APPROVED
RELEASE
READINESS
06161977
PLAN FOR SUCCESS

PRICING
QUOTING
CONFIGURATION

Oracle CPQ Cloud

What’s New in 2016 R2

December 2016

ORACLE

TABLE OF CONTENTS

REVISION HISTORY

This document will continue to evolve as existing sections change and new information is added. All updates are logged below, with the most recent updates at the top.

Date	What's Changed	Notes
14 DEC 2016		Initial Document Creation
22 MAR 2017	• ~ behavior note added to Pre-Upgrade Considerations • Minor revisions to Mobile UI Enhancements	

OVERVIEW

This guide outlines information about new or improved functionality in Oracle Configure, Price, and Quote (CPQ) Cloud 2016 Release 2 (2016 R2). Each section includes a brief description of the feature, the steps you need to take to enable or begin using the feature, any tips or considerations to keep in mind, and the resources available to help you.

GIVE US FEEDBACK

We welcome your comments and suggestions to help us improve this document. Send your feedback to CPQ_Cloud_documentation_us_grp@oracle.com.

RELEASE FEATURE SUMMARY

Some of the new CPQ Cloud 2016 Release 2 features are automatically available to users after the upgrade and some require action from the company administrator or Oracle.

The following table offers a quick view of the actions required to enable each of the features.

Feature	**Action Required to Enable Feature**		
	Automatically Available	**Administrator Action Required**	**Oracle Service Request Required**
Modern Selling Experience			
Mobile UI Enhancements	✓		
Enterprise Platform			
REST API Enhancements	✓	✓	
Transaction REST APIs	✓	✓	
External Configurator REST API	✓	✓	
SOAP API Enhancements	✓	✓	
Performance Enhancements	✓		
Integration Suite			
Commerce Cloud Integration		✓	
External Configurator Integration		✓	
Other Enhancements	✓		

MODERN SELLING EXPERIENCE

Leverage the rich interface and interactive capabilities available in CPQ Cloud to provide a smarter selling experience.

The following enhancements are available in CPQ Cloud 2016 R2:

- Mobile UI Enhancements

MOBILE UI ENHANCEMENTS

CPQ Cloud 2016 R2 introduces Mobile UI enhancements that allow customers to use CPQ Cloud on a mobile phone and provides improved usability on both phones and tablets. In previous releases, the CPQ Cloud Mobile UI was optimized for only landscape orientation on tablet devices. With this release, screen use is dynamically optimized for both landscape and portrait orientations and introduces an improved, modernized user experience.

Reinforcing Oracle's commitment to supporting integration with Salesforce, these Mobile UI enhancements also apply to the integration of the CPQ Cloud Mobile UI with the Salesforce1 Mobile UI. Stylistic improvements on each CPQ Cloud mobile page provide a seamless user experience directly aligning the CPQ Mobile UI with the Salesforce1 Mobile UI.

The 2016 R2 Mobile UI Enhancements are described below:

- Use CPQ Cloud on a mobile phone or tablet device
- View CPQ Cloud in portrait or landscape orientation
- Show columns in the Transaction Line Grid

USE CPQ CLOUD ON A MOBILE PHONE OR TABLET DEVICE

CPQ Cloud 2016 R2 extends the modern selling capabilities of CPQ Cloud through support of CPQ Cloud on mobile phones. Customers can now use the CPQ Cloud Mobile UI on either a mobile phone or a tablet device.

To support this enhancement, several stylistic changes are included on CPQ Cloud mobile pages to provide a more modern and user friendly mobile experience.

- The color scheme of CPQ mobile pages is more directly aligned with other Oracle products.
- The removal of margins from CPQ mobile pages provides more space when using CPQ Cloud on a mobile device.
- The header and footer on CPQ Cloud mobile pages are re-styled. The header includes a navigation menu and tabs, and a Parts Search button directly below the header. The footer contains a list of available actions.

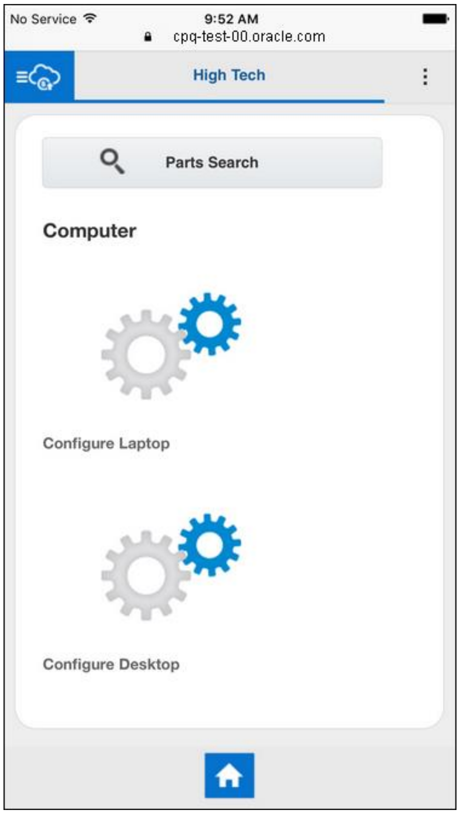
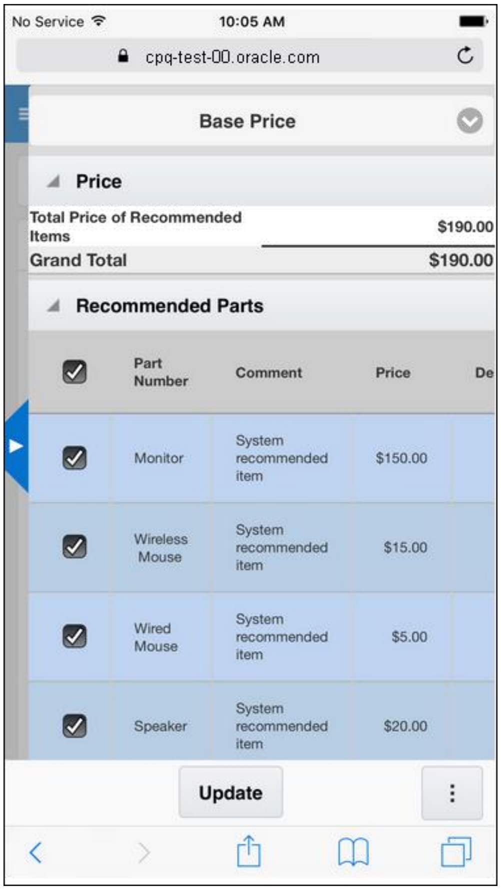

No Service
9:52 AM
cpq-test-00.oracle.com
High Tech
Parts Search
Computer
Configure Laptop
Configure Desktop

Figure 1: CPQ Cloud on a Mobile Phone

VIEW CPQ CLOUD IN PORTRAIT OR LANDSCAPE ORIENTATION

CPQ Cloud 2016 R2 provides the ability to view the CPQ Cloud Mobile UI in portrait orientation. Since landscape orientation was supported in prior releases, customers now have the option of viewing the CPQ Cloud mobile UI in either portrait or landscape orientation.



Base Price

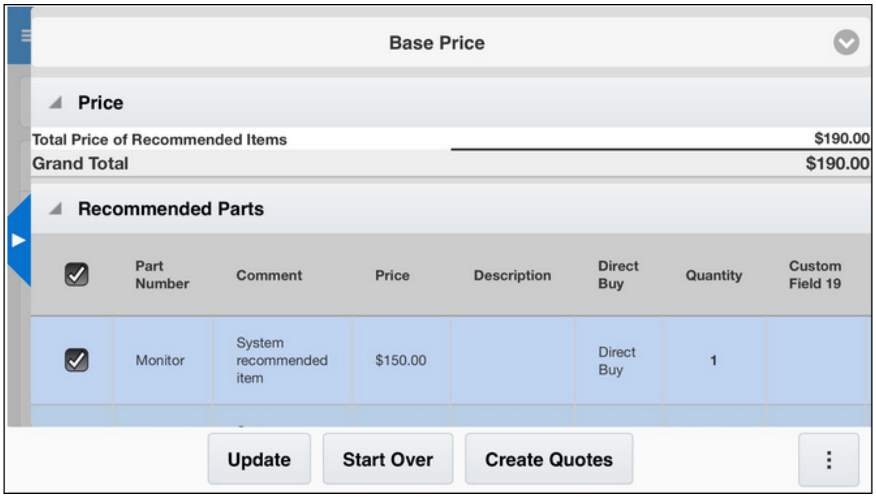
Price

Total Price of Recommended Items	$190.00
Grand Total	$190.00

Recommended Parts

☑	Part Number	Comment	Price	De
☑	Monitor	System recommended item	$150.00	
☑	Wireless Mouse	System recommended item	$15.00	
☑	Wired Mouse	System recommended item	$5.00	
☑	Speaker	System recommended item	$20.00	

Figure 2: Recommended Items Table in Portrait Orientation



Base Price

Price

Total Price of Recommended Items	$190.00
Grand Total	$190.00

Recommended Parts

☑	Part Number	Comment	Price	Description	Direct Buy	Quantity	Custom Field 19
☑	Monitor	System recommended item	$150.00		Direct Buy	1	

Figure 3: Recommended Items Table in Landscape Orientation

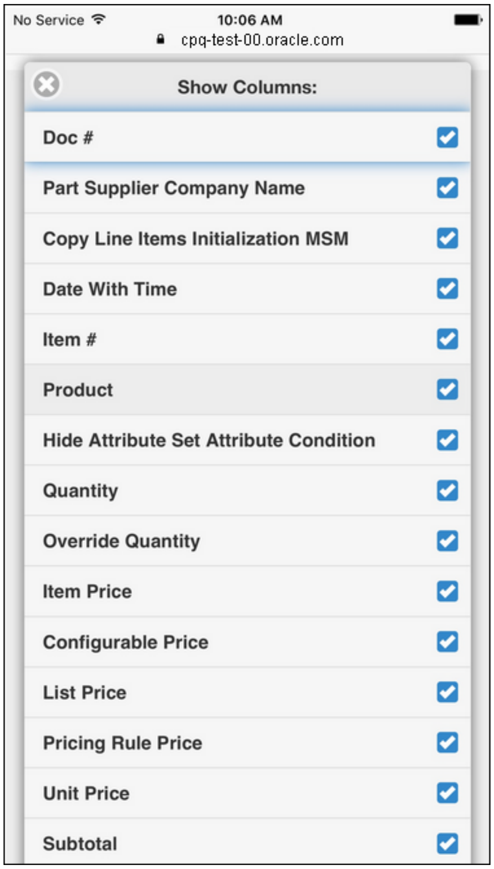
NOTE: The number of buttons that display is based on the width of the screen in the current orientation. Use the ⋮ button to access the action buttons not shown in the footer.

SHOW COLUMNS IN THE TRANSACTION LINE GRID

A menu is now available in the upper-right corner of the Transaction Line Grid. This menu allows customers to show specific columns in the Transaction Line Grid.

To implement this enhancement, the attribute strip available in the previous mobile Commerce layout was removed.

NOTE: When the quote page is re-loaded from the server, the settings are lost. Customers must return to the menu in the upper-right corner of the Transaction Line Grid to re-apply their settings.



No Service 10:06 AM
cpq-test-00.oracle.com

Show Columns:

- Doc # ☑
- Part Supplier Company Name ☑
- Copy Line Items Initialization MSM ☑
- Date With Time ☑
- Item # ☑
- Product ☑
- Hide Attribute Set Attribute Condition ☑
- Quantity ☑
- Override Quantity ☑
- Item Price ☑
- Configurable Price ☑
- List Price ☑
- Pricing Rule Price ☑
- Unit Price ☑
- Subtotal ☑

Figure 4: Show Columns in the Transaction Line Grid

STEPS TO ENABLE

Mobile UI enhancements are automatically available in the CPQ Cloud 2016 R2 mobile UI.

TIPS AND CONSIDERATIONS

Consider the following tips when using the 2016 R2 Mobile UI enhancements:

- Customers must use portrait orientation when using CPQ Cloud in the Salesforce1 application on a mobile phone. Salesforce1 will not display in landscape orientation.
- With the exception of the ability to view the Mobile UI in landscape orientation, all of the other Mobile UI enhancements described in this section work with the Salesforce1 integration introduced in CPQ Cloud 2016 R1.
- In order for mobile browsers to work with the integration of the CPQ Cloud Mobile UI with Salesforce1, the browsers must be set to always allow cookies.
- Due to a known issue with Safari, viewing a web page in landscape orientation on an iPhone Plus device causes buttons and menus to stop working. To resolve this issue, navigate to **Settings** > **Safari** on your iPhone Plus device and disable the **Show Tab Bar** setting.
- Any action that attempts to open a new browser window from CPQ Cloud Mobile UI with Salesforce1 will not work unless pop-up blockers are turned off. This issue effects actions such as the eSignature Details page, Show History, Browse for Accounts, Configuration Pipeline Viewer, Show File Attachment, and Online Help. This issue does not affect the Print action or operations when using Saleforce1 with Safari.
- Users can no longer swipe from left to right to navigate between the CPQ Cloud Home page and pending configuration pages. In addition, users can no longer swipe to show other tabs on the Home, configuration, or commerce pages. Users should access the existing overflow menu to view all tabs or to navigate between pages.

KEY RESOURCES

Refer to the following resources for additional information:

- CPQ Cloud Administration Online Help

ENTERPRISE PLATFORM

Oracle is continually upgrading the functionality of CPQ Cloud to allow customers to build an open and flexible platform that creates value and drives results for companies.

