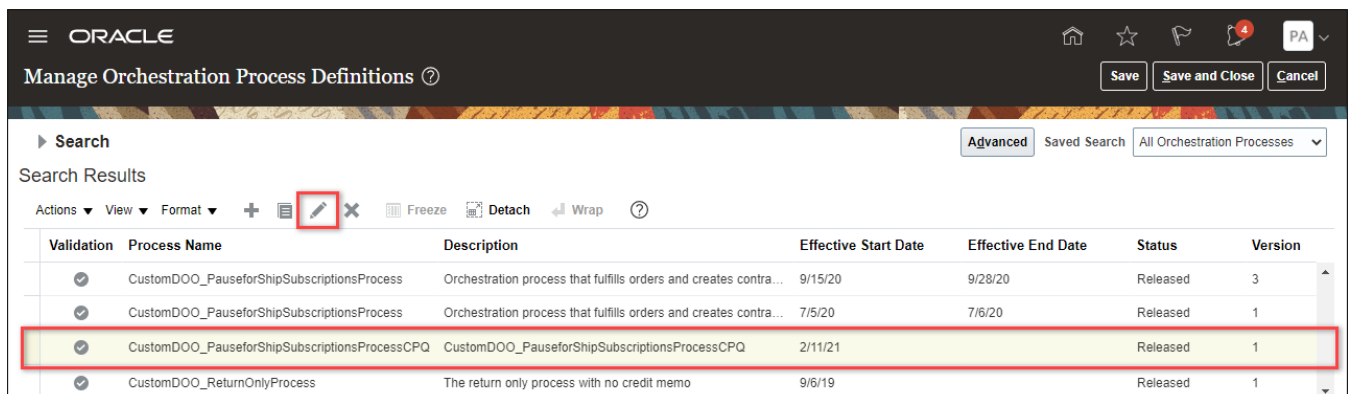
1. Login to your Order Management environment as admin user.
2. In the Navigator, click **Setup and Maintenance**.
3. Select **Order Management** from the Setup drop-down.
4. Search for the **Manage Business Event Trigger Points** task.
5. Click on the **Manage Business Event Trigger Points** task.
6. Select the **Active** option for the following triggers.
 - o Fulfillment Line Status Update
 - o Fulfillment Line Closed
 - o Order Header Status Update

Note: This will raise a business event and send a notification to each subscriber for the selected trigger points.

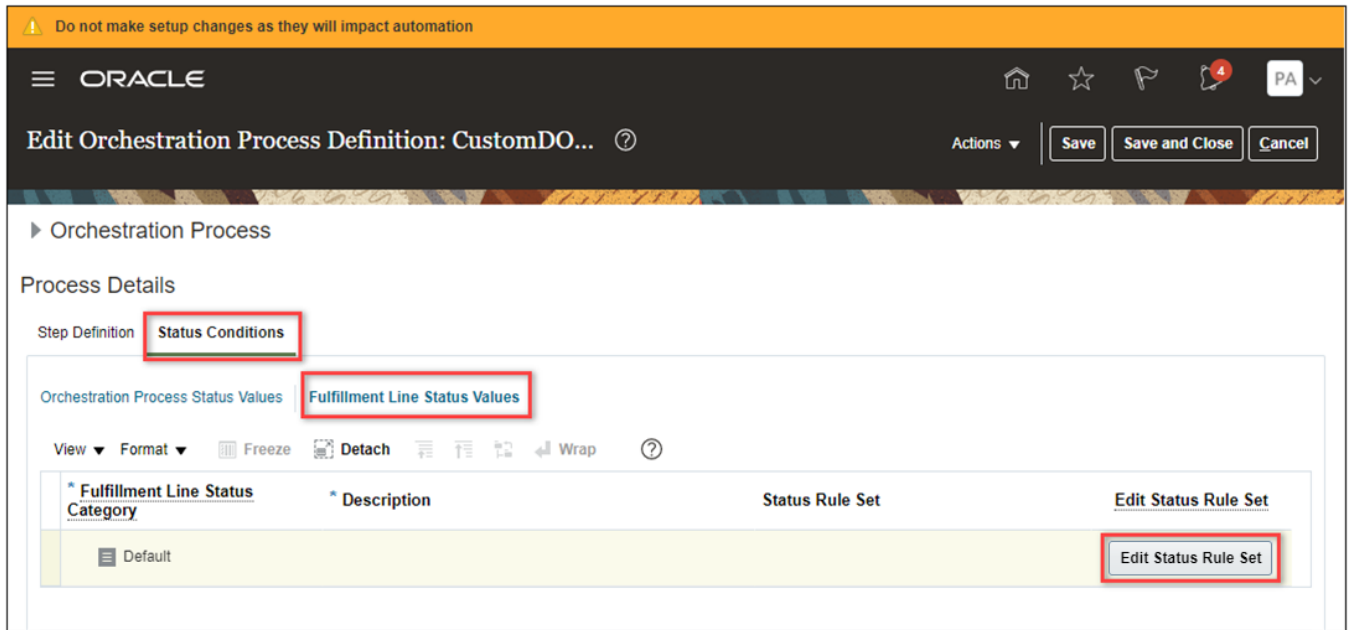
Send Status Updates for Fulfillment Lines

Perform the following steps to send an update to each subscriber when the status changes on a fulfillment line.

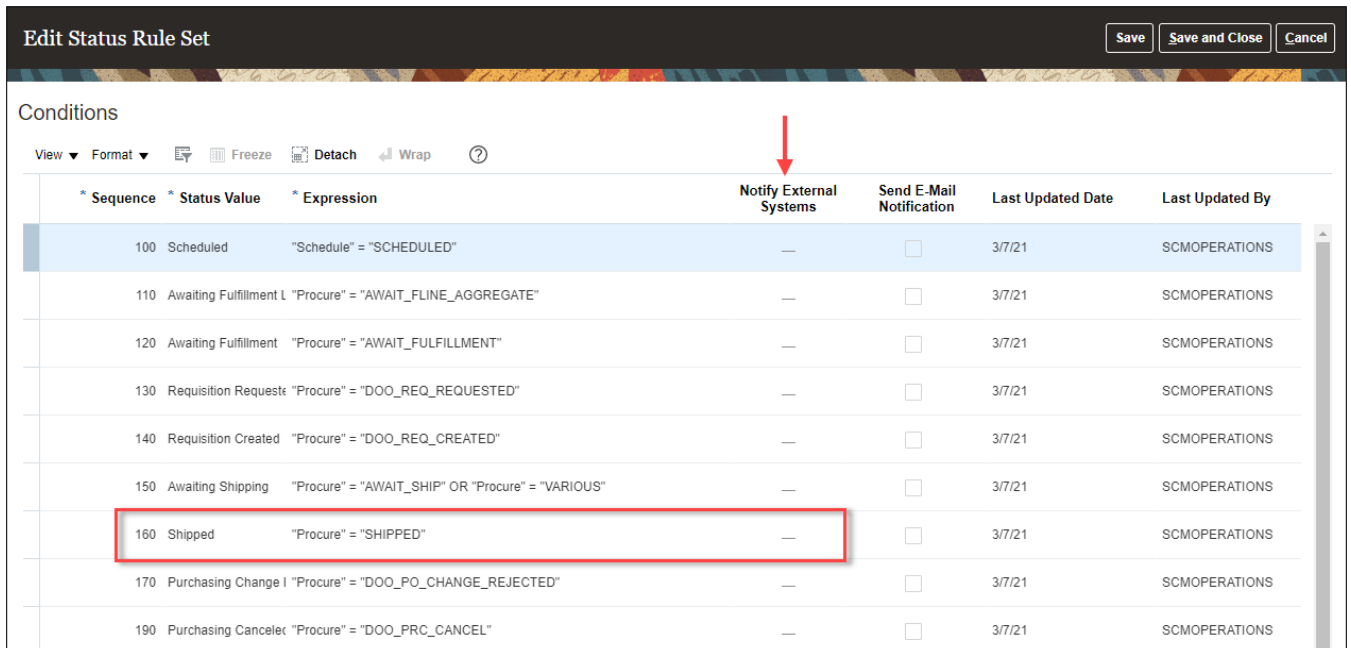
1. Login to your Order Management environment as admin user.
2. In the Navigator, click **Setup and Maintenance**.
3. Select **Order Management** from the Setup drop-down.
4. Search for the **Orchestration Process Definitions** task.
5. Click on the **Orchestration Process Definitions** task.
6. On the Manage Orchestration Process Definitions page, search for the orchestration process that your deployment uses. Each orchestration process controls the status value for each fulfillment line, so you must modify the orchestration process that controls the status value.
7. From search results, click on your process name and click on the pencil icon to edit the process. In the following screenshot we have selected one example process to demonstrate further steps.



- In the Process Details area, click Status Conditions > Fulfillment Line Status Values > Edit Status Rule Set



- Edit Status Rule Set page opens with all status conditions listed. For any status that you want to send out notifications, select the checkbox for 'Notify External Systems' column.



- Click on **Save** or **Save and Close**.
- Deploy
- Repeat this procedure for each orchestration process in your deployment that updates status values.

Send Status Updates for Subscription Lines

Event subscription from Subscription Management

By activating the UpdateSubStatusFromOSS, the OIC integration receives subscription created Business Events. No additional setup required.

Edit Request

Oracle ERP Cloud trigger

Define the purpose of this trigger

- Receive Business Events raised within ERP Cloud
- Receive Callback Message upon completion of FBDI bulk Import Job submitted via another Integration
- Receive Requests from ERP Cloud Applications (explicitly sent from Groovy Script or Business Logic)

Business Event For Subscription

- Strategy External Task opened
- Data Pool Subscription Event
- Subscription AR Intergration Event
- Subscription Created Event**
- Subscription Deleted Event
- Subscription Updated Event

Filter Expr for Subscription Created Event

```
<xpathExpr
xmlns:ns1='http://xmlns.oracle.com/apps/subscriptions/subscriptionManagement/s
ubscriptionIntegrationService/types/'
xmlns:ns2='http://xmlns.oracle.com/apps/subscriptions/subscriptionManagement/s
ubscriptionService/'>$eventPayload/ns1:result/ns2:SourceSystem =
'ORA_BM_CPQ'</xpathExpr>
```

Event Description

This public event signals a subscription is created

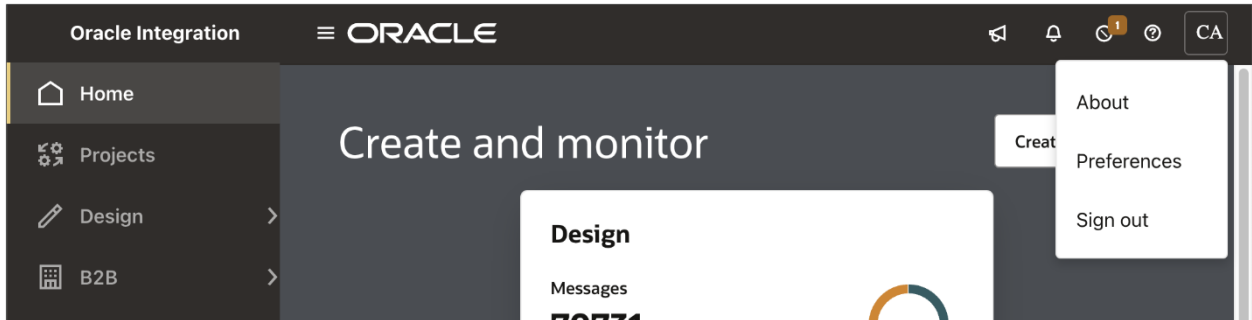
Setting Up Order Management Event Subscriptions

This step is required to subscribe to the Order Management business events using OIC.