The following Enterprise Platform features and enhancements are available in CPQ Cloud 2016 R2:

- Enhancements for REST APIs
- Transaction REST APIs
- External Configurator REST API
- SOAP API Enhancements
- Performance Enhancements

ENHANCEMENTS FOR REST APIS

In 2016 R2, CPQ provides several REST API enhancements. These updates include additional request path parameters, request body parameters, and response data.

- Request Path Enhancement – Provides the Exclude Field Type query.
- Request Body Enhancements – Provide additional request functionality for transaction actions.
- Response Enhancements - Extend and improve the data returned for transaction operations.

REQUEST PATH ENHANCEMENT

In 2016 R2, CPQ Cloud provides the Exclude Field Types request path query to limit returned filed types. Refer to CPQ Cloud Online Help - REST APIs for additional request path query parameters.

EXCLUDE FIELD TYPES

When making REST API calls for CPQ transaction data, the system returns transaction data including all attribute types. This can create performance issues when the transaction is large and includes attributes containing HTML contents. Valid field types include boolean, currency, date, fileAttachment, float, history, integer, language, menu, readOnlyTextOrHtml, richText, secureText, summation, text, textArea, composite, customerId, sequenceNumber, and system.

To limit returned fields, append the "?excludeFieldType=<*field type(s)*>" query parameter to the end of the request path.

REQUEST BODY ENHANCEMENTS

In 2016 R2, CPQ Cloud provides enhancements that allow request body inputs for selection, criteria queries, and user cache interaction. These requests are only available for REST APIs with POST methods.

NOTE: Request body inputs must be written in JSON and adhere to standard JSON syntax.

LINE SELECTION PARAMETER

This enhancement supports using the REST call request body to select transaction lines. This service can be used for transaction and line level actions that support line selection.

Line Selection Sample

```
{"selections": ["transactionLine/2", "transactionLine/3"]}
```

CRITERIA PARAMETER QUERIES

An additional "criteria" object is now available when invoking a REST call on a resource action. Administrators can provide criteria parameters in the request body to retrieve specific content and limit the size of the response.

CRITERIA QUERY PARAMETERS

Parameter	Description / Example	
`Actions`	An array of action names, whose state is requested in the state object response.	
	`"actions": ["save", "print"]`	
`ChildDefs`	An array of child query definitions that specifies query requests for child resources.	
childDef Parameters	`name`	The alias name of the child resource for this query
	`recursive`	This Boolean parameter specifies whether the child resource is recursively expanded.
	`queryDef`	An array of query definitions for the child resource
`exclude FieldTypes`	This parameter allows exclusion of specified attribute types from the response. **Note:** "excludeFieldTypes" and "fields" parameters cannot be used together in the same request body.	
	`"excludefieldTypes": ["fieldtype1", "fieldtype2"]`	
`Expand`	An array of child resource names to be expanded and returned with the parent in the same request.	
	`"expand": ["transactionLines", "accounts"]`	
`expandedDomains`	An array of field names that returns expanded domains in the state object response. The field's domain object will be expanded to include an "availableElements" array.	

Parameter	Description / Example
	`"expandedDomains": ["menuAttr1", "menuAttr2"]`
`fields`	An array of field names to be returned. The "fields" array values determine the "attributes" in the response's state object.
	`"fields": ["field1", "field2"]`
`groups`	An array of group names, whose state is requested in the state object response.
	`"groups": ["tab_123", "tab_234"]`
`orderBy`	An array of field names to specify the order of items returned in the response, each field name could optionally be followed by ASC or DESC.
	`"orderBy": ["listPrice: ASC", "totalPrice: DESC"]`
`Q`	This string declares a query specification expression in MongoDB format.
	`"q": "{'_part_number':{$eq:'part100'}}"`
`state`	This Boolean parameter specifies whether to include the state property. By default, this value is set to false and the state property is not included.
	`"state": false`
`totalResults`	This Boolean parameter specifies whether to calculate the totalResults property. By default, the value is set to false indicating that totalResults is not calculated.
	`"totalResults": false`

Request Body Sample

```
"criteria": {
   "totalResults": {true,false},
    "q": "{'_part_number':{$eq:'part100'}}"}},
   "state": false,
   "orderBy": ["listPrice: ASC", "totalPrice: DESC"],
   "expand": ["transactionLines", "accounts"],
   "fields": ["field1", "field2"],
   "actions": ["save", "print"],
   "groups": ["tab_123", "tab_234"],
   "expandedDomains": ["menuAttr1", "menuAttr2"],
   "childDefs": [ {
      "name": "transactionLines",
      "recursive": false,
      "queryDef": {
         "state": true,
         "fields": ["lineField1", "lineField2]
      }
    }
  ]
}
```

STATE OBJECT

The State Object queries allow filtering of state data in responses based on the query object provided in request body for Commerce Action APIs. Refer State Object Support for a detailed of returned objects.

NOTE: Empty array fields are not allowed for attributes. When an empty array is used in the criteria object for actions, groups, the corresponding state object will be omitted from the response. When an empty array is used in the criteria object for expanded domains, all attributes' domains returned in the state object will omit the "available" elements property.

- **Attributes**: Returns the state data for specified attributes. Use the attribute variable names to specify arrays in the criteria "fields" property. State data for all attributes is returned when criteria "fields" are not specified in the request body.
- **Actions**: Returns the state data for specified actions. Use the action variable names to specify arrays in the criteria "actions" property. State data for all actions is returned when criteria "actions" are not specified in the request body.
- **Groups**: Returns the state data for specified groups. Use group expressions to specify arrays in the criteria "groups" property. State data for all groups is returned when criteria "groups" are not specified in the request body. The expression for a group is the same as is returned in a response; for example, to refer to a tab with an id of 18253632, use the expression "tab_18253632".
- **Expanded Domains**: Returns the available elements data for specified attributes. Use the attribute variable names to specify arrays in the criteria "expandedDomains" property. State data for all elements is returned when criteria "expandedDomains" are not specified in the request body.

CACHE INTERACTION

This enhancement supports user cache interaction, which is used for interim state storage for AJAX rule execution. When invoking any action the user can pass in optional parameter called "cacheInstanceId". If the cache instance ID is not provided, previous changes to the cached object are ignored and the requested action will process the last saved instance of the transaction.

- When "cacheInstanceId": "-1" is provided in an action request body, then system will perform the requested action and load the complete transaction into the user's cache. The "cacheInstanceId" value is returned in the response.
- In subsequent requests the "cacheInstanceId": "value" parameter can be used to work with the cached data.

USE CASES EXAMPLES

1. Open transaction 18097326, and load the document into user cache.

   Request Path Sample

   ```
   https://sitename.oracle.com/rest/v3/commerceDocumentsOraclecpqoTransaction/1809
   7326/actions/_open_transaction
   ```

   Request Body Sample

   ```
   {"cacheInstanceId": "-1"}
   ```

   Sample "cacheInstanceId" Response

   ```
   "cacheInstanceId":
   "E9JuyC9QCgTDS9edghTNxRGOVUm34EMR7RJt9alFbkGTN0X5byTjHGRu1CxrSl1p"
   ```

2. Retrieve cache instance and remove transaction line 8.

   Request Path Sample

   ```
   https://sitename.oracle.com/rest/v3/commerceDocumentsOraclecpqoTransaction/18097
   326/actions/_remove_transactionLine
   ```

   Request Body Sample

   ```
   {
      "cacheInstanceId":
   "E9JuyC9QCgTDS9edghTNxRGOVUm34EMR7RJt9alFbkGTN0X5byTjHGRu1CxrSl1p",
      "selections": ["transactionLine/8"]
   }
   ```

RESPONSE ENHANCEMENTS

In 2016 R2, CPQ Cloud provides several enhancements to return information necessary for integration with external user interfaces.

DESTINATION URL

This enhancement returns the destination URL when a CPQ Cloud action has a destination tab with a configured destination. When available, the destination URL is returned in the JSON.

NOTE: The destination URL value may change from release to release. Consequently, logic should not rely on a specific fragment of the URL.

CURRENCY TYPE ATTRIBUTES

This enhancement returns currency type field as composite values. The composite value contains currency value and currency type.

Currency Type Sample

```
"transactionLineTotal": {"value": 43.35, "currency": "USD"}
```

SINGLE-SELECT AND MULTI-SELECT MENU ATTRIBUTES

This enhancement returns Single-Select and Multi-Select Menus as composite values. For Single-Select Menus, the composite value contains "value" and "displayValue" sub-fields. For Multi-Select Menus, the composite value contains a composite with an array of item elements each containing "value" and "displayValue" sub-fields.

Single-Select Menu Sample

```
"processManagerSSM": {"displayValue": "Value 1 Label", "value": "value1"}
```

Multi-Select Menu Sample

```
"processManagerMSM": {
   "items": [
      {"displayValue": "Value 2 Label", "value": "value2"},
      {"displayValue": "Value 3 Label", "value": "value3"}
   ]
}
```

STATE OBJECT SUPPORT

Transaction data services can now return state objects, if requested in the request body. For example, {"criteria": "state": "true"}. For more information, refer to Data Service Actions and Criteria Parameter Queries sections. The following table lists the state objects that are returned for transactions, transaction line lists, transaction lines, actions, attributes, domains, and groups.

State Object	Transaction	Transaction Line List	Transaction Line	Action	Attribute	Domain	Group
actions	X		X				
attributes	X		X				
availableElements						X	
constraintList						X	
constraintType						X	
defaultRowState		X					
deletable			X				

State Object	Transaction	Transaction Line List	Transaction Line	Action	Attribute	Domain	Group
domain					X		
enabled				X			
groups	X		X				
hasConstraint Violation					X		
hasError	X		X		X		X
insertable		X					
messages	X				X		
queryable		X					
required					X		
sortable		X					
updatable			X		X		
visible				X	X		X

STEPS TO ENABLE

The REST API Enhancements are automatically available on 2016 R2 sites.

KEY RESOURCES

Refer to the CPQ Cloud Administration Online Help for additional information.

TRANSACTION REST APIS

In 2016 R2, CPQ Cloud enhances the Transaction REST APIs to expose additional transactional data, metadata, layout, rules, and actions to enable seamless integration with third party user interfaces, such as the ADF UI and the Salesforce Lightning UI. The new REST APIs are exposed using v3 RESTful services.

This section is divided into the following sections:

- Data Service Actions – These APIs provide access to interface with transaction operations.
- Layout Service – This service returns the desktop layout definition for a Commerce process.
- Metadata Services – These services return definitions for main documents, sub documents, actions, attributes, menu items, and translations.

DATA SERVICE ACTIONS

Data service actions provide access to interface with transaction operations. In 2016 R2, request body parameters can be used to select items, modify items, retrieve specific content, and interact with user cache sessions.

OPEN TRANSACTION

Open Transaction		
Description	This action returns transaction data for a specific Transaction in the Commerce Process including any relevant attributes, actions, or rules.	
Request Path	`/rest/v3/commerceDocuments{processVarName}{MainDocVarName}/(transactionId}/actions/{actionVarName}`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`MainDocVarName`	The variable name of the main document
	`transactionId`	The transaction ID
	`actionVarName`	The variable name of the action to open a transaction
HTTP Method(s)	POST	
Request Body Parameters	Optional, additional parameters can be provided to limit the results returned	
Success Response	The JSON data for the requested transaction, or a subset of transaction items, depending on the input request	

Request Path Sample

Open transaction 18097326

```
https://sitename.oracle.com/rest/v3/commerceDocumentsOraclecpqoTransaction/1809
7326/actions/_open_transaction
```

OPEN TRANSACTION LINE

Open Transaction Line		
Description	This action opens selected transaction line and retrieves transaction line details including any relevant attributes, actions, or rules.	
Request Path	`/rest/v3/commerceDocuments{processVarName}{MainDocVarName}/{transactionId}/actions/{actionVarName}`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`MainDocVarName`	The variable name of the main document
	`transactionId`	The Commerce Transaction ID
	`actionVarName`	The variable name of the action to open a transaction line
HTTP Method(s)	POST	
Request Body Parameters	`selections`	Used to specify the transaction line
Success Response	The JSON data for the parent transaction, or a subset of transaction items, depending on the input request	

Request Path Sample

```
https://sitename.oracle.com/rest/v3/commerceDocumentsOraclecpqoTransaction/18094477/actions/_open_transactionLine
```

Request Body Sample

```
{"selections": ["transactionLine/3"]}
```

CREATE TRANSACTION

Create Transaction		
Description	This action creates a new Transaction in the Commerce Process.	
Request Path	`/rest/v3/commerceDocuments{ProcessVarName}{MainDocVarName}/actions/_new_transaction`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`MainDocVarName`	The variable name of the main document
HTTP Method(s)	POST	
Request Body Parameters	`Criteria`	Optional, use to request additional details, such as the 'state' object, refer to Criteria Parameters for more information
	`cacheInstanceId`	Optional, use a "-1" value to load the complete transaction into the user's cache
Success Response	The JSON data of the new Transaction	

Request Path Sample

```
https://sitename.oracle.com/rest/v3/commerceDocumentsOraclecpqoTransaction/actions/_copy_transaction
```

DESTINATION URL

This enhancement returns the destination URL when a CPQ Cloud action has a destination tab with a configured destination. When available, the destination URL is returned in the JSON.

NOTE: The destination URL value may change from release to release. Consequently, logic should not rely on a specific fragment of the URL.

COPY TRANSACTION

Copy Transaction		
Description	This action duplicates an existing transaction. **NOTE:** When multiple Copy Sequence Lists exist for a user type, the first accessible sequence is processed. Users are not given the option to choose copy options.	
Request Path	`/rest/v3/commerceDocuments{processVarName}{MainDocVarName}/{transactionId}/actions/_copy_transaction`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`MainDocVarName`	The variable name of the main document
	`transactionId`	The Commerce Transaction ID
HTTP Method(s)	POST	
Request Body Parameters	Optional, additional parameters can be provided to limit the results returned	
Success Response	The JSON data for the duplicated transaction, or a subset of transaction items, depending on the input request	

Request Path Sample

```
https://sitename.oracle.com/rest/v3/commerceDocumentsOraclecpqoTransaction/18103143/actions/_copy_transaction
```

COPY TRANSACTION LINE

This action is used to duplicate transaction lines. Request body "selections" and "numberOfCopies" parameters are used to provide specifications.

NUMBER OF COPIES PARAMETER

The "numberOfCopies" parameter is an array to specify the number of copies for each line listed in the "selections" parameter.

- The number of copies for each line is specified in an array in the same sequence as the "selections" array.
- The default value is '1', if the number of copies value is not provided.
- The "numberOfCopies" array size cannot be greater than the "selections" array size.

Copy Transaction Line		
Description	This action duplicates transaction lines within an existing transaction.	
Request Path	`/rest/v3/commerceDocuments{processVarName}{MainDocVarName}/{transactionId}/actions/{actionVarname}`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`MainDocVarName`	The variable name of the main document
	`transactionId`	The Commerce Transaction ID
	`actionVarName`	The variable name of the action to copy transaction line
HTTP Method(s)	POST	
Request Body Parameters	`numberOfCopies`	An array to specify the number of copies for each line listed in the "selections".
	`Selections`	Used to specify the transaction line(s)
	Additional parameters can be provided to limit the results returned	
Success Response	The JSON data for the parent transaction including the copied transaction lines, or a subset of transaction items, depending on the input request transaction	

Request Path Sample

```
https://sitename.oracle.com/rest/v3/commerceDocumentsOraclecpqoTransaction/1809
7326/actions/_copyLineItems_t
```

Request Body Samples

1. Copy transaction lines 5 and 3, copy each item once

```
{
   "numberOfCopies": [1,1],
   "selections": ["transactionLine/5", "transactionLine/3"]
}
```

2. Modify, and then copy transaction lines 5 and 7, copy each item once

```
{
   "documents": {"transactionDescription": "Mar31",
      "transactionLine": {"items": [
            {"_document_number": "5", "_price_quantity": "45"},
            {"_document_number": "7", "_price_quantity": "45"}
      ]}
   },
   "numberOfCopies": [1,1],
   "selections": ["transactionLine/5", "transactionLine/7"]
}
```

REMOVE TRANSACTION LINES

Remove Transaction Lines		
Description	This action deletes the selected transaction line(s) from the transaction.	
Request Path	`/rest/v3/commerceDocuments{processVarName}{MainDocVarName}/{transactionId}/actions/{actionVarName}`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`MainDocVarName`	The variable name of the main document
	`transactionId`	The Commerce Transaction ID
	`actionVarName`	The variable name of the action to remove transaction line(s)
HTTP Method(s)	POST	
Request Body Parameters	`Selections`	Used to specify the transaction line(s)
	Additional parameters can be provided to limit the results returned	
Success Response	The JSON data for the parent transaction without the deleted transaction lines, or a subset of transaction items, depending on the input request	

Request Path Sample

```
https://sitename.oracle.com/rest/v3/commerceDocumentsOraclecpqoTransaction/17994570/actions/_remove_transactionLine
```

Request Body Sample

Use "selections" item in Request Body to select transaction lines 2 and 3

```
{"selections": ["transactionLine/2", "transactionLine/3"]}
```

UPDATE TRANSACTION LINES

Update Transaction Lines		
Description	This action modifies transaction line details and updates the existing transaction.	
Request Path	`/rest/v3/commerceDocuments{processVarName}{MainDocVarName}/{transactionId}/actions/{actionVarName}`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`MainDocVarName`	The variable name of the main document
	`transactionId`	The Commerce Transaction ID
	`actionVarName`	The variable name of the action to update transaction lines
HTTP Method(s)	POST	
Request Body Parameters	Additional parameters can be provided to limit the results returned	
Success Response	JSON data for the transaction, including the updated transaction lines	

Request Path Sample

```
https://sitename.oracle.com/rest/v3/commerceDocumentsOraclecpqoTransaction/17994570/actions/_update_line_items
```

TRANSACTION ACTIONS FOR CONFIGURATION

RECONFIGURE FROM TRANSACTION

Reconfigure from Transaction		
Description	This action provides ability to reconfigure a model transaction line from a transaction.	
Request Path	`/rest/v3/commerceDocuments{ProcessVarName}{MainDocVarName}/{transactionId}/actions/{actionVarName}`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`MainDocVarName`	The variable name of the main document
	`transactionId`	The Commerce Transaction ID
	`actionVarName`	The variable name of the action to reconfigure from a transaction
HTTP Method(s)	POST	
Request Body Parameters	`selections`	Used to specify the transaction line(s)
	Additional parameters can be provided to limit the results returned	
Success Response	The response contains the Destination URL, which can be used to launch the product catalog page to select and configure a product.	

Request Path Sample

```
https://sitename.oracle.com/rest/v3/commerceDocumentsOraclecpqoTransaction/1801
6533/actions/_reconfigure_action
```

Request Body Sample

```
{
   "documents": {"displayHistoryLanguage": "en"},
   "selections": ["transactionLine/2"]
}
```

Response Body Sample

```
{
   "destinationURL": "https://sitename.oracle.com/commerce/buyside
/reconfig_form.jsp?action_id=4356762&doc_number=1&document_id=4356178
&id=18016533&destination_url=&_line_item_list=2",
   "documents": {
      "conditionsOperatorsHelp": null,
      "formulaContains": null,
      "reconfigureFormulasInbound": "0"
   }
}
```

RECONFIGURE FROM TRANSACTION LINE

Reconfigure from Transaction Line		
Description	This action provides ability to reconfigure a transaction line from a line level action.	
Request Path	`/rest/v3/commerceDocuments{ProcessVarName}{MainDocVarName}/{transactionId}/{subDocVarName}/{subDocNumber}/actions/{actionVarName}`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`MainDocVarName`	The variable name of the main document
	`transactionId`	The Commerce Transaction ID
	`subDocVarName`	The variable name of the sub document
	`subDocNumber`	The sub document number
	`actionVarName`	The variable name of the action to reconfigure from a transaction line
HTTP Method(s)	POST	
Request Body Parameters	`Documents`	Optional, JSON data to modify transaction
Success Response	The response contains the "Destination URL", which can be used to launch the product catalog page to select and configure a product.	

Request Path Sample

```
https://sitename.oracle.com/rest/v3/commerceDocumentsOraclecpqoTransaction/1801
6533/transactionLine/2/actions/_reconfigure_action
```

Request Body Sample

```
{
   "documents": {"_price_quantity": "20"}
}
```

Response Body Sample

```
{
   "destinationURL": "https://sitename.oracle.com/commerce/buyside
/reconfig_form.jsp?action_id=4356762&doc_number=1&document_id=4356178
&id=18016533&destination_url=&_line_item_list=2",
   "documents": {
      "orderOfOpsInputSubDoc": null,
      "_line_bom_item_quantity": null,
      "copyLineItemsInitializationMSM": "Value 3~Value 4"
   }
}
```

ADD FROM CATALOG IN TRANSACTION

The Add from Catalog REST call executes the action defined for Add from Catalog and returns a redirect "Destination" URL to launch the catalog. The Destination URL is used to launch the catalog in CPQ Cloud.

Add from Catalog in Transaction		
Description	This action executes the Add from Catalog action and returns the Destination URL to launch the catalog.	
Request Path	`/rest/v3/commerceDocuments{ProcessVarName}{MainDocVarName}/{transactionId}/actions/{addFromCatalogAction}`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`MainDocVarName`	The variable name of the main document
	`transactionId`	The Commerce Transaction ID
	`addFromCatalogAction`	User-defined Add from Catalog action
HTTP Method(s)	POST	
Request Body Parameters	`Documents`	Optional, JSON data to modify the transaction
	`cacheInstanceId`	Optional, use a "-1" value to load the complete transaction into the user's cache
Success Response	The response contains the Destination URL, which is used to launch the product catalog page to select a catalog item.	

Request Path Sample

```
https://sitename.oracle.com/rest/v3/commerceDocumentsOraclecpqoTransaction/1801
6533/actions/addFromCatalog
```

Request Body Sample

```
{
   "documents": {"displayHistoryLanguage": "en"},
   "cacheInstanceId": "-1"
}
```

Response Body Sample

```
{
   "destinationURL": "http://sitename.oracle.com/commerce
/display_company_profile.jsp?action_id=17905905&document_id=4356178&bsId=180165
33
&versionId=15514396&destination_url=",
   "documents": {
      "items": [{
         "conditionsOperatorsHelp": null,
         "formulaContains": null,
         "reconfigureFormulasInbound": "0"
    }]
}
```

ADD FROM CATALOG IN A TRANSACTION LINE

Add from Catalog in a Transaction Line		
Description	This action executes the Add from Catalog action and returns the Destination URL to launch the catalog.	
Request Path	`/rest/v3/commerceDocuments{ProcessVarName}{MainDocVarName}/{transactionId}/{subDocVarName}/{subDocNumber}/actions/{addFromCatalogAction}`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`MainDocVarName`	The variable name of the main document
	`transactionId`	The Commerce Transaction ID
	`subDocVarName`	The variable name or the sub document
	`subDocNumber`	The sub document number
	`addFromCatalogAction`	User-defined Add from Catalog action
HTTP Method(s)	POST	
Request Body Parameters	`documents`	Optional, JSON data to modify the transaction line
Success Response	The response contains the Destination URL, which is used to launch the product catalog page to select a catalog item.	

Request Path Sample

```
https://sitename.oracle.com/rest/v3/commerceDocumentsOraclecpqoTransaction/1801
6533/transactionLine/2/actions/addFromCatalogSubDoc
```

Request Body Sample

```
{
   "documents": {
      "transactionLine": {
         "items": [{"addFromCatalogFormulaInputSubDoc": "pqrs"}]
      }
   }
}
```

Response Body Sample

```
{
   "destinationURL": "http://sitename.oracle.com/commerce
/display_company_profile.jsp?action_id=17905905&document_id=4356178&bsId=180165
33
&versionId=15514396&destination_url=",
   "documents": {
      "items": [{
         "conditionsOperatorsHelp": null,
         "formulaContains": null,
         "reconfigureFormulasInbound": "0"
      }]
   }
}
```

ADVANCED TRANSACTION ACTIONS

CHANGE CURRENCY

NOTE: The currency can only be changed for transactions without any transaction lines.

Change Currency		
Description	This action changes the currency at the Transaction level.	
Request Path	`/rest/v3/commerceDocuments{ProcessVarName}{MainDocVarName}/{transactionId}/actions/{currencyActionVarName}`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`MainDocVarName`	The variable name of the main document
	`transactionId`	The Commerce Transaction ID
	`currencyActionVarName`	The variable name of the change currency action
HTTP Method(s)	POST	
Request Body Parameters	`currencyCode`	The requested currency code, this item is required
	Additional parameters can be provided to limit the results returned	
Success Response	JSON data containing the transaction with the updated currency	

Request Path Sample

```
https://sitename.oracle.com/rest/v3/commerceDocumentsOraclecpqoTransaction/18498612/actions/_change_currency
```

Request Body Sample

```
{"currencyCode": "USD"}
```

EMAIL ACTIONS

In the CPQ Cloud application, the email feature is implemented using a two-step process. The first action modifies the transaction and returns the objects required to send an email. The second action generates the email attachment and sends the email.