Get the CSF Key

Perform the following steps to acquire the CSF (Credential Store Framework) key.

1. Log in to the OIC site as an administration user.
2. Select **About** from the account drop-down.



3. Record the CSF Key which is a combination of **Identity Domain** and **Service Instance**:
<Identity Domain><Service Instance>

About



ORACLE®
Integration

Version: **23.12**

Service instance: **sample-instance-name**

Instance id: **sample.integrationinstance.id**

Identity domain: **sample-tenant-name**

Service type: **ENTERPRISEX**

Outbound IP: **0.0.0.0**

Inbound IP: **0.0.0.0**

Bring your own license (BYOL): **false**

Number of message packs: **1**

> **Enabled features**

[About Oracle](#) | [Contact Us](#) | [Legal Notice](#) | [Term of Use](#) | [Privacy Rights](#)
Oracle © 2015, 2023, Oracle and/or its affiliates. All rights reserved.

Close

Register the CSF Key in Order Management

Perform the following steps to register the CSF key in Order Management

1. Login to `https://<OM host name>/soa/composer`.
2. Click the **Manage Security** button, located in upper right corner.
3. Enter CSF key recorded from the [Get the CSF Key](#) procedure.
4. Enter your OIC admin credentials (OIC username and password).
5. Reenter the password for confirmation.
6. Click **Register**.
7. Verify the success message is printed on top.

Oracle CPQ Field Setup

Oracle CPQ administrators must setup fields in the CPQ environment and obtain the values for these fields from the Order Management administrator.

Business Unit ID Field

The Business Unit Id or Organization Id uniquely identifies an organization or company in the Oracle Order Management Fusion site. For new sites, configure the field in Fusion. The Commerce process has a “Business Unit ID” attribute at the Transaction level. Administrators must obtain the value for this field from the Order Management administrator and populate the attribute in the Transaction accordingly.

Account Fields

The account information provided during order creation is used for order billing purposes. A sales user can obtain the account information for a customer by entering a customer company name and clicking **Customer Details** (getAccountDetails action). The following fields are populated and are mapped to the associated Order Management fields. These field mappings support the creation of a new order in Order Management based on the information provided in the CPQ Oracle Transaction. The fields should be added to the Customer Details tab of the Transaction layout.

- **Invoice To Party ID** or **Account PartyId**: BillToCustomer-AccountNumber in Order Management
- **Invoice To Company Name** or **Customer Company Name**: BuyingPartyName in Order Management
- **Customer ID**: Associated with the '_customer_id' attribute in CPQ for ABO use.

The screenshot shows the 'Transaction' interface with the 'Customer Details' tab selected. The fields and their values are as follows:

Field	Value
Customer Id	1006
Customer Company Name	Computer Service and Rentals
Party Id	1006
Bill To Site Use Id	1025
SubscriptionProfileId	300100172161473
Invoice To First Name	
Invoice To Last Name	
Invoice To Company Name	
Invoice To Party ID	1006
Ship To Party ID	
Ship To Company Name	

Part Subscription Product Type Field

Note: Beginning in Oracle CPQ 23B and onwards, the '_part_sales_product_type' main document attribute is used to identify a child level subscription product, and 24B and onwards, the '_model_sales_product_type' transaction line level is used to identify a root level subscription product.

Subscription products can have any of the following values:

- SUBSCRIPTION
- PREVENTIVE_ MAINTENANCE
- SOFTWARE_ MAINTENANCE
- COVERAGE
- INCLUDED_ WARRANTY
- product

Layout Fields

This section contains the attributes and actions that need to be added to the Transaction layout.

The following main document attributes are required for Order details.

- Order Number
- Status
- Cancel Reason
- Payment Terms
- Customer Company Name
- Business Unit Name
- Customer Id (_customer_id)
- Invoice to Party ID
- Account PartyIdBill to Site Use Id

The following main document actions are required.

- Create Order
- Cancel Order
- Customer Details
- Get SO Status from FOM
- Save
- Save (Line)

The following sub-document attributes are required in Line Item Grid.

- Document Number
- Status
- Fulfillment Status
- Fulfillment Line Id
- Action Code
- Cancel Reason
- Instance ID
- RootAssetKey

Demo Product Setup (Optional)

Oracle provides an ATO, KIT, and PTO demo products, which is a sample Configuration in CPQ. Use the demo product to understand the functionality available in Oracle Order management.

Note: The packages shown below are optional packages to install ATO, KIT, and PTO demo products. You can download these packages from [CPQ Standard Process – Order Management Integration \(Doc ID 2942508.1\)](#).

File Name	Description
Demo_SetupModels_23B.zip	Migration package to setup ATO, KIT, and PTO demo configuration models.
Parts_Demo_23C.zip -or- Parts_Demo_23B.zip	Part items to support demo products. Notes: <ul style="list-style-type: none">• If you are installing the Parts Package for 23C Demo, use the Parts_Demo_23C.zip file.• If you are installing the Parts Package for 23B Demo, use the Parts_Demo_23B.zip file.
Oracle_BomItemDef.zip and Oracle_BomItemMap.zip	BOM Data Tables to support demo products.

Import the Demo Migration Package

Perform the following steps to install the 23B or 23C Demo Product Migration Package.

1. Download the Demo_SetupModels_23B.zip from [My Oracle Support](#).
2. Navigate to the Admin Home page.
3. Select **Migration** under **Utilities**.
4. Select **Import Package** from the **Select A Mode** drop-down. The Upload Package dialog opens.
5. Click **Browse** and navigate to the Demo_SetupModels_23B.zip.
6. Click **Upload**.
7. Click **Migrate**.
8. When the migration completes, check the migration logs for errors.

Deploy Product Families

1. Navigate to the Admin Home page.
2. Select **Catalog Definition** under **Products**.
3. Select **Product Families** from the Navigation drop-down, and then click **List**.
4. Click **Add**.
5. Select **ATO, KIT, and PTO**, and then click **Add**.
6. Deploy the ATO Product Family
 - a. Select **Deployment Center** from the **ATO** Navigation drop-down, and then click **List**.
 - b. Click **Deploy**.
 - c. Click **Back**.

7. Deploy the KIT Product Family.
 - a. Select **Deployment Center** from the **KIT** Navigation drop-down, and then click **List**.
 - b. Click **Deploy**.
 - c. Click **Back**.
8. Deploy the PTO Product Family.
 - a. Select **Deployment Center** from the **PTO** Navigation drop-down, and then click **List**.
 - b. Click **Deploy**.

Deploy the Home Page

Deploy the Oracle CPQ Home page to make the CPQ FOM Demo Package available on the Oracle CPQ Home page.

Perform the following steps to deploy the Home page.

1. Navigate to the Admin Home page.
2. Select **Home Page** under **Styles and Templates**. The Home Page Setup page opens.

Home Page Setup

Use : Standard Custom

Parts

Name	Restrict Access	Punch-in
Parts Search		

Catalog

Product Family	Restrict Access	Selector Punch-in	Last Refreshed On
ATO			
KIT			
PTO			

[Back to Top](#)

3. Click on the Refresh icon for the **ATO** line.
4. Click on the Refresh icon for the **KIT** line.
5. Click on the Refresh icon for the **PTO** line
6. Click **Deployment Center**. The Deployment Center opens.
7. Click **Deploy**.
8. Click **Refresh** to verify the successful deployment of the Home page.

Home Page Setup

Use : Standard Custom

Parts

Name	Restrict Access	Punch-in
Parts Search		

Catalog

Product Family	Restrict Access	Selector Punch-in	Last Refreshed On
ATO			04/13/2023 2:38 PM
KIT			04/13/2023 2:38 PM
PTO			04/13/2023 2:38 PM

[Back to Top](#)

Install the Parts Package

The Parts_Demo_23B.zip or Parts_Demo_23C.zip package adds parts for the ATO, KIT, and PTO demo products. To view these services in the CPQ sample Configuration, refer to the following examples:

- [Appendix A1: Create Order – Standard Item Workflow](#)
- [Appendix A2: Create Order – Configurable Item Workflow](#)

Note: Monthly Fee, Consumption Fee, and Activation Fee products appear in the calculatedInfo field in the Line Item Grid.

Perform the following steps to install the Parts package.

1. Download Parts_Demo_23B.zip or Parts_Demo_23C.zip from [My Oracle Support](#).



Notes:

- If you are installing the Parts Package for 23C Demo, use the Parts_Demo_23C.zip file.
- If you are installing the Parts Package for 23B Demo, use the Parts_Demo_23B.zip file.

2. Navigate to the Admin Home page.
3. Select **Upload** under **Utilities**. The Upload Files List page opens.
4. Click **Choose File**.
5. Navigate to the downloaded Parts_Demo_23B.zip file, and then click **Add**.
6. Select the Parts_Demo_23B.zip file, and then click **Upload**.
7. Click **Refresh** to check the status of the uploaded file.
8. Check the log corresponding to the uploaded file for errors.

Import BOM Data Tables

Perform the following steps to import the BOM Item Definition and BOM Item Map data tables.

1. Download Oracle_BomItemDef.zip and Oracle_BomItemMap.zip from [My Oracle Support](#).
2. Navigate to the Admin Home page.
3. Click **Data Tables** in the **Developer Tools** section.
4. If a BOM Tables folder does not exist, create a BOM Tables folder.
 - a. Click on the Add Folder icon , in the Admin Toolbar.
 - b. Enter the folder **Name** and **Variable Name**, and then click **OK**.
5. Import the Oracle_BomItemDef.zip file.
 - a. Select **Import** from the navigation panel menu  drop-down.
 - b. Drag and drop the Oracle_BomItemDef.zip file into the Import window or click inside the dashed box to browse and select the Oracle_BomItemDef.zip file.
 - c. Select **BOM Tables** from the **Destination Folder** drop-down.
 - d. Click **Import**.
 - e. Repeat this step to import the Oracle_BomItemMap.zip file.
6. Set options and deploy the Oracle_BomItemDef data table.
 - a. Open the Oracle_BomItemDef data table.
 - b. Select the **Schema** tab.
 - c. Select the **Index** and **Key** options for the **VariableName** field.
 - d. Click **Save**.

- e. Click **Deploy**.
- f. Repeat this step to set options and deploy the Oracle_BomItemMap.zip file.

Activate and Map Data Tables

This procedure is only required the first time BOM tables are installed. Proceed to [Verify](#)

The mapping status of the BOM table displays “Incomplete” for the first activation. After mapping, the status changes to “Complete”.

1. Navigate to the Admin Home page.
2. Select **BOM** under **Products**.
3. Select **BOM Tables** under **BOM Declaration**.
4. Select the **Active** option for the **BOM Item Definition** and **BOM Item Mapping** tables, and then click **Save**.
5. Click on the **BOM Item Definition** Name link.
6. Select the **Oracle_BomItemDef** table in the Table Name drop-down menu.
7. The column mapping is automatic.

Note: If the column names and data types do not match, map the columns manually. Select the appropriate columns from the Column Mapping drop-down menus.

8. Click **Save** when column mapping is complete.
9. Click **Cancel** to return to the Bills of Materials Tables page.
10. Click on the **BOM Item Mapping** Name link.
11. Select the **Oracle_BomItemMap** table in the Table Name drop-down menu.
12. The column mapping is automatic.

Note: If the column names and data types do not match, map the columns manually. Select the appropriate columns from the Column Mapping drop-down menus.

13. Click **Save** when column mapping is complete.

Verify BOM Products

Perform the following steps to verify that all the BOM parts from the BOM package were added to the Oracle CPQ site.

1. Navigate to the Admin Home page.
2. Select **BOM** under **Products**.
3. Select **BOM Root Item List** under **BOM Products**.
4. Click on the variable names to verify that parts were added.
Missing parts are shown in red on this page.

Notes:

- If any parts are missing, add the missing parts to your Oracle CPQ site.
- Make sure to activate BOM tables from Admin-BOM-BOM Tables and Associate/Map them with respective tables.

Order Management Pricing Integration

Standard item, configurable item, and subscription item pricing are derived using the pricing engine in CPQ.

Note:

Beginning in Oracle CPQ 23C, Commerce will support multiple charges for transaction line items. Oracle's Fusion Order Management application requires that each Charge for an Item have a unique combination of Price Type and Charge Type. If you are integrating CPQ with Order Management, you will need to define a different Charge Type to distinguish Charges with the same Price Type but different Price Periods.

Enable Pricing

Perform the following steps in Oracle CPQ to enable Pricing.

1. Navigate to the Admin Home page.
2. Click **Pricing Portal** in the **Products** section.
3. Click on the **Pricing Options** card.
4. Click **Edit**.
5. Select **Version 4** from the **Pricing Behavior** drop-down.
6. Click **Update**.

Pricing Related Utility BML

The **Populate Charges** BML populates calculated pricing information for charges array. This BML is invoked from the Commerce library 'system_calculations_process' when it is invoked from Transaction Line Advanced Default - After Formulas.

Prices Array Set

The Prices Array Set (`oRCL_charges`) is present in line level it is used to hold the prices and charges information of an item.

This arrayset is used in OIC mappings to generate the payload for Fusion Order Management integration. Commerce library "oRCL_fOM_populateCharges" included in the Commerce Standard Process is used to populate this arrayset with all the applicable charges defined for an item in the pricing portal.

The following attributes are added in the array set.

Attribute Name	Variable Name	Description	Expected Value
Charge Name	<code>oRCL_chargeName</code>	Name of the charges	Example Values <ul style="list-style-type: none">• Sale Price• Recurring Sale Price
List Price	<code>oRCL_listPrice</code>	List Price of the item	
Charge Type	<code>oRCL_chargeType</code>	Charge Type of the item. This attribute name is no longer a standard attribute in CPQ release 23C	ORA_SALE
Charge Type	<code>_s_oRCL_chargeType</code>	Charge Type of the item. This new attribute will replace <code>oRCL_chargeType</code> attribute from CPQ release 23C onwards.	
Net Price	<code>oRCL_netPrice</code>	The price paid by the customer for this item.	
Sequence Number	<code>oRCL_chargeSequenceNumber</code>	Sequence number of the charges	
Apply To	<code>oRCL_applyTo</code>	On which type of price it is applied	Price
Primary Charge	<code>oRCL_primaryCharge</code>	Primary charge is used to determine whether the charge is primary or not.	true / false
List Amount	<code>oRCL_listAmount</code>	The extended list amount for this line.	

Attribute Name	Variable Name	Description	Expected Value
Net Amount	oRCL_netAmount	The extended list amount minus any discounts for this line.	
Custom Discount Amount	oRCL_customDiscountAmount	The amount by which to adjust the list price because of a custom discount.	
Discount Amount	oRCL_discountAmount	The discount amount for this line.	
Contract List Value	oRCL_contractListValue	The total of all contract list value of this line, including all one-time and recurring charges	
Contract Discount Amount	oRCL_contractDiscountAmount	The total discount of this line including all one-time and recurring charges.	
Contract Net Amount	oRCL_contractNetValue	The total contract value of this line including all one-time and recurring charges.	
Price Type	_s_oRCL_priceType	Indicates whether the amount on this line is charged once or periodically. This attribute is available from CPQ Release 23C onwards.	
Price Period	_s_oRCL_pricePeriod	The time period, frequency or recurrence with which the price or charge is levied. This attribute is available from CPQ Release 23C onwards.	
Price Unit of Measure	_s_oRCL_priceUOM	The unit of measure of usage for usage charges, which generally differs from the unit of measure used for the parent product. This attribute is available from CPQ Release 23C onwards.	
Custom Discount Type	_s_oRCL_customDiscountType	The type of custom discount applied to this charge. Typically this may be an override price, a discount value, or a discount percentage. This attribute is available from CPQ Release 23C onwards.	
Charge Definition Integration ID	_s_oRCL_chargeDefIntegrationId	The charge definition code in the integrated order management system. This was added in 23D.	

Oracle CPQ Account Integration

This section identifies the library functions that support account integration and the manual changes that administrators must make to the INT_SYSTEM_DETAILS Data Table and the INT_SYSTEM_TEMPLATES Data Table to support account integration.

Note: For more details about Account Integration with Oracle Customer Data Management (CDM), refer to CPQ-CDM Integration Whitepaper on [My Oracle Support](#) under CPQ to Fusion Financials Integration (Doc ID 2012010.1).

Library Functions

The library functions within this section support the CDM integration by retrieving account details.

String getTemplateLocation(String system, String operation)

Queries the template location from the INT_SYSTEM_TEMPLATES Data Table based on the system and operation. This is a Commerce library function.

The Return Type, input information, and attributes used by this library function are shown in the following image.

The screenshot shows the 'Commerce BML Library Function Editor: Properties & Parameters' window. It contains the following information:

- Name:** getTemplateLocation
- Variable Name:** getTemplateLocation
- Description:** Queries the template location from the datatable INT_SYSTEM_TEMPLATES based on the System and Operation.
- Return Type:** String

#	Parameter Name	Parameter Type
1	system	String
2	operation	String

The 'Function Editor' section below shows three attribute panels: 'Main Document Attribute', 'Sub Document Attribute', and 'System Attribute'. Each panel has an 'Add Attributes' button. The 'Sub Document Attribute' panel currently contains 'transactionLine'.

String invokeWebService(String system, String soapReq)

This is a Commerce library function that invokes Web Services and returns the response.

The Return Type, input information, and attributes used by this library function are shown in the following image.

The screenshot shows the 'Commerce BML Library Function Editor: Properties & Parameters' window. It contains the following information:

- Name:** invokeWebService
- Variable Name:** invokeWebService
- Description:** Invokes Web service and returns the response.
- Return Type:** String

#	Parameter Name	Parameter Type
1	system	String
2	soapReq	String

Refer to [Appendix D: Miscellaneous Commerce Library Functions](#) for BML scripts.

Manual Data Table Changes

The INT_SYSTEM_DETAILS and INT_SYSTEM_TEMPLATES data tables are added to the CPQ site for account integration.

INT_SYSTEM_DETAILS

System (Key)	Username	Endpoint	Description
TCA-OrgService	<Enter the username here to call the web service endpoint>	<Enter the web service endpoint to call the service related to TCA>	TCA Find Organization details
TCA-AccService	<Enter the username here to call the web service endpoint>	<Enter the web service endpoint to call the service related to TCA>	TCA customer Account

As shown in the following image, administrators must manually select the **Key** option for the **System** column.

Edit Data Table: INT_SYSTEM_DETAILS

Data Schema Details

View ▼

Add Foreign Key
Add Relationship
Remove Relationship

#	Index	Key	Type	Name	Description	Date Added	Date Modified	Validation Type
1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	String	System		10/24/2017 10:22 AM	02/01/2021 8:19 AM	None
2	<input type="checkbox"/>	<input type="checkbox"/>	String	Description		10/24/2017 10:22 AM	02/01/2021 8:19 AM	None
3	<input type="checkbox"/>	<input type="checkbox"/>	String	Username		10/24/2017 10:22 AM	02/01/2021 8:19 AM	None
4	<input type="checkbox"/>	<input type="checkbox"/>	Secure	Password		10/24/2017 10:22 AM	03/15/2021 2:15 PM	None
5	<input type="checkbox"/>	<input type="checkbox"/>	Integer	MaxLinesInPayload		10/24/2017 10:22 AM	02/01/2021 8:19 AM	None
6	<input type="checkbox"/>	<input type="checkbox"/>	String	Endpoint		02/01/2021 8:19 AM	02/01/2021 8:19 AM	None
7	<input type="checkbox"/>	<input type="checkbox"/>	String	SoapEndpoint		10/24/2017 10:22 AM	02/01/2021 8:19 AM	None

INT_SYSTEM_TEMPLATES

System (Key)	Operation (Key)	Templates
TCA-OrgService	FindOrg	<Enter the template URL path that is uploaded in File Manager>
TCA-AccService	FindAcc	<Enter the template URL path that is uploaded in File Manager>

As shown in the following image, administrators must manually select the **Key** option for the **System and Operation** columns.

Edit Data Table: INT_SYSTEM_TEMPLATES

Data Schema Details

View ▼

Add Foreign Key
Add Relationship
Remove Relationship

#	Index	Key	Type	Name	Description	Date Added	Date Modified	Validation Type	Validation Mapping
1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	String	System		10/24/2017 10:22 AM	02/01/2021 8:19 AM	None	
2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	String	Operation		10/24/2017 10:22 AM	02/01/2021 8:19 AM	None	
3	<input type="checkbox"/>	<input type="checkbox"/>	String	Template		10/24/2017 10:22 AM	02/01/2021 8:19 AM	None	

Add Template Dependencies to File Manager

Oracle CPQ administrators must add the Find Organization payload template file and the Customer Account payload template file to File Manager. These template files support account integration.

Perform the following steps to add the template dependencies to File Manager.

1. Download the **findOrganizationPayload.txt** and **customerAccountPayload.txt** payload template files from [My Oracle Support](#).
2. Navigate to the Admin Home page.
3. Navigate to **File Manager** under **Utilities** category and create a new folder named **TCA**.
4. Click **Browse** under **Add Files**. The Choose File to upload dialog opens.
5. Navigate to the findOrganizationPayload.txt file and click **Open**.
6. Click **Add File**. The findOrganizationPayload.txt file displays in File Manager.
7. Complete steps 2-6 for the customerAccountPayload.txt file.