MODIFY TRANSACTION FOR EMAIL

Modify Transaction for Email		
Description	This action modifies and saves a transaction for email.	
Request Path	`/rest/v3/commerceDocuments{ProcessVarName}{MainDocVarName}/{transactionId}/actions/{emailActionVarName}`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`MainDocVarName`	The variable name of the main document
	`transactionId`	The Commerce Transaction ID
	`emailActionVarName`	The variable name of the modify email action
HTTP Method(s)	POST	
Request Body Parameters	`documents`	Optional, JSON data to modify the transaction
	`cacheInstanceId`	Optional, use a "-1" value to load the complete transaction into the user's cache
Success Response	Returns the transaction, email fields, XSL views, attachments, available attachments for merge, generated proposal attachment for merge, and the cache instance ID (if requested)	
Response Body Parameters	`ccEmail`	cc email address(es)
	`fromName`	The sender’s name
	`fromEmail`	The sender’s email address
	`subject`	The email subject line
	`bccEmail`	bcc email address(es)
	`toEmail`	The recipient's email address(es)
	`toName`	The recipient's name
	`comments`	Comments to be sent in the email body
	`xslView`	Printer friendly proposals or history outputs generated from predefined XSL templates
	`attachments`	Attachments to send with the email
	`available AttachmentsForMerge`	The list of attachments which can be merged with the proposal document
	`generatedProposal AttachmentForMerge`	The proposal attachment selected by default. The default ID value for the transaction is -2.
	`cacheInstanceID`	The cache instance ID, if requested

Request Path Sample

```
https://sitename.oracle.com/rest/v3/commerceDocumentsOraclecpqoTransaction/1849
8612/actions/transactionEmailAction
```

Request Body Sample

Change language to German, update the price quantity for documents 2 and 3, and load the complete transaction to the user's cache.

```
{
   "documents": {
      "languageForEmailAndPrint": "de",
      "transactionLine": {
         "items": [
            {"_price_quantity": 33, "_document_number": 3},
            {"_price_quantity": 22, "_document_number": 2}
         ]
      }
   },
   "cacheInstanceId": "-1"
}
```

Response Body Sample

modifyEmailTransaction.txt

SEND EMAIL TRANSACTION

Send Transaction Email		
Description	This action collects email form data, generates the attachment, pushes the email to SMTP servers, and returns confirmation on email sent time. The email recipient receives an email with the comments in the email body and an attachment that contains the transaction information.	
Request Path	`/rest/v3/commerceDocuments{ProcessVarName}{MainDocVarName}/{transactionId}/actions/_send_email`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`MainDocVarName`	The variable name of the main document
	`transactionId`	The Commerce Transaction ID
HTTP Method(s)	POST	
Request Body Parameters	`fromName`	The sender's name
	`ccEmail`	cc email address(es)
	`subject`	The email subject line This parameter is required to send email.
	`bccEmail`	bcc email address(es)
	`toEmail`	The recipient's email address(es) This parameter is required to send email.
	`toName`	The recipient's name
	`comments`	Comments to be sent in the email body
	`actionVariableName`	The email action variable name
	`attachmentIds`	The attachment IDs
	`attachmentIdsForMerge`	The attachment IDs to merge
	`cacheInstanceId`	The cache instance ID
Success Response	JSON with summary message for sending email	

Request Path Sample

```
https://sitename.oracle.com/rest/v3/commerceDocumentsOraclecpqoTransaction/1849
8612/actions/_send_email
```

Request Body Sample

```
{
   "ccEmail": null,
   "fromName": null,
   "subject": "From REST: Quote Email Action Subject",
   "bccEmail": null,
   "toEmail": "firstname.lastname@oracle.com",
   "toName": null,
   "comments": "Comments for Quote Email",
   "actionVariableName": "quoteEmailAction",
   "attachmentIds": ["18763342", "18763343", "18763344"],
   "attachmentIdsForMerge": ["-2", "18763342", "18763577"],
   "cacheInstanceId":
"5J9MTQAbF1ZmcJGnul6y4WOrAydsRpSyR2vvW1o7fWkrEkO89aiFxbEQSIUCEnF4"
}
```

Response Body Sample

```
{
   "message": "Your email was sent on 08/26/2016 4: 42PM",
   "cacheInstanceId":
"5J9MTQAbF1ZmcJGnul6y4WOrAydsRpSyR2vvW1o7fWkrEkO89aiFxbEQSIUCEnF4"
}
```

MODIFY TRANSACTION LINES FOR EMAIL

Modify Transaction Lines for Email		
Description	This action modifies and saves transaction lines for email.	
Request Path	`/rest/v3/commerceDocuments{ProcessVarName}{MainDocVarName}/{transactionId}/{subDocVarName}/{subDocNum}/actions/{emailActionVarName}`	
Request Parameters	`processVarName`	The variable name of the Commerce process
	`MainDocVarName`	The variable name of the main document
	`transactionId`	The Commerce Transaction ID
	`subDocVarName`	The variable name for the sub document
	`subDocNum`	The sub document number
	`emailActionVarName`	The variable name of the modify email action
HTTP Method(s)	POST	
Request Body Parameters	`documents`	Optional, JSON data to modify the transaction line
	`cacheInstanceId`	Optional, use a "-1" value to load the complete transaction into the user's cache
Success Response	Returns the transaction line, email fields, XSL views, attachments, available attachments for merge, generated proposal attachment for merge, and the cache instance ID (if requested)	
Response Body Parameters	`ccEmail`	cc email address(es)
	`fromName`	The sender's name
	`fromEmail`	The sender's email address
	`subject`	The email subject line
	`bccEmail`	bcc email address(es)
	`toEmail`	The recipient's email address(es)
	`toName`	The recipient's name
	`comments`	Comments to be sent in the email body
	`xslView`	Printer friendly proposals or history outputs generated from predefined XSL templates
	`attachments`	Attachments to send with the email
	`available AttachmentsForMerge`	The list of attachments which can be merged with the proposal document
	`generatedProposal AttachmentForMerge`	The proposal attachment selected by default. The default ID value for the transaction is -2.
	`cacheInstanceID`	The cache instance ID, if requested

Request Path Sample

```
https://sitename.oracle.com/rest/v3/commerceDocumentsOraclecpqoTransaction/1849
8612/transactionLine/3/actions/lineEmailAction
```

Request Body Sample

```
{
   "documents": {"_price_quantity": 101},
   "cacheInstanceId": "-1"
}
```

Response Body Sample

modifyEmailTransactionLine.txt

SEND TRANSACTION LINES EMAIL

Send Transaction Lines Email		
Description	This action collects email form data, generates the attachment, pushes the email to SMTP servers, and returns confirmation on email sent time. The email recipient receives an email with the comments in the email body and an attachment that contains the transaction information.	
Request Path	`/rest/v3/commerceDocuments{ProcessVarName}{MainDocVarName}/{transactionId}/{subDocVarName}/{subDocNum}/actions/_send_email`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`MainDocVarName`	The variable name of the main document
	`transactionId`	The Commerce Transaction ID
	`subDocVarName`	The variable name for the sub document
	`subDocNum`	The sub document number
HTTP Method(s)	POST	
Request Body Parameters	`fromName`	The sender's name
	`ccEmail`	cc email address(es)
	`fromName`	The sender’s name
	`subject`	The email subject line This parameter is required to send email.
	`bccEmail`	bcc email address(es)
	`toEmail`	The recipient's email address(es) This parameter is required to send email.
	`toName`	The recipient's name
	`comments`	Comments to be sent in the email body
	`actionVariableName`	The email action variable name
	`attachmentIds`	The attachment IDs
	`attachmentIdsForMerge`	The attachment IDs to merge
	`cacheInstanceId`	The cache instance ID
Success Response	JSON with summary message for sending email	

Request Path Sample

```
https://sitename.oracle.com/rest/v3/commerceDocumentsOraclecpqoTransaction/1849
8612/transactionLine/3/actions/_send_email
```

Request Body Sample

```
{
   "ccEmail": null,
   "fromName": null,
   "subject": "From REST: Line Email Action Subject",
   "bccEmail": null,
   "toEmail": "firstname.lastname@oracle.com",
   "toName": null,
   "comments": "Comments for Send Transaction Line Email",
   "actionVariableName": "lineEmailAction",
   "attachmentIds": ["18763576", "18763577"],
   "attachmentIdsForMerge": ["-2", "18763576", "18763577"],
   "cacheInstanceId":
"8ddwWVX37LMIC9oO1zs0ZOFkmUp93ZFxL5v9s2hT2p1q6Zmph9u1LKjkRLIOasNo"
}
```

Response Body Sample

```
{
   "message": "Your email was sent on 08/26/2016 4: 42PM"
   "cacheInstanceId":
"8ddwWVX37LMIC9oO1zs0ZOFkmUp93ZFxL5v9s2hT2p1q6Zmph9u1LKjkRLIOasNo"
}
```

PRINT ACTIONS

In the CPQ Cloud application, the print feature is implemented using a two-step process. The first step modifies the transaction for print. The second step prints the transaction, using the data collected during modification.

MODIFY TRANSACTION FOR PRINT

Modify Transaction for Print		
Description	This action modifies and saves a transaction for print.	
Request Path	`/rest/v3/commerceDocuments{ProcessVarName}{MainDocVarName}/{transactionId}/actions/{printActionVarName}`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`MainDocVarName`	The variable name of the main document
	`transactionId`	The Commerce Transaction ID
	`printActionVarName`	The variable name of the print preparation action
HTTP Method(s)	POST	
Request Body Parameters	`documents`	Optional, JSON data to modify the transaction
	`cacheInstanceId`	Optional, use a "-1" value to load the complete transaction into the user's cache
Success Response	Returns the transaction, XSL views, available attachments for merge, generated proposal attachment for merge, and the cache instance ID (if requested)	
Response Body Parameters	`xslView`	Printer friendly proposals or history outputs generated from predefined XSL templates
	`available AttachmentsForMerge`	The list of attachments which can be merged with the print document
	`generatedProposal AttachmentForMerge`	The Attachment Selected by Default. The default ID value for the transaction is -2.
	`cacheInstanceID`	The cache instance ID, if requested

Request Path Sample

```
https://sitename.oracle.com/rest/v3/commerceDocumentsOraclecpqoTransaction/1849
8612/actions/transactionPrintAction
```

Request Body Sample

Change language to German, update the price quantity for documents 2 and 3, and load the complete transaction to the user's cache.

```
{
   "documents": {
      "languageForEmailAndPrint": "de",
      "transactionLine": {
         "items": [
            {"_price_quantity": 33, "_document_number": 3},
            {"_price_quantity": 22, "_document_number": 2}
         ]
      }
   },
   "cacheInstanceId": "-1"
}
```

Response Body Sample

modifyPrintTransaction.txt

PRINT TRANSACTION

Print Transaction		
Description	This action validates user access and collects transaction, XSL, and language settings. If requested, the print document is saved and printed to the file attachment attribute in the transaction.	
Request Path	/rest/v3/commerceDocuments{ProcessVarName}{MainDocVarName}/{transactionId}/actions/_print_document	
Request Path Parameters	processVarName	The variable name of the Commerce process
	MainDocVarName	The variable name of the main document
	transactionId	The Commerce Transaction ID
HTTP Method(s)	POST	
Request Body Parameters	actionVarName	The print preparation action variable name
	saveToQuote	Indicates if the print document is saved to the transaction, valid values are {true, false}
	Note: If this value conflicts with the setting in the print action (e.g. the "Save Document Upon Printing - Desktop", an error is returned.	
	attachmentsIdsForMerge	The attachment IDs to merge
	cacheInstanceID	The cache instance ID
Success Response	JSON with summary message for printing transaction	
Response Body Parameters	fileContent	The Base64 encoded binary document value
	fileContentType	The file content type
	fileTitle	The file name used when saving to the transaction file attachment attribute, or when downloading the file
	fileAttachmentAttribute	Used for confirmation
	cacheInstanceID	The cache instance ID

Request Path Sample

```
https://sitename.oracle.com/rest/v3/commerceDocumentsOraclecpqoTransaction/1849
8612/actions/_print_document
```

Request Body Sample

```
{
   "actionVariableName": "transactionPrintAction",
   "saveToQuote": "true",
   "attachmentIdsForMerge": ["-2", "18763342", "18763344", "18763576",
"18763577"],
   "cacheInstanceId":
"sKtuHHB4bc2LkKSjmJEK6Gl2xaRXI49bOWCWjuwOPMDcqRYyhEn355YbKRQ3fbcs"
}
```

Response Body Sample

```
{
   "fileContent": "base64 encoded binary document value here",
   "fileContentType": "application/pdf",
   "fileTitle": "Document.pdf",
   "fileAttachmentAttribute": "File Attachment",
   "cacheInstanceId":
"sKtuHHB4bc2LkKSjmJEK6Gl2xaRXI49bOWCWjuwOPMDcqRYyhEn355YbKRQ3fbcs"}
```

MODIFY TRANSACTION LINES FOR PRINT

Modify Transaction Lines for Print		
Description	This action modifies and saves transaction lines for print.	
Request Path	/rest/v3/commerceDocuments{ProcessVarName}{MainDocVarName}/{transactionId}/{subDocVarName}/{subDocNum}/actions/_print_document	
Request Path Parameters	processVarName	The variable name of the Commerce process
	MainDocVarName	The variable name of the main document
	transactionId	The Commerce Transaction ID
	subDocVarName	The variable name for the sub document
	subDocNum	The sub document number
HTTP Method(s)	POST	
Request Body Parameters	documents	Optional, JSON data to modify the transaction line
	cacheInstanceId	Optional, use a "-1" value to load the complete transaction into the user's cache
Success Response	Returns the transaction line, XSL views, available attachments for merge, generated proposal attachment for merge, and the cache instance ID (if requested)	
Response Body Parameters	documents	The JSON transaction
	xslView	Printer friendly proposals or history outputs generated from predefined XSL templates
	available AttachmentsForMerge	The list of attachments which can be merged with the print document
	generatedProposal AttachmentForMerge	The Attachment Selected by Default. The default value for the transaction is -2.
	cacheInstanceID	The cache instance ID, if requested

Request Path Sample

```
https://sitename.oracle.com/rest/v3/commerceDocumentsOraclecpqoTransaction/1849
8612/transactionLine/3/actions/linePrintAction
```

Request Body Sample

```
{
   "documents": {"_price_quantity": 101},
   "cacheInstanceId": "-1"
}
```

Response Body Sample

```
{
   "documents": {
      "items": [{
            "_price_quantity": 101,
            "_document_number": 2
         }
      ]
   },
   "xslView": "Print RTE (PDF)(Printer Friendly/History)",
   "generatedProposalAttachmentForMerge": [{
         "id": "-2",
         "name": "GENERATED PROPOSAL"
      }
   ],
   "availableAttachmentsForMerge": [{
         "id": "18763576",
         "name": "Merge1.pdf (Transaction Line - File Attachment Sub Doc 1)"
      }, {
         "id": "18763577",
         "name": "Merge2.pdf (Transaction Line - File Attachment Sub Doc 2)"
      }
   ],
   "fileAttachmentAttribute": "File Attachment",
   "cacheInstanceId":
"8ddwWVX37LMIC9oO1zs0ZOFkmUp93ZFxL5v9s2hT2p1q6Zmph9u1LKjkRLIOasNo"
}
```