Note: To view the BML included in the payload template files, refer to [Appendix C: Payload Template Files](#).

Appendix A: Workflows

Appendix A1: Create Order – Standard Item Workflow

OIC is the middleware used to establish an integration between CPQ and Order Management. Once this integration is established, sales users can use CPQ to create a Transaction and invoke OIC integration to create an order in Order Management. Perform the following steps in Oracle CPQ to create a standard order.

1. Navigate to the Transaction Manager.
2. Click **New Transaction**. The Transaction page opens.
3. Add Standard products to the line item using Quick Add.

The screenshot shows the Oracle CPQ Transaction Manager interface. The top navigation bar includes buttons for Refresh, Save, Submit, Create Order, Version Transaction, Customer Details, Alternate Order, and Update Line Item. The main content area is divided into three tabs: Transaction Details, Customer Details, and Pricing Details. The Transaction Details tab is active, showing fields for Transaction Name (11), Transaction Number (CPQ-171-36677682), Version (1), Business Unit Name, Owner (Sales User), Created Date (02/02/2021), Last Updated By (Super User), Order Number, Status (Created), Last Updated (04/01/2021 9:40 AM), Default Request Date, Order Date, and Ordered By. Below the transaction details is a section for Line Items (LIG) with a toolbar containing icons for View, Add, Edit, Delete, and Freeze. A table of line items is displayed with columns for #Doc, Product #, Quantity, Status, Fulfillment Status, Contract Start Date, Contract End Date, Req Date, Action Code, and Instance ID. The table contains two rows of data for product AS92888. The first row has a quantity of 1, status of Created, fulfillment status of Created, contract start date of 02/02/2021, contract end date of 02/01/2022, req date of 02/02/2021, action code of Add, and instance ID of abo_8242ff68-d621-4b76-9a68-1de9fcc96bac. The second row has a quantity of 1, status of Created, fulfillment status of Created, contract start date of 04/01/2021, contract end date of 03/31/2022, req date of 04/01/2021, action code of Add, and instance ID of abo_b607dc8e-fa5f-48f9-8685-6702d6e5c181. The bottom of the screen shows pagination information: Page 1 of 1 (1-2 of 2 items) and buttons for Add Line Item and Copy Line Items.

4. Edit the Transaction and Transaction Line details as required.
5. Click **Save**.
6. Click **Create Order**.
7. Order Number, Order Status, and Fulfillment Id for each line are returned from Order Management on successful creation of order.

The screenshot shows the Oracle CPQ Transaction Manager interface after the transaction has been processed. The top navigation bar is the same as in the previous screenshot. The Transaction Details tab is active, showing fields for Transaction Name (11), Transaction Number (CPQ-171-36677682), Version (1), Business Unit Name, Owner (Sales User), Created Date (02/02/2021), Last Updated By (Super User), Order Number (527641), Status (Processing), Last Updated (04/01/2021 9:40 AM), Default Request Date, Order Date, and Ordered By. The LIG section is also present with the same toolbar. The table of line items now shows updated fulfillment status. The first row has a quantity of 1, status of Created, fulfillment status of Being Fulfilled, contract start date of 02/02/2021, contract end date of 02/01/2022, req date of 02/02/2021, action code of Add, and fulfillment ID of 300100551483970. The second row has a quantity of 1, status of Created, fulfillment status of Being Fulfilled, contract start date of 04/01/2021, contract end date of 03/31/2022, req date of 04/01/2021, action code of Add, and fulfillment ID of 300100551483977. The bottom of the screen shows pagination information: Page 1 of 1 (1-2 of 2 items) and buttons for Add Line Item and Copy Line Items.

Note: Use Oracle Order Management to check the order creation using returned Order Number.

Appendix A2: Create Order – Configurable Item Workflow

Perform the following steps in Oracle CPQ to create a configurable item order.

1. Open Model Configuration.
2. Select the root model.
3. Select the **addOptionalClass** check box.
4. Click **Update**. This will display optional class available for the root model.

Model Configuration

Update Start Over Create Transaction

OM_SI_ATO_MODEL
 addOptionalClass

Select	Lead Time	Comment	Price	Extended Description 2	Part Number	Extended Description 1	Description	Units	Direct Buy	Quantity	Product Group
<input checked="" type="checkbox"/>			Not Defined		OM_SI_ATO_MODEL					1	
<input checked="" type="checkbox"/>			Not Defined		OM_SI_ATO_MODEL1_OC1					1	
<input checked="" type="checkbox"/>			On Request	View...	OM_SI_ATO_MODEL1_OC1_OI1	View...			Direct Buy	1	

5. Click **Create Transaction**. This will create new transaction with the configurable items.

Transaction

Refresh Save Submit Create Order Version Transaction Customer Details Alternate Order Update Line Item

Transaction Details Customer Details Pricing Details

Transaction Name: Owner: Sales User * Status: Created

Transaction Number: CPQ-406-36717898 Created Date: 03/26/2021 Last Updated: 03/26/2021 11:17 AM

Version: 1 Last Updated By: Sales User Default Request Date:

Business Unit Name: Vision Operations Order Number: Order Date: Ordered By:

LIG

#Doc	Product #	Quantity	Status	Fulfillment Status	Contract Start Date	Contract End Date	Req Date	Action Code	Fulfillment ID	Change Reason	CF
2	OM_SI_ATO_MODEL	1	Created	Created	03/26/2021	03/25/2022		Add			
3	OM_SI_ATO_MODEL1_OC1	1	Created		03/26/2021	03/25/2022		Add			
4	OM_SI_ATO_MODEL1_OC1_OI1	12	Created		03/26/2021	03/25/2022		Add			

Page 1 of 1 (1-3 of 3 items) | < 1 > X

Add Line Item Copy Line Items

6. Edit the Transaction and Transaction Line details as required.
7. Click **Save**.
8. Click **Create Order**.
9. The Order Number, Order Status, and Fulfillment Id for each line are returned from Order Management on successful creation of order.

Notes:

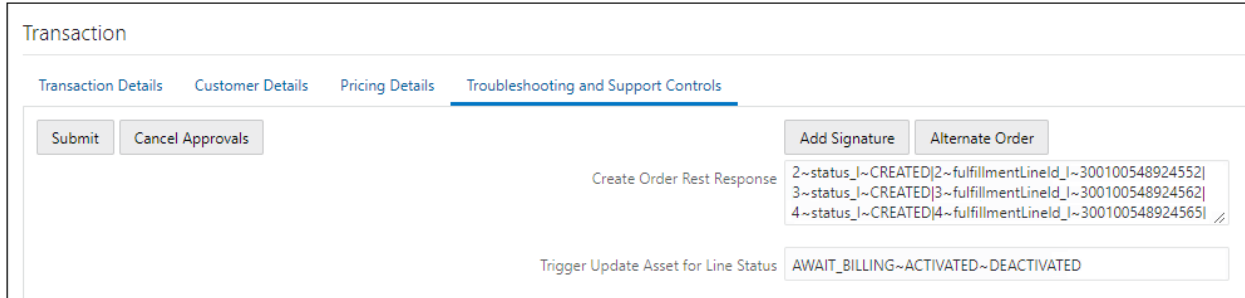
- Use Order Management to check the order creation using the returned Order Number.
- BOM setup is required for the configuration.

Appendix A3: Auto Sync Status Workflow

Oracle Order Management allows customers to configure Business Events to be raised during various stages of fulfilment. We subscribe to these Business events through OIC integrations and invoke CPQ REST APIs to update vital information.

This functionality synchronizes the Order and Order Line status from Order Management to CPQ.

1. Create a transaction in CPQ.
2. Record the **Trigger Update Asset for Line Status** value.



The screenshot shows the 'Transaction' interface in CPQ. The 'Troubleshooting and Support Controls' tab is active. The 'Trigger Update Asset for Line Status' field is populated with the value 'AWAIT_BILLING~ACTIVATED~DEACTIVATED'. The 'Create Order Rest Response' field contains three lines of JSON data: '2~status_I~CREATED|2~fulfillmentLineId_I~300100548924552|', '3~status_I~CREATED|3~fulfillmentLineId_I~300100548924562|', and '4~status_I~CREATED|4~fulfillmentLineId_I~300100548924565|'. The interface also includes buttons for 'Submit', 'Cancel Approvals', 'Add Signature', and 'Alternate Order'.

3. Click **Create Order** to create an order in Order Management. Refer to [Appendix A1: Create Order – Standard Item Workflow](#).
4. Use Order Management to check the order creation using the Order Number returned on order creation.
5. When the Order/ Order Line status changes on Order Management, it should reflect the updated Status attribute in CPQ based on Order Status Mapping in OIC. Refer to [Appendix E: CPQ-OM Status Mapping](#).
6. If the line status in CPQ is one of the values in 'Trigger Update Asset for Line Status', the Update Asset functionality will also be triggered, and Fulfilment Status will be updated to 'Fulfilled'.
7. Open the Subscription Workbench page and observe the asset is created for the Fulfilled lines.

Appendix A4: Full Order Cancellation

This workflow supports the cancellation of full order (all lines).

1. Navigate to the commerce and open the transaction in which order was created.
2. Validate there is an order number associated with the transaction.
3. Validate the order is in processing status.
4. Click the **Select All** option in the Line Item Grid.
5. (Optional) Select the cancellation reason in the transaction details.
6. Click **Cancel Order**.

The screenshot displays the 'Transaction' interface with the 'Cancel Order' button highlighted in red. Below the transaction details is the 'LIG' (Line Item Grid) section. The LIG table has the following data:

#Doc	Product #	Quantity	Status	Fulfillment Status	Contract Start Date	Contract End Date	Req Date	Action Code	Instance ID
<input checked="" type="checkbox"/>	3	AS54888	1	Created	Created	04/01/2021	03/31/2022	Add	abo_2a42a210-846b-4088-9b12-406e39cace35
<input checked="" type="checkbox"/>	4	AS92888	1	Created	Created	04/01/2021	03/31/2022	Add	abo_4aa00005-7e26-4b9b-ae32-443223974e26

The LIG table also includes a toolbar with various icons and buttons like 'Customer Assets' and 'Get SO Status from FOM'. The page footer shows 'Page 1 of 1 (1-3 of 3 items)' and '2 selected'.

7. After the cancellation action is complete, verify the line item status is updated to Cancellation Requested or Cancelled.

Appendix A5 Partial Order Cancellation

This workflow supports the cancellation partial order (selected line items).

1. Navigate to the commerce and open the transaction in which order was created.
2. Validate there is an order number associated with the transaction.
3. Validate the order is in processing status.
4. Select the line items to cancel in the Line Item Grid.
5. (Optional) Select a cancellation reason for each of the lines to be cancelled in the Line Item Grid.
6. Click **Cancel Order**.

The screenshot shows the 'Transaction' page with a 'Cancel Order' button highlighted. Below it is the 'LIG' (Line Item Grid) with a table of line items. The first row is selected, and its columns are highlighted with a red box.

#Doc	Product #	Quantity	Status	Fulfillment Status	Contract Start Date	Contract End Date	Req Date	Action Code	Instance ID
<input checked="" type="checkbox"/>	3	AS54888	1	Created	Created	04/01/2021	03/31/2022	Add	abo_2a42a210-84bb-4088-9b12-406e39cace35
<input type="checkbox"/>	4	AS92888	1	Created	Created	04/01/2021	03/31/2022	Add	abo_4aa00005-7e26-4b9b-ae32-443223974e26

7. After the cancellation action is complete, verify the status is updated to 'Cancellation Requested' or 'Canceled' for the selected lines.

Notes:

- Cancellation is not allowed for line with status such as 'BILLED', 'SHIPPED', 'ACTIVATED', 'FULFILLED', 'AWAIT_BILLING', 'DELIVERED', 'AWAIT_ASSET_CREATION', 'SHIPPED' etc.
- An order which is in DRAFT status cannot be cancelled.
- For configurable items only root items can be cancelled.

Appendix A6: Terminate Subscription Product Workflow

Sales users can use the Customer Assets page to terminate subscription products that are in an active state in OSS.

1. Create a transaction in CPQ.
2. Navigate to the Customer Assets page.
3. Select the asset that corresponds to the subscription to terminate.
4. Click **Terminate**. The Transaction page opens.
5. Provide Contract End date, Change Reason and Change Method for the Transaction Line.
6. Edit the Transaction and other Transaction Line details as needed.
7. Click **Save** and **Create Order**.

Appendix A7: Renewal Subscription Workflow

OSS Renewal consolidation process picks up all the eligible subscription products to get renewed and it submits an event for OIC to create the renewal quote.

Note: With **Oracle CPQ 24B release**, renewal subscription is supporting for the following 2 item structures:

- a Single Subscription Product with/without coverage subscription product.
 - b PTO model with only Optional Items.
1. OSS Ess job “Automatically Renew Eligible Subscriptions” will consolidate all the eligible subscription products and creates a draft subscription.
 2. OIC “OSS CPQ CreateRenewalQuote” will listen for Subscription creation event (ORA_DRAFT status):
 - a It creates an opportunity.
 - b Creates Quote Header and all lines for all the received renewal subscription products.
 - c Generates the proposal and prints the document.
 - d Syncs the Revenue lines.
 3. Open the transaction in CPQ:
 - a In OIC, navigate to Observe tab and Instances section and then click on OSS CPQ CreateRenewalQuote
 - b Open the Activity Stream for the latest Subscription event.
 - c Navigate to Step “**While CreateQuoteHeaderRetry**”, expand the iteration and View step “**Logger LogCreatedQuoteHeader**” to get the transaction id.

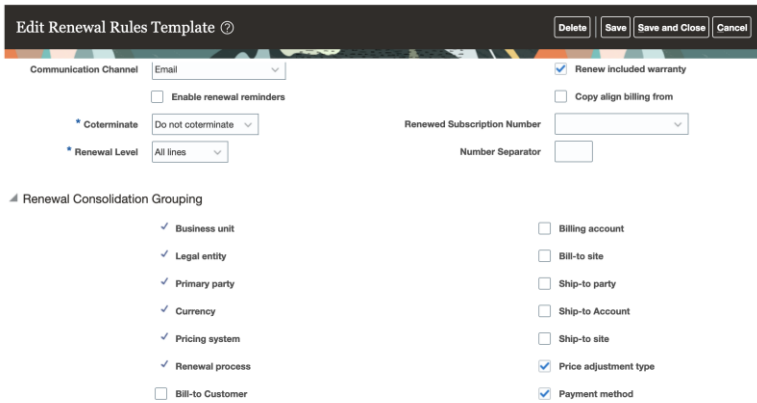
The screenshot displays the Oracle CPQ Activity Stream interface. On the left, there are tabs for 'Design', 'Deploy', and 'Observe'. Below these is a table with columns 'Instance ID' and 'Business Identif'. The main area shows an 'Activity stream' for Instance ID: o58TeODzEe6ADAIQa6PPWw. The tracing level is set to 'Audit'. The activity stream shows a sequence of events: 'While CreateQuoteHeaderRetry' (1 iteration), 'Invoke CreateQuoteHeader', and 'Logger LogCreatedQuoteHeader'. The 'Logger LogCreatedQuoteHeader' event is highlighted, showing a JSON payload: `{Created Quote Header : '38642018'}`. Other events include 'Message received by Assignment ResetRetryVariables2', 'Logger LogGettingAssetKeys', and 'Data Mapping completed'.

- d Open the transaction.
4. Root level transaction line contract start and end dates will be stamped for all of the child level transaction lines.
 5. Edit the Transaction and other Transaction Line details as needed.

6. Click **Save** and **Create Order**.

Note: With **Oracle CPQ 24B release**, renewal subscription is supporting for the following 2 item structures:

- a. Please check used OSS Renewal Template to get Renewal Consolidation grouping parameters and based on which it consolidates all the eligible subscription products to renew.



Edit Renewal Rules Template [Delete] [Save] [Save and Close] [Cancel]

Communication Channel: Email

Enable renewal reminders

* Coterminate: Do not coterminate

* Renewal Level: All lines

Renew included warranty

Copy align billing from

Renewed Subscription Number: []

Number Separator: []

Renewal Consolidation Grouping

- Business unit
- Legal entity
- Primary party
- Currency
- Pricing system
- Renewal process
- Bill-to Customer
- Billing account
- Bill-to site
- Ship-to party
- Ship-to Account
- Ship-to site
- Price adjustment type
- Payment method

- b. Please make sure used OSS Renewal Template should have the following values set:

- 1) Either Renewal Process should be Manual or Customer Acceptance Required should be Required.
- 2) Minimum Renewal Duration is as low as possible (ex: 10 days).
- 3) Consolidate subscriptions should be checked.
- 4) Internal Approved to Required.

Appendix B: BML

BML – Process Create Order Response

This BML is used to process the rest response received from Create Order ICS integration. It is associated with Create Order(`order_start`) action.

BML Input Attributes:

- Transaction: `createOrderRestResponse`
- Transaction Line: `_document_number, _line_bom_parent_id`

```
retStr = "";
if(createOrderRestResponse <> ""){
    for line in transactionLine {
        if (line._line_bom_parent_id == "") {
            retStr = retStr + line._document_number + "~fulfillmentStatus_1~BEING_FULFILLED|";
        }
    }
    retStr = retStr + createOrderRestResponse;
}
return retStr;
```

BML - Get Account Details

The BML for the Get Account Details action is used to support account integration. When sales users enter a customer company name and click Customer Details, the BML retrieves the `PrimaryPartyId`, `BillToAccountId`, and `BillToSiteUseld` fields from the Oracle EBS Customer Data Management application:

```
//1. Get Template Location
returnPartyId = "";
defaultErrorMessage = "";
errorString = "Error in TCA Service";
organizationSoapResponse = "";
if(isnull(_transaction_customer_id) OR _transaction_customer_id == "")
{
    system="TCA-OrgService";
    operation="FindOrg";

    organizationSoapRequestLocation = commerce.getTemplateLocation(system, operation);

    //payload = commerce.getUserAttributes(system);

    organizationSoapRequest=applytemplate(organizationSoapRequestLocation ,dict("string"),
    defaultErrorMessage);
    print "request";
    print organizationSoapRequest;
    print "response";
    organizationSoapResponse = commerce.invokeWebService(system, organizationSoapRequest );
    print organizationSoapResponse;

    xpaths = string[1];
    xpaths[0] = "//ns2:PartyId";

    output = readxmlsingle(organizationSoapResponse, xpaths);
    if (containskey(output,xpaths[0]))
        { returnPartyId = get(output,xpaths[0]); }
    else {
        returnPartyId = "Check if customer company name is valid & also TCA service is up.";
        returnSiteNumber = "Check if customer company name is valid & also TCA service is up" ;
    }
}
```

```

        return
"1~invoiceToPartyID_t~"+returnPartyId+"|1~_s_accountPartyId_t~"+returnPartyId+"|"+"1~billToSiteUseId_t~"+r
eturnSiteNumber+"|";
}
}else{
returnPartyId = _transaction_customer_id;
}

// Get Template Location
system="TCA-AccService";
operation="FindAcc";

customerAccountSoapRequestLocation = commerce.getTemplateLocation(system, operation);
//payload1 = commerce.getUserAttributes(system,returnPartyId);
payload1 = dict("string");
put(payload1,"returnPartyId",returnPartyId);

customerAccountSoapRequest=applytemplate(customerAccountSoapRequestLocation,payload1,
defaultErrorMessage);
print customerAccountSoapRequest;

customerAccountsoapResponse = commerce.invokeWebService("TCA-AccService", customerAccountSoapRequest);
print customerAccountsoapResponse ;

xpathAcct = string[1];
xpathAcct[0] = "//ns2:CustomerAccountId";
returnAccountNumber = "" ;
outputAcct = readxmlsingle(customerAccountsoapResponse, xpathAcct);
if (containskey(outputAcct,xpathAcct[0]))
    { returnAccountNumber = get(outputAcct,xpathAcct[0]); }
else {
        returnAccountNumber = "Check if customer company name is valid & also TCA service is
up";
        returnSiteNumber = "Check if customer company name is valid & also TCA service is up ";
return
"1~_s_accountPartyNumber_t~"+returnAccountNumber+"|"+"1~billToSiteUseId_t~"+returnSiteNumber +"|";
}
returnBillToSiteUseId="";
xpathbilltositeuseid = string[1];
xpathbilltositeuseid[0] =
"//ns2:Value/ns2:CustomerAccountSite/ns2:CustomerAccountSiteUse[ns2:SiteUseCode='BILL_TO' and
ns2:PrimaryFlag='true']/ns2:SiteUseId";
outputbilltositeuseid = readxmlsingle(customerAccountsoapResponse, xpathbilltositeuseid);
if (containskey(outputbilltositeuseid ,xpathbilltositeuseid[0])) {
        returnBillToSiteUseId= get(outputbilltositeuseid,xpathbilltositeuseid[0]); }
else {
        returnBillToSiteUseId= "Check if customer company name is valid & also TCA service is up. ";
}
xpathCustomerAccountSiteUse = string[1];
// get the XML Element called Customer AccountSiteUse where primary is true and use is BILL TO
xpathCustomerAccountSiteUse[0] =
"//ns2:Value/ns2:CustomerAccountSite/ns2:CustomerAccountSiteUse[ns2:SiteUseCode='BILL_TO' and
ns2:PrimaryFlag='true']";

outputCustomerAccountSiteUse = readxmlsingle(customerAccountsoapResponse, xpathCustomerAccountSiteUse);
returnSiteId = "";
if (containsKey(outputCustomerAccountSiteUse,xpathCustomerAccountSiteUse[0]))
{
    CustomerAccountSiteUseXmlFragment = get(outputCustomerAccountSiteUse,xpathCustomerAccountSiteUse[0]);
    xpath1 = string[1];