PRINT TRANSACTION LINES

Print Transaction Lines		
Description	This action validates user access and collects transaction, XSL, and language settings. If requested, the print document is saved and printed to the file attachment attribute in the transaction.	
Request Path	`/rest/v3/commerceDocuments{ProcessVarName}{MainDocVarName}/{transactionId}/actions/_print_document`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`MainDocVarName`	The variable name of the main document
	`transactionId`	The Commerce Transaction ID
	`subDocVarName`	The variable name for the sub document
	`subDocNum`	The sub document number
HTTP Method(s)	POST	
Request Body Parameters	`printActionVarName`	The print preparation action variable name
	`saveToQuote`	Indicates if the print document is saved to the transaction, valid values are {true, false}
	Note: If this value conflicts with the setting in the print action (e.g. the "Save Document Upon Printing - Desktop", an error is returned.	
	`attachmentsIdsForMerge`	The attachment IDs to merge
	`cacheInstanceID`	The cache instance ID
Success Response	JSON with summary message for printing transaction	
Response Body Parameters	`fileContent`	The Base64 encoded binary document value
	`fileContentType`	The file content type
	`fileTitle`	The file name used when saving to the transaction file attachment attribute, or when downloading the file
	`fileAttachmentAttribute`	Used for confirmation
	`cacheInstanceID`	The cache instance ID

Request Path Sample

```
https://sitename.oracle.com/rest/v3/commerceDocumentsOraclecpqoTransaction/1849
8612transactionLine/2/actions/_print_document
```

Request Body Sample

```
{
   "documents": {
      "actionVariableName": "linePrintAction",
      "saveToQuote": "true",
      "attachmentIdsForMerge": ["-2", "18763576", "18763577"]
   },
   "cacheInstanceId":
"8ddwWVX37LMIC9oO1zs0ZOFkmUp93ZFxL5v9s2hT2p1q6Zmph9u1LKjkRLIOasNo"
}
```

Response Body Sample

```
{
   "fileContent": "base64 encoded binary document value here",
   "fileContentType": "application/pdf",
   "fileTitle": "Document.pdf",
   "fileAttachmentAttribute": "File Attachment",
   "cacheInstanceId":
"8ddwWVX37LMIC9oO1zs0ZOFkmUp93ZFxL5v9s2hT2p1q6Zmph9u1LKjkRLIOasNo"
}
```

INTERACTION RELATED ACTIONS

These actions allow the consumer to interact with a Commerce transaction without saving changes to the database. Interact and Drop Cache actions are included.

INTERACT ACTION

Add Interact is a special action that allows consumer to run auto-update rules. It modifies transaction attributes values and state in the specified cache for a transaction.

In addition to "cacheInstanceId", the "triggerAttrVarName" parameter can be used to run rules based on the trigger attribute. If omitted, no new rules will run. The triggering attribute value is the transaction or transaction line attribute variable name.

Interact Action		
Description	This action modifies transaction attribute values, and optionally triggers auto-update rules without committing or retrieving the latest data from the database. Since this action does not save the transaction to the database, it does not affect how other users see the transaction.	
Request Path	/rest/v3/commerceDocuments{processVarName}{mainDocVarName}/{transactionId}/actions/_interact	
Request Path Parameters	processVarName	The variable name of the Commerce process
	mainDocVarName	The variable name of the main document
	transactionId	The Commerce Transaction ID
HTTP Method(s)	POST	
Request Path Parameters	cacheInstanceId	The cache instance ID
Success Response	The requested cached transaction is returned	

Request Path Sample

Interact with an item within transaction 18330547

```
https://sitename.oracle.com/rest/v3/commerceDocumentsOraclecpqoTransaction/1833
0547/actions/_interact
```

Request Body Sample

Retrieve a cache instance, modify the "_price_quantity" for transaction line 2, and save the updates to cache using "transactionLine/2/_price_quantity" as the trigging attribute

```
{
   "cacheInstanceId":
"E9JuyC9QCgTDS9edghTNxRGOVUm34EMR7RJt9alFbkGTN0X5byTjHGRu1CxrSl1p",
   "triggerAttrVarName": "transactionLine/2/_price_quantity",
   "documents": {
      "transactionLine": {
         "items": [{"_price_quantity": "100", "_document_number": "2"}]
      }
   }
}
```

DROP CACHE

Drop Cache		
Description	This action clears the cache created through rest services. **Note:** Unless dropcache is invoked, user cache remains in memory for the duration of the timeout specified in administration. Leaving cached objects in memory will have a negative effect on site performance.	
Request Path	`/rest/v3/commerceDocuments{processVarName}{mainDocVarName}/{transactionId}/actions/_dropCache`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`mainDocVarName`	The variable name of the main document
	`transactionId`	The Commerce Transaction ID
HTTP Method(s)	POST	
Request Path Parameters	`cacheInstanceId`	The cache instance ID
Success Response	The requested cache instance is cleared.	

Request Path Sample

```
https://sitename.oracle.com/rest/v3/commerceDocumentsOraclecpqoTransaction/1833
0547/actions/_dropCache
```

Request Body Sample

```
{"cacheInstanceId":
"E9JuyC9QCgTDS9edghTNxRGOVUm34EMR7RJt9alFbkGTN0X5byTjHGRu1CxrSl1p"}
```

LAYOUT SERVICE

This REST service returns CPQ transaction desktop layout definitions for UI frameworks, like Oracle ADF (Application Development Framework) and Salesforce.com Lightning UI, to consume. The desktop layout definition contains the following items:

- Layout parameters and properties for panels, tabs, columns, action strips, and transaction line grids
- Properties for actions and attributes
- Translations for layout items

GET TRANSACTION LAYOUT

Get Transaction Layout		
Description	This action returns the desktop layout definition for a specific Commerce process.	
Request Path	`/rest/v3/commerceProcesses/{processVarName}/layouts/{mainDocVarName}`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`mainDocVarName`	The variable name of the main document
HTTP Method(s)	GET	
Request Body Parameters	None	
Success Response	JSON data containing the desktop layout definition for the requested Commerce process	

Request Path Sample

```
https://sitename.oracle.com/rest/v3/commerceProcesses/oraclecpqo/layouts/transaction
```

METADATA SERVICES

The metadata endpoints expose information about the definitions of the Commerce main document, sub document, actions, attributes, and translations. The information for these items is organized in a hierarchical structure. The following table displays the parent child relationship for main documents, sub documents, actions, attributes, menu items, and translations.

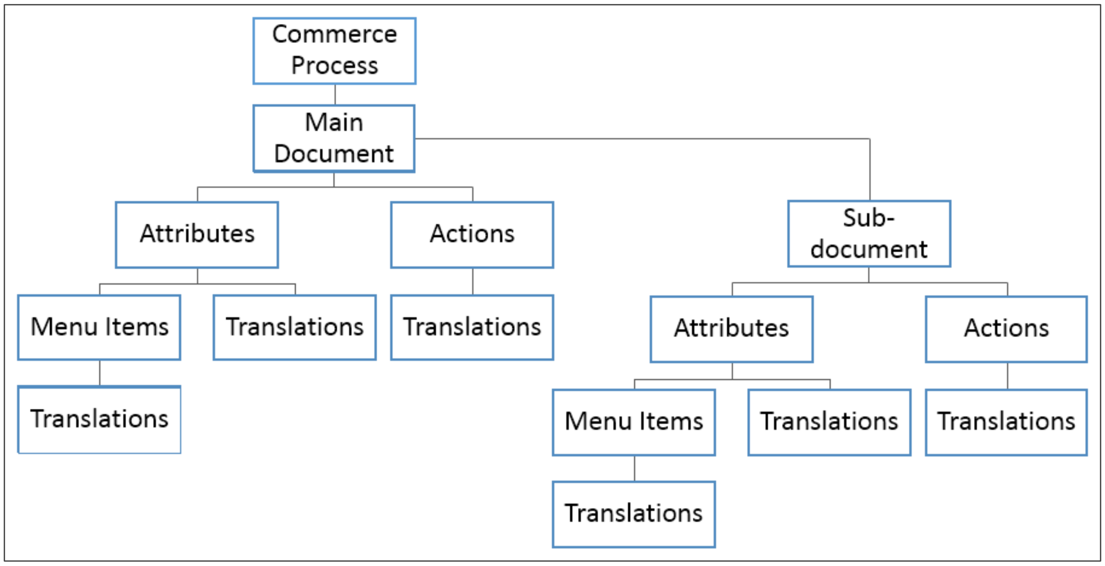


Figure 5: Document Hierarchical Structure

METADATA APIS

A REST call to retrieve data for an object will return object's information and links to the child objects. For example, "Get Metadata for a Document" will return the object's definition information and links to actions and attributes. Request path queries can be used to expand or limit the returned data.

- The "Expand All" query parameter can be used to return expanded actions, attributes, menu items, and translations as part of the response. To include child data in the response, append the "?expand=all*" query parameter to the end of the request path.
- Refer to CPQ Cloud Online Help - REST APIs for additional request path query parameters.

DOCUMENT METADATA

This service returns the document metadata for the requested Commerce document. Links to the current REST call, actions, and attributes are included in the response.

GET DOCUMENTS METADATA

Get Documents Metadata		
Description	This service returns the document metadata for all Commerce documents.	
Request Path	`/rest/v3/commerceProcesses/{processVarName}/documents`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
HTTP Method(s)	GET	
Request Body Parameters	None	
Success Response	JSON data containing the document metadata for all documents	

Request Path Sample

```
https://sitename.oracle.com/rest/v3/commerceProcesses/oraclecpqo/documents
```

Response Body Sample

getAllDocsMetadata.txt

GET DOCUMENT METADATA

Get Document Metadata		
Description	This service returns the metadata for a specific Commerce document.	
Request Path	`/rest/v3/commerceProcesses/{processVarName}/documents/{docVarName}`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`docVarName`	The variable name of the Commerce document
HTTP Method(s)	GET	
Request Body Parameters	None	
Success Response	JSON data containing the document metadata for the specified document	

USE CASE EXAMPLES

1. Get document metadata for "transaction"

   Request Path Sample

   ```
   https://sitename.oracle.com/rest/v3/commerceProcesses/oraclecpqo/documents/tran
   saction
   ```

   Response Body Sample

   getDocMetadata.txt

2. Get document metadata for "transaction" using the 'Expand All' parameter to include metadata for all related actions, attributes, and translations

   Request Path Sample

   ```
   https://sitename.oracle.com/rest/v3/commerceProcesses/oraclecpqo/documents
   /transaction?expand=all*
   ```

ACTION METADATA

This service returns the action metadata for the specified Commerce main document or sub document. The action metadata is comprised of the following information:

- The defined action properties that are defined on the Admin Action page
- Possible action states, such as whether an action can be visible or enabled

- Dependencies and possible states, if an action can be altered by a step or rule
- Links to the current REST call, parent items, and translations

NOTE: Process actions are only returned under the main document and are not accessible for process or sub document requests. Standard process actions include copy, email, print, or create new transaction.

GET ACTIONS METADATA

Get Actions Metadata		
Description	This service returns the metadata for all actions defined for the specified Commerce document.	
Request Path	`/rest/v3/commerceProcesses/{processVarName}/documents/{docVarName}/actionDefs`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`docVarName`	The variable name of the document
HTTP Method(s)	GET	
Request Body Parameters	None	
Success Response	JSON data containing metadata for all actions related to the specified document	

Request Path Samples

1. Get metadata for all "transaction" actions

```
https://sitename.oracle.com/rest/v3/commerceProcesses/oraclecpqo/documents
/transaction/actionDefs
```

2. Get metadata for all "transaction" actions using the 'Expand All' parameter to include all translations

```
https://sitename.oracle.com/rest/v3/commerceProcesses/oraclecpqo/documents
/transaction/actionDefs?expand=all*
```

GET ACTION METADATA

Get Action Metadata		
Description	This service returns the metadata for a specific Commerce action.	
Request Path	`/rest/v3/commerceProcesses/{processVarName}/documents /{docVarName}/actionDefs/{actionVarName}`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`docVarName`	The variable name of the document
	`actionVarName`	The variable name of a Commerce action
HTTP Method(s)	GET	
Request Body Parameters	None	
Success Response	JSON data containing the action metadata for a specific Commerce action	

Request Path Sample

Get action metadata for "update_line_items"

```
https://sitename.oracle.com/rest/v3/commerceProcesses/oraclecpqo/documents
/transaction/actionDefs/_update_line_items
```

Response Body Sample

getActionMetadata.txt

ATTRIBUTE METADATA

This service returns the attribute metadata for the specified Commerce document. The attribute metadata is comprised of the following information:

- The defined attribute properties that are defined on the Attribute Editor page
- Possible attribute states, such as whether an attribute is updatable, required, or visible. Other attribute states are domain, messages, hasError, hasWarning, or hasContraintViolation.
- Dependencies and possible states, if an attribute can be altered by a formula, step, or rule
- Links to the current REST call, parent items, menu items, and translations

GET ATTRIBUTES METADATA

Get Attributes Metadata		
Description	This service returns the metadata for all attributes defined for the specified Commerce document.	
Request Path	`/rest/v3/commerceProcesses/{processVarName}/documents/{docVarName}/attributes`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`docVarName`	The variable name of the document
HTTP Method(s)	GET	
Request Body Parameters	None	
Success Response	JSON data containing metadata for all attributes related to the specified document	

Request Path Samples

1. Get metadata for all "transaction" attributes

```
https://sitename.oracle.com/rest/v3/commerceProcesses/oraclecpqo/documents
/transaction/attributes
```