```

```

xpath1[0] = "//ns2:CustomerAccountSiteId";
output1 = readxmlsingle(CustomerAccountSiteUseXmlFragment,xpath1);
if (containsKey(output1,xpath1[0]))
{
    returnSiteId = get(output1,xpath1[0]);
}
}
if (returnSiteId == "") {
    returnSiteId = errorString;
}

partySiteId = "";
xpathCustomerAccountSite = string[1];
// get the XML Element called Customer AccountSite where ID = returnSiteId
xpathCustomerAccountSite[0] = "//ns2:Value/ns2:CustomerAccountSite[ns2:CustomerAccountSiteId=" +
returnSiteId + "]";

outputCustomerAccountSite = readxmlsingle(customerAccountsSoapResponse, xpathCustomerAccountSite);

if (containsKey(outputCustomerAccountSite,xpathCustomerAccountSite[0]))
{
    CustomerAccountSiteXmlFragment = get(outputCustomerAccountSite,xpathCustomerAccountSite[0]);

    xpath2 = string[1];
    xpath2[0] = "//ns2:PartySiteId";
    output2 = readxmlsingle(CustomerAccountSiteXmlFragment,xpath2);
    if (containsKey(output2,xpath2[0]))
    {
        partySiteId = get(output2,xpath2[0]);
    }

    if (partySiteId == "") {
        partySiteId = errorString;
    }
}

partySiteNumber = "";
xpathPartySiteNumber = string[1];
// get the XML Element called Customer AccountSite where ID = returnSiteId
xpathPartySiteNumber[0] = "//ns2:Value/ns2:PartySite[ns1:PartySiteId=" + partySiteId + "]";

outputPartySiteNumber = readxmlsingle(organizationSoapResponse, xpathPartySiteNumber);

if (containsKey(outputPartySiteNumber,xpathPartySiteNumber[0]))
{
    PartySiteNumberXmlFragment = get(outputPartySiteNumber,xpathPartySiteNumber[0]);

    xpath3 = string[1];
    xpath3[0] = "//ns1:PartySiteNumber";
    output3 = readxmlsingle(PartySiteNumberXmlFragment,xpath3);
    if (containsKey(output3,xpath3[0]))
    {
        partySiteNumber = get(output3,xpath3[0]);
    }
}

return
"1~invoiceToPartyID_t~"+returnPartyId+"|1~_s_accountPartyId_t~"+returnPartyId+"|"+"1~_s_accountPartyNumber

```



```
_t~"+returnAccountNumber+"|"+"1~billToSiteUseId_t~"+returnBillToSiteUseId  
+"|"+"1~_customer_id~"+returnPartyId+"|";
```

Appendix C: Payload Template Files

The BML associated with the payload template files referenced in Add Template Dependencies to File Manager follows.

findOrganizationPayload.txt

```
<?xml version="1.0"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body
xmlns:ns1="http://xmlns.oracle.com/apps/cdm/foundation/parties/organizationService/applicationModule/types/">
    <ns1:findOrganization>
      <ns1:findCriteria xmlns:ns2="http://xmlns.oracle.com/adf/svc/types/">
        <ns2:filter>
          <ns2:conjunction>And</ns2:conjunction>
          <ns2:group>
            <ns2:conjunction>And</ns2:conjunction>
            <ns2:item>
              <ns2:conjunction>And</ns2:conjunction>
              <ns2:attribute>PartyName</ns2:attribute>
              <ns2:operator>=</ns2:operator>
              <ns2:value>{{customerCompanyName_t}}</ns2:value>
            </ns2:item>
          </ns2:group>
        </ns2:filter>
      </ns1:findCriteria>
      <ns1:findControl xmlns:ns3="http://xmlns.oracle.com/adf/svc/types/">
        <ns3:retrieveAllTranslations/>
      </ns1:findControl>
    </ns1:findOrganization>
  </soap:Body>
</soap:Envelope>
```

customerAccountPayload.txt

```
<?xml version="1.0"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ns1="http://xmlns.oracle.com/apps/cdm/foundation/parties/customerAccountService/applicationModule/types/"
xmlns:ns2="http://xmlns.oracle.com/adf/svc/types/">
  <soapenv:Body>
    <ns1:findCustomerAccount>
      <ns1:findCriteria>
        <ns2:filter>
          <ns2:conjunction>And</ns2:conjunction>
          <ns2:group>
            <ns2:conjunction>And</ns2:conjunction>
            <ns2:item>
              <ns2:conjunction>And</ns2:conjunction>
              <ns2:attribute>PartyId</ns2:attribute>
              <ns2:operator>=</ns2:operator>
              <ns2:value>{{returnPartyId}}</ns2:value>
            </ns2:item>
          </ns2:group>
        </ns2:filter>
      </ns1:findCriteria>
      <ns1:findControl>
        <ns2:retrieveAllTranslations>false</ns2:retrieveAllTranslations>
      </ns1:findControl>
    </ns1:findCustomerAccount>
  </soapenv:Body>
</soapenv:Envelope>
```

Appendix D: Miscellaneous Commerce Library Functions

The Oracle CPQ Order Management package adds several library functions to the Commerce process. This topic provides the BML associated with each of the library functions.

String getTemplateLocation(String system, String operation)

The code for this library function is provided the following for reference.

```
//1. Get Template File
templateUrl = "";
//bmqL query
resultSet = bmqL("Select Template from INT_SYSTEM_TEMPLATES where System = $system and Operation =
$operation");
//loop through the records
for record in resultSet {
templateUrl = get(record,"Template");
}
temp=split(templateUrl,"image")
return temp[1];
```

String invokeWebService(String system, String soapReq)

The code for this library function is provided the following for reference.

```
//1. Get webservice endpoint for the system
resultSet = bmqL("Select Endpoint,Username,Password from INT_SYSTEM_DETAILS where System = $system");
endpoint = "";
username = "";
password = "";
//loop through the records
for record in resultSet {
endpoint = get(record,"Endpoint");
username = get(record,"Username");
password = get(record,"Password");
}
//2. Invoke the web serviceheader
Values = dict("string");
put(headerValues, "Content-Type", "text/xml; charset=utf-8");
encodeCredential = encodebase64(username+"."+password);auth = "Basic " + encodeCredential;
put(headerValues,"Authorization",auth);
errorString = "Error in "+system+" invocation";
soapResponse= urldatabypost(endPoint , soapReq,errorString,headerValues,true);
// sends the soap call and returns response to variable.
print "going to print soapResponse";
//3. Return the response
return soapResponse;
```


Appendix E: CPQ-OM Status Mapping

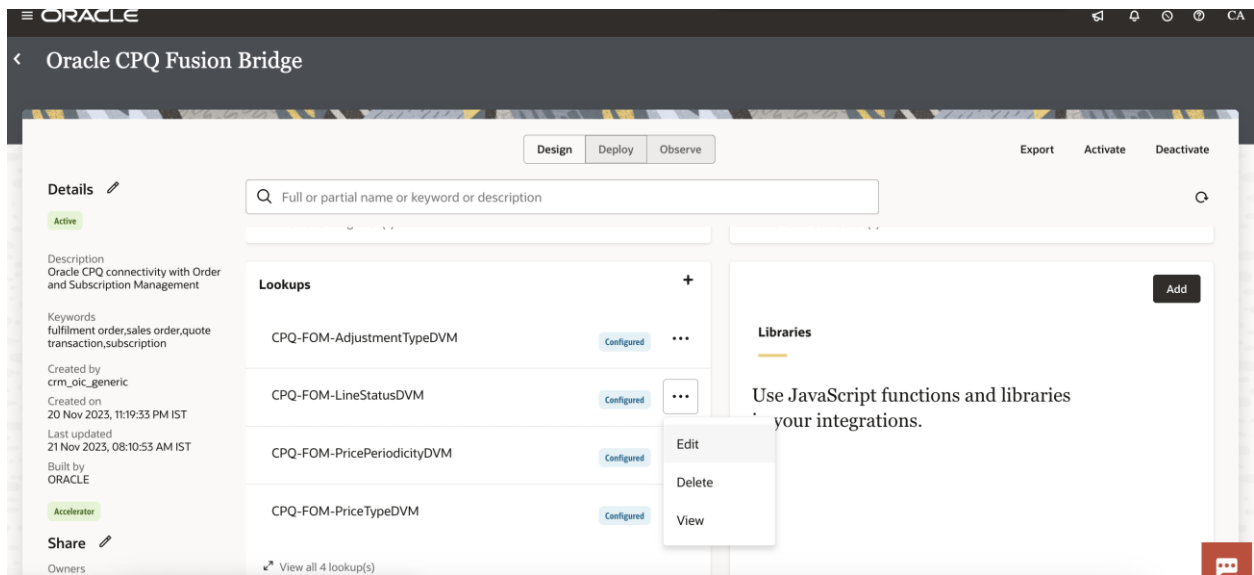
Map CPQ Line Status to Order Management Line Status

Update 24A or later:

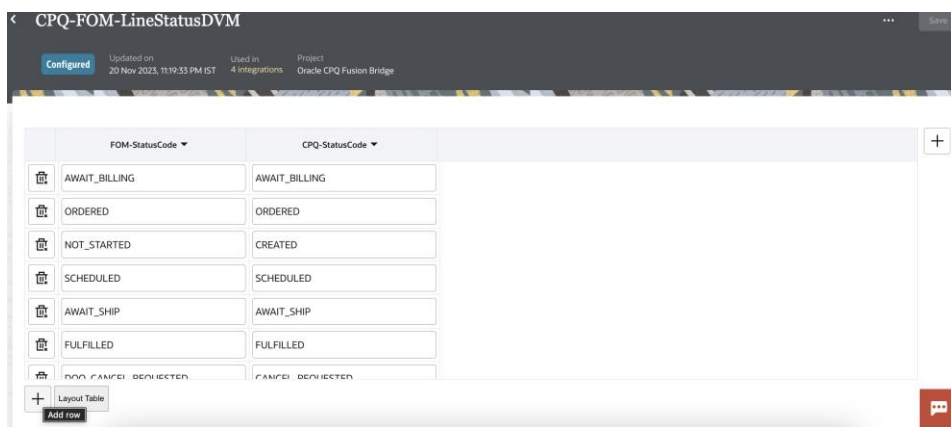
Administrators can use the CPQ-FOM-LineStatusDVM Lookup in OIC to map existing CPQ status codes with the Order Management status codes.

Perform the following steps to modify Lookups defined in OIC

1. Log in to the OIC application.
2. Select **Projects** in the left side navigation panel, and then click on project **Oracle CPQ Fusion Bridge**
3. In the Lookups Panel, click on 3 dots  and Edit option for the corresponding lookup.