2. Get metadata for all "transaction" attributes using the 'Expand All' parameter to include all translations

```
https://sitename.oracle.com/rest/v3/commerceProcesses/oraclecpqo/documents
/transaction/attributes?expand=all*
```

GET ATTRIBUTE METADATA

Get Attribute Metadata		
Description	This service returns the metadata for a specific Commerce attribute.	
Request Path	`/rest/v3/commerceProcesses/{processVarName}/documents/{docVarName}/attributes/{attributeVarName}`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`docVarName`	The variable name of the document
	`attributeVarName`	The variable name of a Commerce attribute
HTTP Method(s)	GET	
Request Body Parameters	None	
Success Response	JSON data containing the attribute metadata for a specific Commerce attribute	

Request Path Sample

Get attribute metadata for "_customer_t_email "

```
https://sitename.oracle.com/rest/v3/commerceProcesses/oraclecpqo/documents/tran
saction/attributes/_customer_t_email
```

Response Body Sample

getAttributeMetadata.txt

MENU ITEM METADATA

This service returns the menu item metadata for the specified attribute. The menu item metadata is comprised of the following information:

- Menu items are returned as composite values containing menu properties and domain items
- Links to the current REST call, parent items, and translations

GET MENU ITEMS METADATA

Get Menu Items Metadata		
Description	This service returns the metadata for all menu items defined for the specified Commerce attribute.	
Request Path	`/rest/v3/commerceProcesses/{processVarName}/documents/{docVarName}/attributes/{attributeVarName}/menuItems`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`docVarName`	The variable name of the document
	`attributeVarName`	The variable name of the Commerce menu type attribute
HTTP Method(s)	GET	
Request Body Parameters	None	
Success Response	JSON data containing metadata for all menu items related to the specified attribute	

Request Path Sample

Get menu item metadata for " paymentTerms_t "

```
https://sitename.oracle.com/rest/v3/commerceProcesses/oraclecpqo/documents/transaction/attributes/paymentTerms_t/menuItems
```

Response Body Sample

getAllmenuItemsMetadata.txt

GET MENU ITEM METADATA

Get Menu Item Metadata		
Description	This service returns the metadata for a specific menu item.	
Request Path	`/rest/v3/commerceProcesses/{processVarName}/documents /{docVarName}/attributes/{attributeVarName}/menuItems/{id}`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`docVarName`	The variable name of the document
	`attributeVarName`	The variable name of the Commerce menu type attribute
	`Id`	The id of the menu item
HTTP Method(s)	GET	
Request Body Parameters	None	
Success Response	JSON data containing the menu item metadata for a specific menu item	

Request Path Sample

Get metadata for the "paymentTerms_t" menu item 36244405

```
https://sitename.oracle.com/rest/v3/commerceProcesses/oraclecpqo/documents/tran
saction/attributes/paymentTerms_t/menuItems/36244405
```

Response Body Sample

getMenuItemMetadata.txt

TRANSLATION SERVICES

The translation services retrieve translations for actions, attributes, and menu items. Translations for all site-supported languages can be retrieved. If a translation is not defined, an empty string is returned for the requested value. The following fields are translatable:

- Actions - labels and descriptions
- Attributes - labels, descriptions, and user defaults
- Menu Items - display values

GET ACTION TRANSLATIONS

This service returns the user-defined translations for action labels and descriptions. This service also returns links to the current REST call and the parent action item.

Get Action Translations		
Description	This service returns all language translations for an action.	
Request Path	`/rest/v3/commerceProcesses/{processVarName}/documents/{docVarName}/actionDefs/{actionVarName}/translations`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`docVarName`	The variable name of the document
	`actionVarName`	The variable name of the Commerce action
HTTP Method(s)	GET	
Request Body Parameters	None	
Success Response	JSON data containing all translations for an action	

Request Path Sample

Get all translations for "_update_line_items" action

```
https://sitename.oracle.com/rest/v3/commerceProcesses/oraclecpqo/documents/tran
saction/actionDefs/_update_line_items/translations
```

GET ACTION TRANSLATION

Get Action Translation		
Description	This service returns a specific language translation for an action.	
Request Path	`/rest/v3/commerceProcesses/{processVarName}/documents/{docVarName}/actionDefs/{actionVarName}/translations/{languageCode}`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`docVarName`	The variable name of the document
	`actionVarName`	The variable name of the Commerce action
	`languageCode`	The language code
HTTP Method(s)	GET	
Request Body Parameters	None	
Success Response	JSON data containing specific language translation for an action	

Request Path Sample

Get French translation for "_update_line_items" action

```
https://sitename.oracle.com/rest/v3/commerceProcesses/oraclecpqo/documents/tran
saction/actionDefs/_update_line_items/translations/fr
```

Response Body Sample

```
{
   "language": {"languageCode": "fr", "languageNumber": 1},
   "links": [{
         "rel": "self",
         "href":
"https://sitename.oracle.com/rest/v3/commerceProcesses/oraclecpqo/documents
/transaction/actionDefs/_update_line_items/translations/fr"
      }, {
         "rel": "parent",
         "href":
"https://sitename.oracle.com/rest/v3/commerceProcesses/oraclecpqo/documents
/transaction/actionDefs/_update_line_items"
      }
   ],
   "label": "Mettre à jour les lignes de revenu",
   "description": "Action par défaut de mise à jour des lignes de revenu"
}
```

GET ATTRIBUTE TRANSLATIONS

This service returns the user-defined translations for an attribute as part of the REST response. This service also returns links to the current REST call and the parent attribute item.

Get Attribute Translations		
Description	This service returns all language translations for an attribute	
Request Path	`/rest/v3/commerceProcesses/{processVarName}/documents/docVarName/attributes/{attributeVarName}/translations`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`docVarName`	The variable name of the document
	`attributeVarName`	The variable name of the Commerce attribute
HTTP Method(s)	GET	
Request Body Parameters	None	
Success Response	JSON data containing all translations for an attribute	

Request Path Sample

Get all translations for "_customer_t_email" attribute

```
https://sitename.oracle.com/rest/v3/commerceProcesses/oraclecpqo/documents/transaction/attributes/_customer_t_email/translations
```

GET ATTRIBUTE TRANSLATION

Get Attribute Translation		
Description	This service returns the specified language translation for an attribute	
Request Path	`/rest/v3/commerceProcesses/{processVarName}/documents/{docVarName}/attributes/{attributeVarName}/translations/{languageCode}`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`docVarName`	The variable name of the document
	`attributeVarName`	The variable name of the Commerce attribute
	`languageCode`	The language code
HTTP Method(s)	GET	
Request Body Parameters	None	
Success Response	JSON data containing the specified language translation information for an attribute	

Request Path Sample

Get German translation for "_customer_t_email " attribute

```
https://sitename.oracle.com/rest/v3/commerceProcesses/oraclecpqo/documents/tran
saction/attributes/_customer_t_email/translations/de
```

Response Body Sample

```
{
   "language": {"languageCode": "de", "languageNumber": 0},
   "links": [{
         "rel": "self",
         "href":
https://sitename.oracle.com/rest/v3/commerceProcesses/oraclecpqo/documents
/transaction/attributes/_customer_t_email/translations/de
      }, {
         "rel": "parent",
         "href":
https://sitename.oracle.com/rest/v3/commerceProcesses/oraclecpqo/documents
/transaction/attributes/_customer_t_email
      }
   ],
   "description": "",
   "label": "E-Mail Kunde",
   "userDefault": ""
}
```

GET MENU ITEM TRANSLATIONS

This service returns the user-defined translation for a menu item as part of the REST response. This service also returns the links to the current REST call and the parent menu item.

Get Menu Item Translations		
Description	This service returns all translations for a menu item.	
Request Path	`/rest/v3/commerceProcesses/{processVarName}/documents/{docVarName}/attributes/{attributeVarName}/menuItems/{id}/translations`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`docVarName`	The variable name of the document
	`attributeVarName`	The variable name of the attribute
	`id`	The menu item id number
HTTP Method(s)	GET	
Request Body Parameters	None	
Success Response	JSON data containing all translations for a menu item	

Request Path Sample

Get all translations for "paymentTerms_t" menu item 3624405

```
https://sitename.oracle.com/rest/v3/commerceProcesses/oraclecpqo/documents/tran
saction/attributes/paymentTerms_t/menuItems/36244405/translations
```

GET MENU ITEM TRANSLATION

Get Menu Item Translation		
Description	This service returns the specified language translation for a menu item.	
Request Path	`/rest/v3/commerceProcesses/{processVarName}/documents/{docVarName}/attributes/{attributeVarName}/menuItems/{id}/translations/{languageCode}`	
Request Path Parameters	`processVarName`	The variable name of the Commerce process
	`docVarName`	The variable name of the document
	`attributeVarName`	The variable name of the Commerce attribute
	`id`	The id of the menu item
	`languageCode`	The language code
HTTP Method(s)	GET	
Request Body Parameters	None	
Success Response	JSON data containing the specified language translation for a menu item	

Request Path Sample

Get Spanish translation for "paymentTerms_t" menu item 3624405

```
https://sitename.oracle.com/rest/v3/commerceProcesses/oraclecpqo/documents/tran
saction/attributes/paymentTerms_t/menuItems/36244405/translations/es
```

Response Body Sample

```
{"language": {"languageCode": "es", "languageNumber": 2},
   "links": [{
         "rel": "self",
         "href": "
https://sitename.oracle.com/rest/v3/commerceProcesses/oraclecpqo/documents
/transaction/attributes/paymentTerms_t/menuItems/36244405/translations/es"
      }, {
         "rel": "parent",
         "href": "
https://sitename.oracle.com//rest/v3/commerceProcesses/oraclecpqo/documents
/transaction/attributes/paymentTerms_t/menuItems/36244405"
      }
   ],
   "displayValue": "Neto a 60"
}
```

STEPS TO ENABLE

The Transaction REST APIs enhanced features are available on 2016 R2 sites using v3 RESTful services.

KEY RESOURCES

The CPQ Cloud Interface Catalog provides the framework to expose and describe the Transaction REST APIs. To access the Interface Catalog navigate to Admin > Interface Catalogs.

For additional resources, refer to the following CPQ Cloud Administration Online Help topics:

- Integrating with CPQ Cloud > REST APIs
- Integrating With CPQ Cloud > Integration Guides > Interface Catalog

EXTERNAL CONFIGURATOR REST API

This REST API provides support for the External Configurator Integration feature available in CPQ Cloud 2016 R2. The REST service invokes the External Configurator Advanced Save function to create a transaction, Add from Catalog, or Reconfigure.

- If External Configuration is enabled for the model, the return result updates a transaction.
    - A new transaction is created if the Transaction ID is not provided.
    - Add from Catalog is performed if the Transaction ID is provided, but the document number is not provided.
    - Reconfigure is performed if the Transaction ID and the document number are provided.
- If External Configuration is not enabled for the model, an error is returned.

External Configurator REST API		
Description	This service invokes the External Configurator Advanced Save function.	
Request Path	/rest/v3/productFamilies/{pfVarName}/productLines/{plVarName}/models/{modelVarName}/actions/_externalConfigurationSave	
Request Path Parameters	pfVarName	The variable name of the product family
	plVarName	The variable name of the product line
	modelVarName	The variable name of the model
HTTP Method(s)	POST	
Success Response	Returns the Transaction ID, document number, status, and a link to the saved transaction	
Request Body Parameters	bsId	The CPQ Transaction ID Required for Add from Catalog and Reconfigure
	documentNumber	The document number of a model line Required for Reconfigure
	reconfigActionVarName	The Reconfigure Action Variable Name Required for Reconfigure If this value is not provided, the default "_reconfigure_action" is used
	reconfigSource	The entity to be reconfigured Required for Reconfigure Valid values: Quote and Line Item; if this value is not provided, Quote is used as the default value
	parameters	Generic string parameters passed to the Save BML "ext_config_url_params" for an external configuration rule.

USE CASE EXAMPLES

Request Body Sample

The following sample is a Reconfigure request.

```
{
   "bsId": 12345,
   "documentNumber": 2,
   "reconfigActionVarName": "_reconfigure_action",
   "reconfigSource": "Quote",
   "parameters":{
      "configurationHeaderId": 23456,
      "configurationRevisionNumber": 789,
      "configurationPath": "987654321"
   }
}
```

Response Samples

1. **Save Successful**: The Transaction ID, document number, and a link to the saved transaction are returned for a successful save.

   ```
   HTTP status code: 200 OK
   ```

```
{
   "status": "save",
   "bsId": 12345,
   "documentNumber": 2,
   "href": "..."
}
```

2. **Cancel / None:** If the Save BML function is cancelled, the status returned is "cancel". If the Save BML function is not defined, the status returned is "none".

   ```
   HTTP status code: 200 OK
   ```

```
{"status": "cancel"}
```

3. **Error:** When a Save BML function error occurs, the status returned is "error". The response body is a JSON array of the messages returned from the Save BML.

   ```
   HTTP status code: 500
   ```

```
{
   "status": "error",
   [
      {"type": "error",
       "info": "Failed to launch the configuration session."},
      {"type": "warning",
       "info": "invalid configuration attributes while launching the
configuration session."}
   ]
}
```

4. If the error is due to the server-side processing error, the response body is the exception error message.

```
External configuration is not defined for model myModel belonging to product
line myLine under product family myFamily.
```

STEPS TO ENABLE

The External Configurator REST API is automatically available on 2016 R2 sites.

KEY RESOURCES

Refer to the following CPQ Cloud Administration Online Help topics for additional resources:

- Integrating with CPQ Cloud > REST APIs
- Integrating With CPQ Cloud > Integration Guides > Interface Catalog

SOAP API ENHANCEMENTS

The following SOAP API enhancements are available in CPQ Cloud 2016 R2 and support the CPQ Cloud – Commerce Cloud integration and the External Configurator integration.

- Commerce SOAP API
- Configure SOAP API

COMMERCE SOAP API

In CPQ Cloud 2016 R2, changes were made to the Commerce SOAP API to support the CPQ Cloud – Commerce Cloud integration. The configuration SOAP API is invoked for creating transactions with configurable transaction lines. No changes were made directly to the SOAP API. The changes are internal changes that provide the functionality to create transactions denominated in a non-base currency.

CONFIGURE SOAP API

In addition to the above changes to the Commerce SOAP API, 2016 R2 provides the following Configure SOAP API enhancements:

- Currency enhancements allow models with pricing rules defined in the base currency to be used to add lines to a transaction that has a non-base currency. The new lines are converted into the transaction currency.
- The new “getConfigurations" operation supports CPQ Cloud – Commerce Cloud Integration.
- Reconfigure updates enable easy refresh of the configuration data from an external system or from within the CPQ transaction

GET CONFIGURATIONS OPERATION

The “getConfigurations” operation is new in 2016 R2. This operation is used to return configuration data that is saved for a client-side integration. For additional information, refer to the *Integrating Oracle Commerce Cloud Service and CPQ Cloud Service Implementation Guide*.

The following table lists input and response variables for the "getConfigurations" operation.

Input Variables	Response Variables

Required input variables:
`configurationId`
Required for v1 only:
`category`
`schemaLocation`
	The following items are returned for each configuration ID requested:
	• `status`
	• `success`
	• `message`
	• `configurationId`
	• `currency`
	• `item`

	If "responseIncludes" is set to "true", the following items are returned: • `attribute` • `price` • `spare`
Optional input variables • `locale` • `currency`	The following items are returned for when locale and currency are used: • `price` • `spare`

Perform the following steps to generate Input SOAP XML for the "getConfigurations" operation.

1. Navigate to the Admin Home page.
2. Click **Web Services** in the Integration Platform section.
   The **Web Services Test** page appears.
3. Select the appropriate version from the **Web Service Version** drop-down.
4. Click the **Configuration** tab.
5. Select **getConfigurations** from the **API** menu.
6. Enter a **Punchin URL** or select a **Product Family**, **Product Line**, and **Model**.
7. Click **Generate Input** to generate an Input SOAP XML skeleton.

RECONFIGURE UPDATES TO CONFIGURE OPERATION

In 2016 R2, CPQ Cloud adds the reconfigure functionality to the Configure SOAP API "configure" operation. The SOAP API reconfigure functions the same as the UI Reconfigure during Configuration to Quote. It executes the Inbound Reconfigure Action and preserves the recommended items based on the user-defined rules.

NOTE: The Inbound Reconfigure Action can only be executed at the Main document level.

Customers can now use a SOAP API to perform the following actions:

- Refresh a stored configuration in the transaction without returning to the CPQ Configuration UI.
- Retrieve the latest configuration for a configured item sent to downstream systems.

A new document_number attribute has been added to the Input SOAP XML for the Configure SOAP API. This attribute determines whether configure or reconfigure operations are invoked. The following table summarizes behavior actions.

If	Then
A valid document_number value is provided	The reconfigure operation is invoked
An invalid document_number value is provided	An error is reported in the web service response
The document_number is not provided	The configure operation is invoked

Document Number Input Sample

<bm:document_number>2</bm:document_number>

Perform the following steps to generate Input SOAP XML.

1. Navigate to the Admin Home page.
2. Click **Web Services** in the Integration Platform section.
   The **Web Services Test** page appears.
3. Select the appropriate version from the **Web Service Version** drop-down.
4. Click the **Configuration** tab.

5. Select **configure** from the **API** menu.
6. Enter a **Punchin URL** or select a **Product Family**, **Product Line**, and **Model**.
7. Click **Generate Input** to generate an Input SOAP XML skeleton.

STEPS TO ENABLE

The SOAP API enhancements available in CPQ Cloud 2016 R2 are automatically available on CPQ Cloud 2016 R2 sites.

KEY RESOURCES

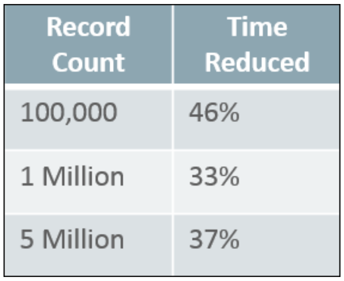
Refer to the following resources for additional information:

- CPQ Cloud Online Help: SOAP Configuration API with Web Service 1.0 and SOAP Configuration API with Web Service 2.0.
- *Integrating Oracle Commerce Cloud Service and CPQ Cloud Service Implementation Guide*

PERFORMANCE ENHANCEMENTS

As part of Oracle's ongoing effort to improve performance of CPQ Cloud, several enhancements are provided in CPQ Cloud Data Tables. With these enhancements, administrators can perform several operations faster, including: bulk uploading and migrating large data tables, uploading XML data tables and CSV files, and deploying large data tables. Administrators can also clear data faster when using the "Clear Data" or "Delete All" actions on existing data tables.

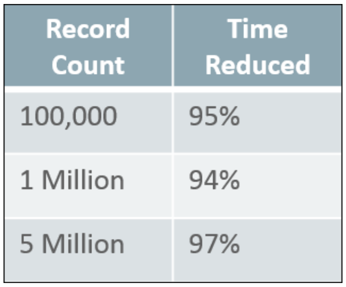
As an example of improved performance when uploading large CSV files, the following figure illustrates that the time taken to upload CSV files with a large number of records was significantly reduced from 2016 R1 to 2016 R2.



Record Count	Time Reduced
100,000	46%
1 Million	33%
5 Million	37%

Figure** 6: **CSV Upload Performance Enhancements

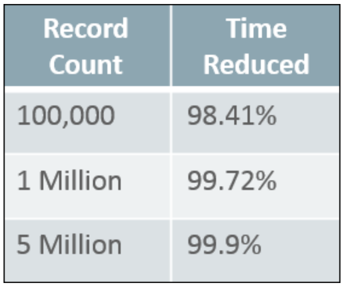
The time taken to deploy large data tables was also significantly reduced from 2016 R1 to 2016 R2.



Record Count	Time Reduced
100,000	95%
1 Million	94%
5 Million	97%

Figure 7: Deploy Large Data Table Performance Enhancements

In addition, the time taken to perform the "Delete All" action on a large number of records was significantly reduced from 2016 R1 to 2016 R2.



Record Count	Time Reduced
100,000	98.41%
1 Million	99.72%
5 Million	99.9%

Figure 8: Delete All Performance Enhancements

STEPS TO ENABLE

Performance Enhancements are automatically available on all 2016 R2 sites.

TIPS AND CONSIDERATIONS

Consider the following tips when using the Performance Enhancements available in CPQ Cloud 2016 R2:

- Order number sorting was removed to improve the performance of the Administration Platform.
- While the "Clear Data" and "Delete All" actions are not new, they now execute faster.

KEY RESOURCES

Refer to the following resources for additional information:

- CPQ Cloud Administration Online Help

INTEGRATED SUITE

Leverage the power of CPQ Cloud by integrating with other software applications. CPQ Cloud administrators can use these pre-defined integrations out-of-the-box or enhance the provided integration patterns to build a strong Oracle Cloud suite.

The following new features and enhancements are available in CPQ Cloud 2016 R2:

- Commerce Cloud Integration
- External Configurator Integration
- Other Enhancements

COMMERCE CLOUD INTEGRATION

CPQ Cloud 2016 R2 provides an integration between CPQ Cloud and Commerce Cloud. Self-service users in Commerce Cloud can configure complex products for purchase in Commerce Cloud using the CPQ Cloud configurator. In addition, Commerce Cloud self-service users can request a CPQ Cloud quote, thereby initiating a CPQ transaction that a sales specialist can modify, reconfigure, or discount. Once finalized in CPQ Cloud, the quote is returned to Commerce Cloud for acceptance and ordering by the self-service user. The integration of CPQ Cloud with Commerce Cloud uses the Oracle Integration Cloud Service (ICS) to provide pre-built integrations for the two user flows.

The integration provides the following functionality to Commerce Cloud self-service users:

- Access a CPQ Cloud Site as guest users
- Add items to a Commerce Cloud cart from CPQ Cloud
- Edit a Commerce Cloud cart from CPQ Cloud
- Request a quote from a CPQ Cloud sales specialist

NOTE: For instructions on how to implement the above functionality, refer to the Integrating Oracle Commerce Cloud Service and CPQ Cloud Service Implementation Guide.

ACCESS A CPQ CLOUD SITE AS GUEST USERS

Commerce Cloud self-service users can access a CPQ Cloud site as guest users via an iFrame that displays within Commerce Cloud. When a CPQ site is accessed by a Commerce Cloud self-service user, session parameters are passed from Commerce Cloud to CPQ Cloud. This provides a seamless user experience and eliminates the need for Commerce Cloud self-service users to enter login credentials when entering a CPQ Cloud site from Commerce Cloud.

Session parameters include currency, language, and locale preferences such as number format, units, and date format. For example: If a Commerce Cloud self-service user's language preference is set to German, the text in the CPQ Cloud interface displays in German when the user accesses CPQ Cloud. The user's currency and locale preferences are also passed from Commerce Cloud and display in CPQ Cloud.

ADD ITEMS TO A COMMERCE CLOUD CART FROM CPQ CLOUD

The "Add to Cart" action sends items to a Commerce Cloud cart via an **Add to Cart** button, which displays on the CPQ Cloud interface following configuration. A "Client Side" integration type is available in CPQ Cloud 2016 R2 and enables the sharing of data between CPQ Cloud and Commerce Cloud. CPQ Cloud administrators must configure a "Client Side" integration to add the **Add to Cart** button on a CPQ Cloud site.

CPQ Cloud administrators must also add payload template files to File Manager. The template files support the "Add to Cart" action and include simple configuration information such as config id, quantity, part, and model name. BML reads these template files and replaces the values in brackets, such as {{quantity}}, with dynamic values.

To add items to a Commerce Cloud cart:

1. The Commerce Cloud administrator adds a **Customize** button to a Commerce Cloud site. The **Customize** button is shown below and is part of the CPQ Cloud – Commerce Cloud integration.

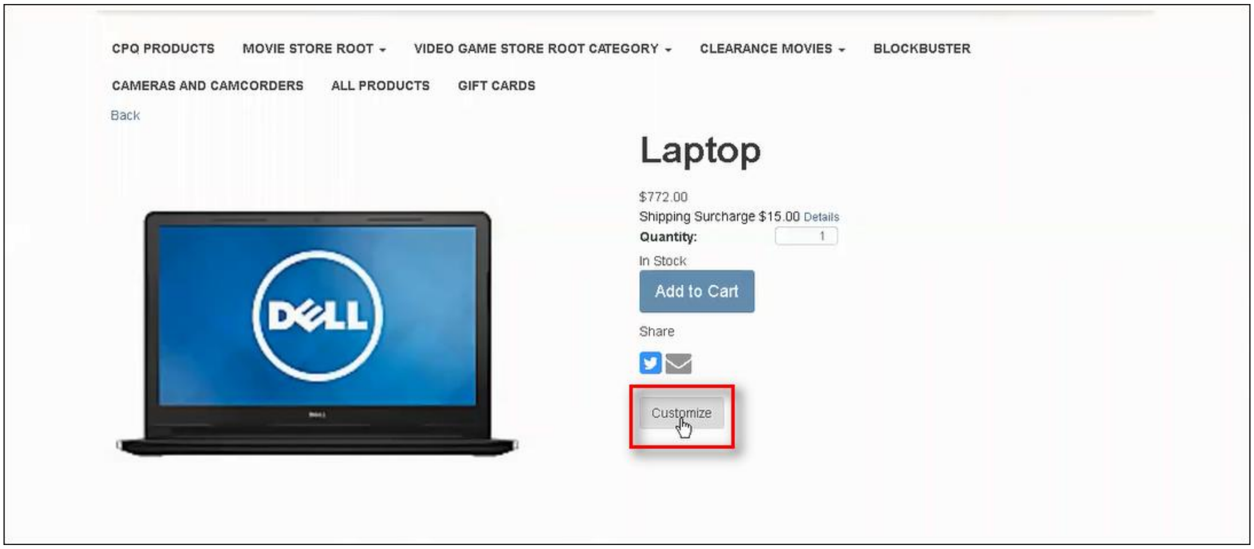
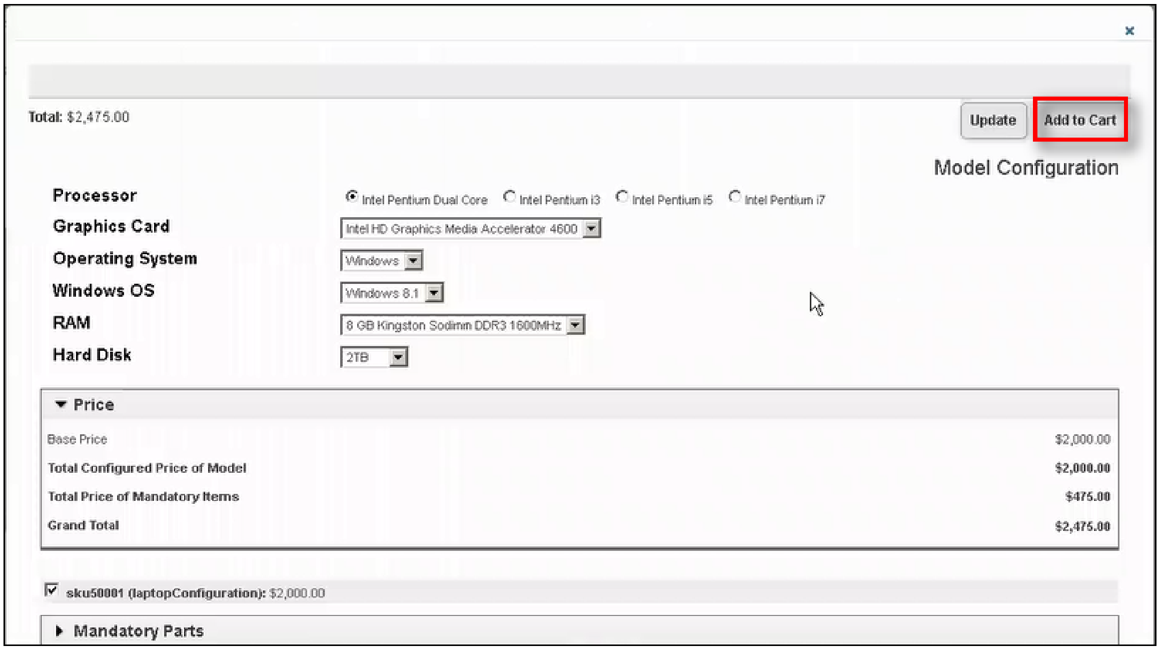


Figure 9: Commerce Cloud Site with Customize Button

2. A Commerce Cloud self-service user clicks **Customize**. The CPQ Cloud **Model Configuration** page opens within an iFrame in Commerce Cloud and contains the **Add to Cart** button.