4. Click on the Add Row icon (+) at the bottom of the mapping table to add a new mapping.




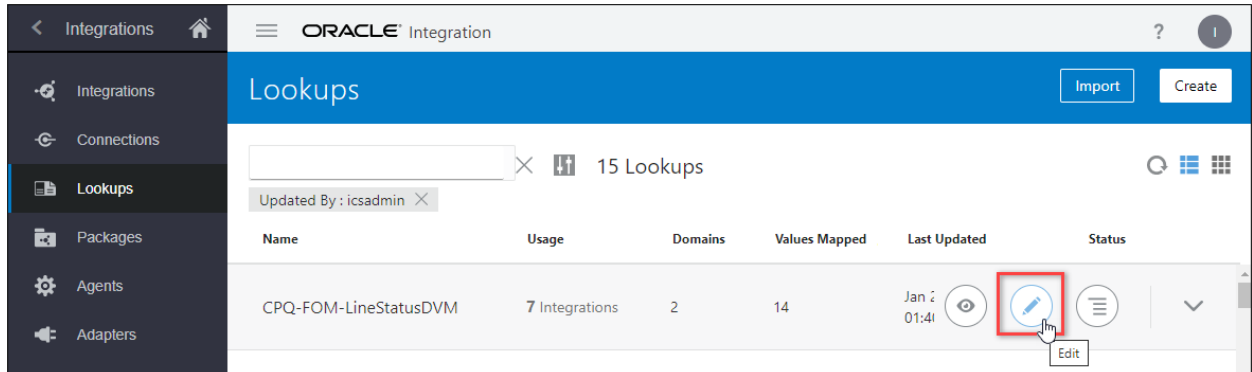
5. Add the FOM Status code in the 'FOM-StatusCode' column and CPQ status code in the 'CPQ-StatusCode' column.
6. Click **Save** after all modification are complete.
7. A dialog window opens listing the integrations using this lookup. Record those integrations, save the window, and then reactivate these integrations to apply the changes.

Update 23B, 23C, or 23D:

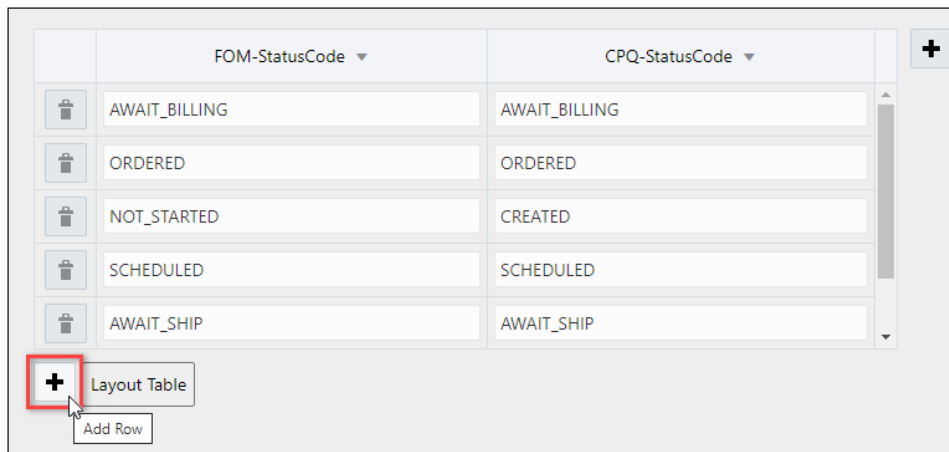
Administrators can use the CPQ-FOM-LineStatusDVM Lookup in OIC to map existing CPQ status codes with the Order Management status codes.

Perform the following steps to modify Lookups defined in OIC

1. Log in to the OIC application.
2. Select **Integrations** in the left side navigation panel, and then select **Lookups**.
3. If required, search for the applicable Lookup.
4. Click the Edit icon  for the corresponding lookup.



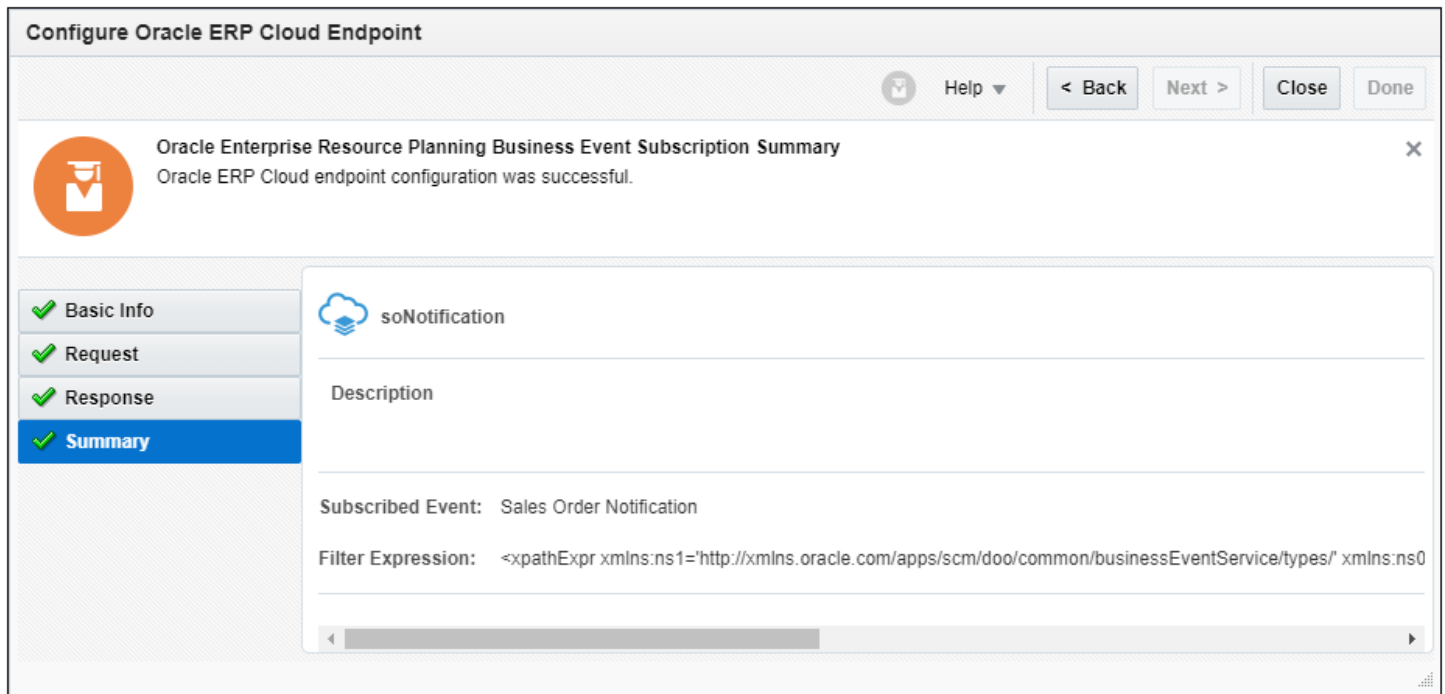
5. Click on the Add Row icon (+) at the bottom of the mapping table to add a new mapping.



6. Add the FOM Status code in the 'FOM-StatusCode' column and CPQ status code in the 'CPQ-StatusCode' column.
7. Click **Save** after all modification are complete.
8. A dialog window opens listing the integration(s) using this lookup. Record those integrations and then reactivate these integrations to apply the changes.

Appendix E1: OIC Integration – UpdateSOStatusFromFOM/ FOM CPQ SalesOrderStatusSync

The integrations UpdateSOStatusFromFOM (Update 23B, 23C, or 23D) and FOM CPQ SalesOrderStatusSync (Update 24A or later) uses Oracle ERP Cloud adapter to subscribe to order management **Sales Order Notification** business event.



There is a filter expression defined to subscribe to the events for only those orders that were originated from CPQ by specifying the Source System 'ORA_BM_CPQ' in the filter expression.

In Order Management, Sales Order Notification business event is raised for an order each time any of the following condition occurs:

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

Whenever the above events are raised, this integration is invoked and the quote header status and line status are updated in Oracle CPQ.

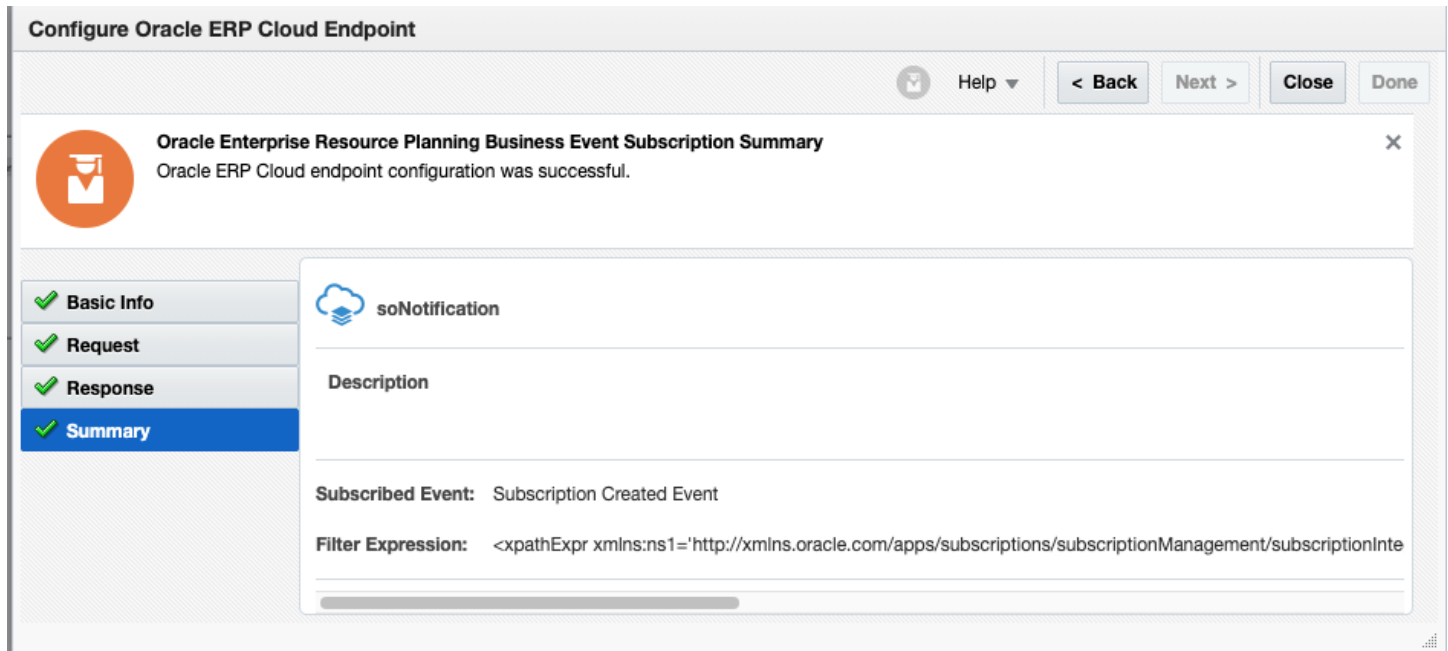
To enable business events in Order Management for different Fulfillment Status of order orchestration process, refer to the **Order Management Setup** section.

Notes:

- Fulfillment Status for which business events are enabled can only be synched back to Oracle CPQ.
- The CPQ Integration User should have Read/Write permissions on the status attributes which are being updated through this integration.

Appendix E2: OIC Integration – UpdateSubStatusFromOSS/ OSS CPQ AssetUpsertFlow

The integrations UpdateSubStatusFromOSS (Update 23B, 23C, or 23D) and OSS CPQ AssetUpsertFlow (Update 24A or later) uses Oracle ERP Cloud adapter to subscribe to subscription management **Subscription Created Event** business event.



Configure Oracle ERP Cloud Endpoint

Oracle Enterprise Resource Planning Business Event Subscription Summary
Oracle ERP Cloud endpoint configuration was successful.

soNotification

Description

Subscribed Event: Subscription Created Event

Filter Expression: <xpathExpr xmlns:ns1='http://xmlns.oracle.com/apps/subscriptions/subscriptionManagement/subscriptionInte

There is a filter expression defined to subscribe to the events for only those orders that were originated from Oracle CPQ by specifying the Source System 'ORA_BM_CPQ' in the filter expression.

In Subscription Management, Subscription Created Event business event is raised each time when a subscription created event occurs.

Whenever the above events are raised, this integration is invoked and quote line status is updated to ACTIVATED in Oracle CPQ. The Update Asset action REST API is invoked to create an Asset in CPQ Asset Repository and update the CPQ quote line Fulfillment Status.

Notes:


- Fulfillment Status for which business events are enabled can only be synched back to Oracle CPQ.
- Status mentioned in CPQ attribute 'Trigger Update Asset for Line Status' should have the business event enabled for corresponding Fulfillment Status in Order management. Without this Asset cannot be created in Oracle CPQ.
- The Oracle CPQ Integration User should have Read/Write permissions on the status attributes which are being updated through this integration.

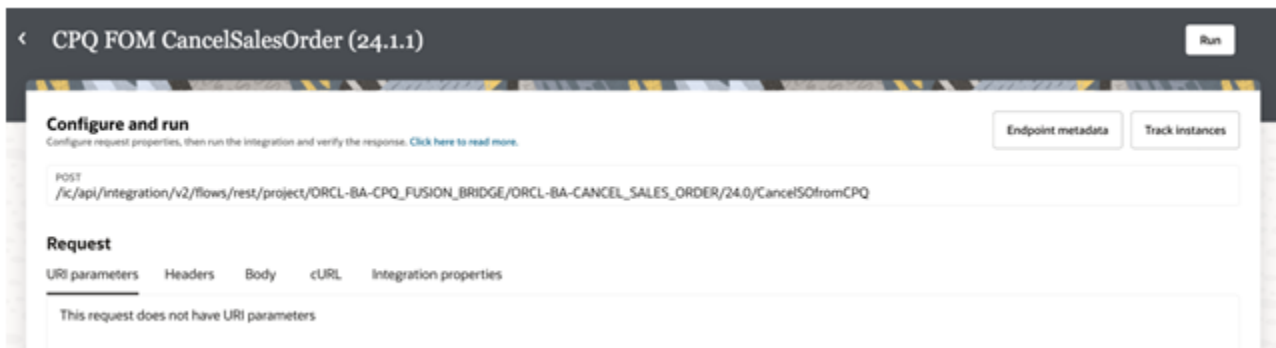
Appendix F: Retrieve OIC Integration Endpoint URL

Update 24A or later:

Integration endpoints are required for defining the Generic Integration in the Integration Center for the cancel order and get order status flows.


Perform the following steps to retrieve an integration endpoint.

1. Log in to the OIC application.
2. Select Projects in the left side navigation panel, and then Click Oracle CPQ Fusion Bridge Project.
3. **CPQ FOM CancelSalesOrder** integration:
 - a. Select **CPQ FOM CancelSalesOrder** Integration, click on 3 dots  and then click on the **Run** option. Configuration and Run UI displays.



- b. Click on the **Endpoint metadata** button. A page displays with the Endpoint URL in the Metadata tab section. Record the Endpoint URL.
`https://<OIC hostname>/ic/api/integration/v2/flows/rest/project/ORCL-BA-CPQ_FUSION_BRIDGE/ORCL-BA-CANCEL_SALES_ORDER/24.0/CancelSOfromCPQ`
Note: The hostname is the OIC environment name.
4. **CPQ FOM CreateSalesOrder** integration:
 - a. Select **CPQ FOM CreateSalesOrder** Integration, click on 3 dots  and then click on the **Run Details** option. Run Details UI displays.
 - b. Record the Metadata URL.
`https://<OIC hostname>/ic/ws/integration/v2/flows/cpq/project/ORCL-BA-CPQ_FUSION_BRIDGE/ORCL-BA-CREATE_SALES_ORDER/24.0`
Note: The hostname is the OIC environment name.
 5. **OSS CPQ AssetUpsertFlow** integration:
 - a. Select **OSS CPQ AssetUpsertFlow** Integration, click on 3 dots  and then click on the **Run Details** option. Run Details UI displays.
 - b. Record the Metadata URL.
`https://<OIC hostname>/ic/ws/integration/v2/flows/erp/project/ORCL-BA-CPQ_FUSION_BRIDGE/ORCL-BA-ASSET_UPSERT_FLOW/24.0`
Note: The hostname is the OIC environment name.

6. **FOM CPQ SalesOrderStatusSync** integration:


a. Select **FOM CPQ SalesOrderStatusSync** Integration, click on 3 dots  and then click on the **Run Details** option. Run Details UI displays.

b. Record the Metadata URL.

`https://<OIC hostname>/ic/ws/integration/v2/flows/erp/project/ORCL-BA-CPQ_FUSION_BRIDGE/ORCL-BA-SALES_ORDER_STATUS_SYNC/24.0`

Note: The hostname is the OIC environment name.

7. **FOM CPQ GetSalesOrderStatus** integration:

a. Select **FOM CPQ GetSalesOrderStatus** Integration, click on 3 dots  and then click on the **Run** option. Configure and Run UI displays.

b. Click on the EndPoint metadata button. A page displays with the Endpoint URL in the Metadata tab section. Record the Metadata URL.

`https://<OIC hostname>/ic/api/integration/v2/flows/rest/project/ORCL-BA-CPQ_FUSION_BRIDGE/ORCL-BA-GET_SALES_ORDER_STATUS/24.0/GetSOStatusfromFOM`

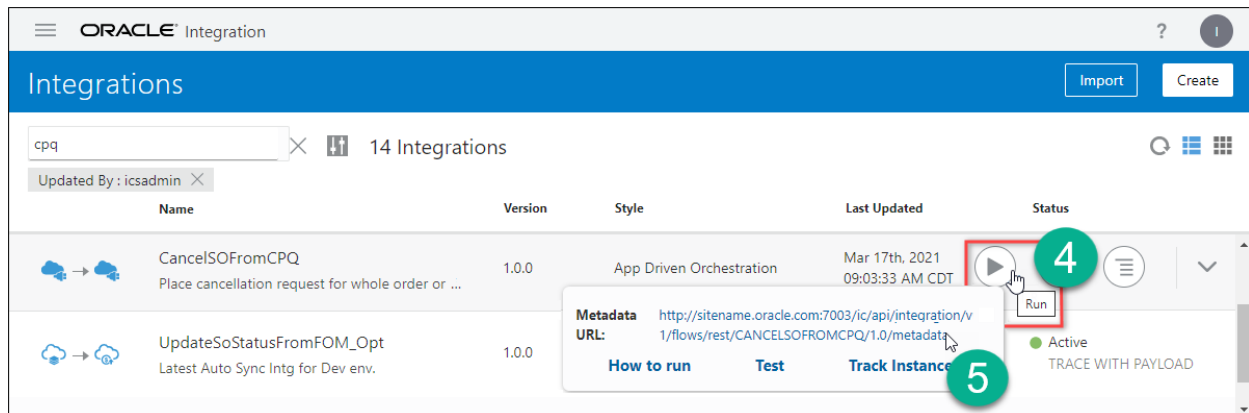
Note: The hostname is the OIC environment name.

Update 23B, 23C, or 23D:

Integration endpoints are required for defining the Generic Integration in the Integration Center for the cancel order and get order status flows.

Perform the following steps to retrieve an integration endpoint.

1. Log in to the OIC application.
2. Select **Integrations** in the left side navigation panel, and then select **Integrations**.
3. If required, search icon to find your integration (e.g. CancelSOFromCPQ).
4. Click the Run icon for the applicable integration. A pop-up window displays with the Metadata URL.



5. Click on the Metadata URL link. A page will display with the Endpoint URL. Record the Endpoint URL.

Endpoint Description

Endpoint URL

<http://sitename.oracle.com:7003/ic/api/integration/v1/flows/rest/CANCELISOFROMCPQ/1.0/CancelISOfromCPQ>

Swagger

<http://sitename.oracle.com:7003/ic/api/integration/v1/flows/rest/CANCELISOFROMCPQ/1.0/metadata/swagger>

Open API

<http://sitename.oracle.com:7003/ic/api/integration/v1/flows/rest/CANCELISOFROMCPQ/1.0/metadata/openapi>

Appendix G: Limitations and Troubleshooting

Appendix G1: Limitations

OIC integration using the Rest Adapter has a 10MB limit in the size of a message that can be received by the adapter. If the request payload for create order exceeds 10MB in OIC, you will see the following error message.

```
"Content received of length <payload size> exceeds the maximum allowed threshold of 10485760 bytes"
```

Appendix G2: Troubleshooting

Resolve Issues with Create Order Action

If installing the CPQ Order Management package results in issues with the Create Order action, a workaround is available.

Perform the following steps to resolve issues with the Create Order action.

1. Open Oracle CPQ.
2. Select **Integration Center** under **Integration Platform**. The Integration Center opens.
3. Select the **ICS integration** from the left pane.
4. Unmark the **Enable Integration** check box to disable the integration.
5. Click **Save**.
6. Re-enable the ICS integration.
7. Click **Save**.

Assets Not Created in CPQ Assets Repository

- Check whether the business events are enabled for the Fulfilment status corresponding to status values mentioned in the CPQ attribute 'Trigger Update Asset for Line Status'.
- Check integration UpdateSOSStatusFromFOM instance in OIC for errors. If there are not any you do not find any UpdateSOSStatusFromFOM instance errors, there is an issue with the event subscription setup in OIC.

Blank Line Status on click of 'Get SO Status from FOM' button

If a line status becomes blank after clicking **Get SO Status from FOM**, it means that either the order management line status lookup code is not populated in the `status_l` attribute or the status mapping doesn't exist in OIC lookup **CPQ-FOM-LineStatusDVM** for the Order Management line status of that line.

To resolve this, either add a value in the `status_l` attribute or add mapping to the **CPQ-FOM-LineStatusDVM** lookup in OIC for that status.

CONNECT WITH US

Call +1.800.ORACLE1 or visit oracle.com.

Outside North America, find your local office at oracle.com/contact.



blogs.oracle.com



facebook.com/oracle



twitter.com/oracle

Copyright © 1994, 2024 Oracle and/or its affiliates.

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

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

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

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

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

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

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

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

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.