Total: $2,475.00

Model Configuration

Processor	● Intel Pentium Dual Core ○ Intel Pentium i3 ○ Intel Pentium i5 ○ Intel Pentium i7
Graphics Card	Intel HD Graphics Media Accelerator 4600
Operating System	Windows
Windows OS	Windows 8.1
RAM	8 GB Kingston Sodimm DDR3 1600MHz
Hard Disk	2TB

▾ Price

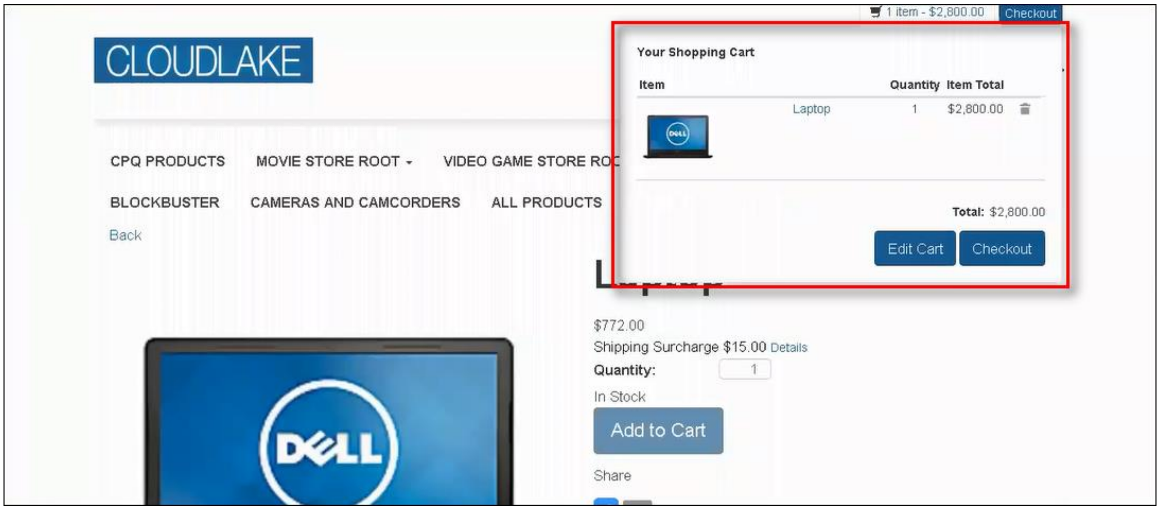
Base Price	$2,000.00
Total Configured Price of Model	$2,000.00
Total Price of Mandatory Items	$475.00
Grand Total	$2,475.00

☑ sku50001 (laptopConfiguration): $2,000.00

▸ Mandatory Parts

Figure 10: CPQ Cloud Model Configuration Page with Add to Cart Button

3. From the CPQ Cloud **Model Configuration** page, the Commerce Cloud self-service user customizes the laptop order and clicks **Add to Cart**. As shown in the following figure, the Commerce Cloud cart is then updated based on the specified customizations.



Your Shopping Cart

Item		Quantity	Item Total
	Laptop	1	$2,800.00

Total: $2,800.00

Laptop

$772.00

Shipping Surcharge $15.00 Details

Quantity: 1

In Stock

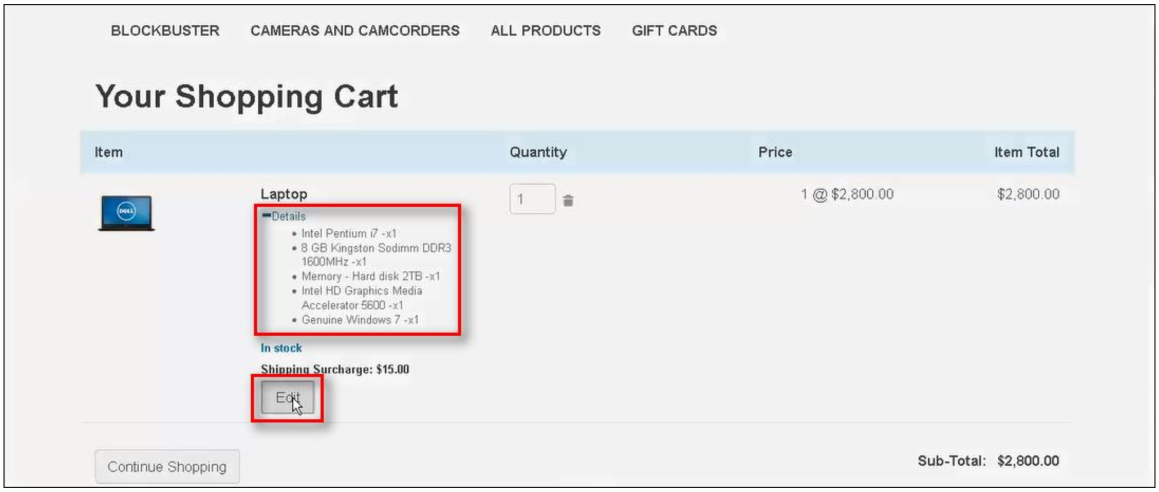
Figure 11: Updated Commerce Cloud Cart

EDIT A COMMERCE CLOUD CART FROM CPQ CLOUD

After a Commerce Cloud self-service user adds an item to a Commerce Cloud cart, the user can return to the shopping cart and edit the details of the items in the shopping cart. All of the CPQ Cloud configured parts that display in the "Details" area of the shopping cart are recommended items.

To edit a Commerce Cloud cart from CPQ Cloud:

1. Open the Commerce Cloud cart.



Your Shopping Cart

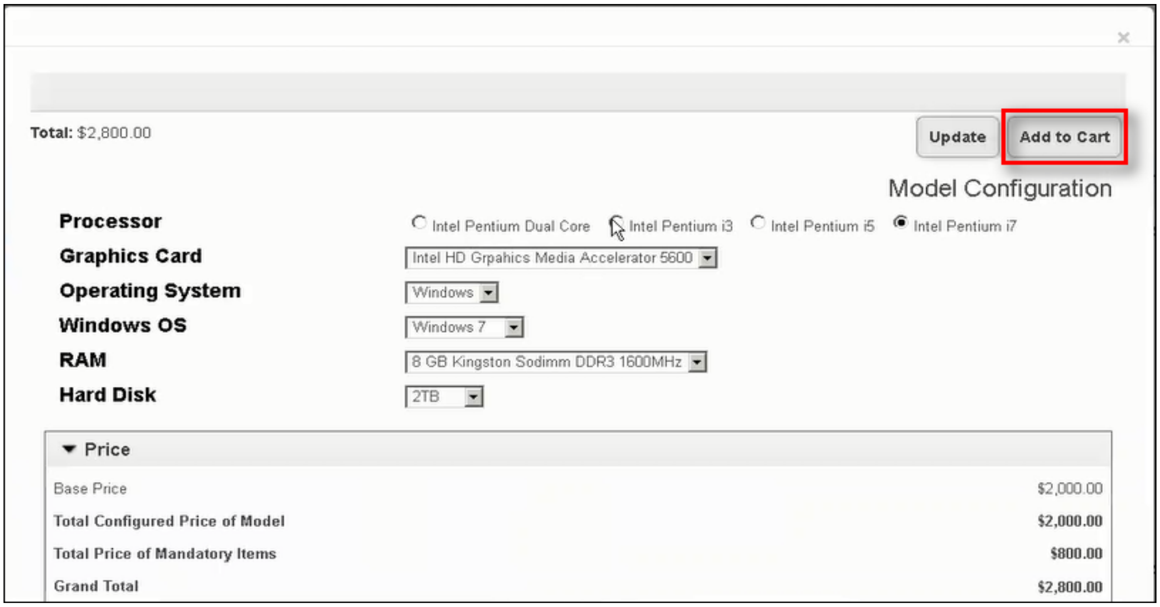
Item		Quantity	Price	Item Total
	Laptop Details • Intel Pentium i7 -x1 • 8 GB Kingston Sodimm DDR3 1600MHz -x1 • Memory - Hard disk 2TB -x1 • Intel HD Graphics Media Accelerator 5600 -x1 • Genuine Windows 7 -x1 In stock Shipping Surcharge: $15.00 Edit	1	1 @ $2,800.00	$2,800.00

Continue Shopping

Sub-Total: $2,800.00

Figure 12: Commerce Cloud Shopping Cart Showing Recommended Items Configured in CPQ Cloud

2. **Click Edit.**
   The CPQ Cloud Model Configuration page opens within an iFrame in Commerce Cloud.



Total: $2,800.00

Model Configuration

Processor: ○ Intel Pentium Dual Core ○ Intel Pentium i3 ○ Intel Pentium i5 ● Intel Pentium i7

Graphics Card: Intel HD Grpahics Media Accelerator 5600

Operating System: Windows

Windows OS: Windows 7

RAM: 8 GB Kingston Sodimm DDR3 1600MHz

Hard Disk: 2TB

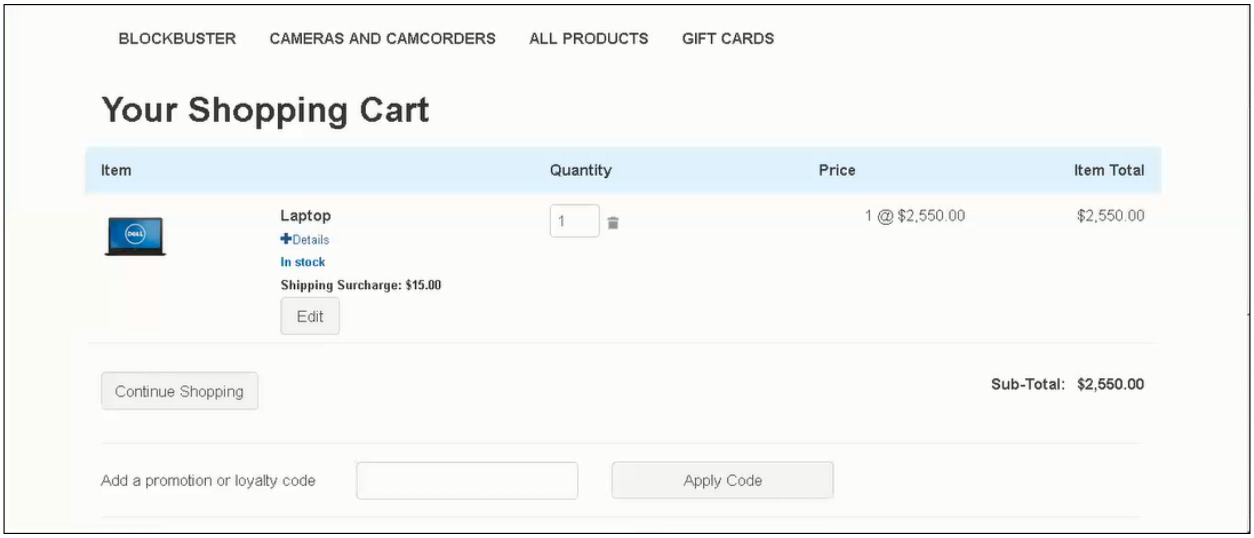
Price

Base Price	$2,000.00
Total Configured Price of Model	$2,000.00
Total Price of Mandatory Items	$800.00
Grand Total	$2,800.00

Figure 13: CPQ Cloud Model Configuration Page

3. Update the Commerce Cloud shopping cart.

4. Click **Add to Cart**.
   The Commerce Cloud shopping cart opens and displays the updates.



BLOCKBUSTER CAMERAS AND CAMCORDERS ALL PRODUCTS GIFT CARDS

Your Shopping Cart

Item	Quantity	Price	Item Total
Laptop +Details In stock Shipping Surcharge: $15.00 Edit	1	1 @ $2,550.00	$2,550.00

Continue Shopping

Sub-Total: $2,550.00

Add a promotion or loyalty code

Apply Code

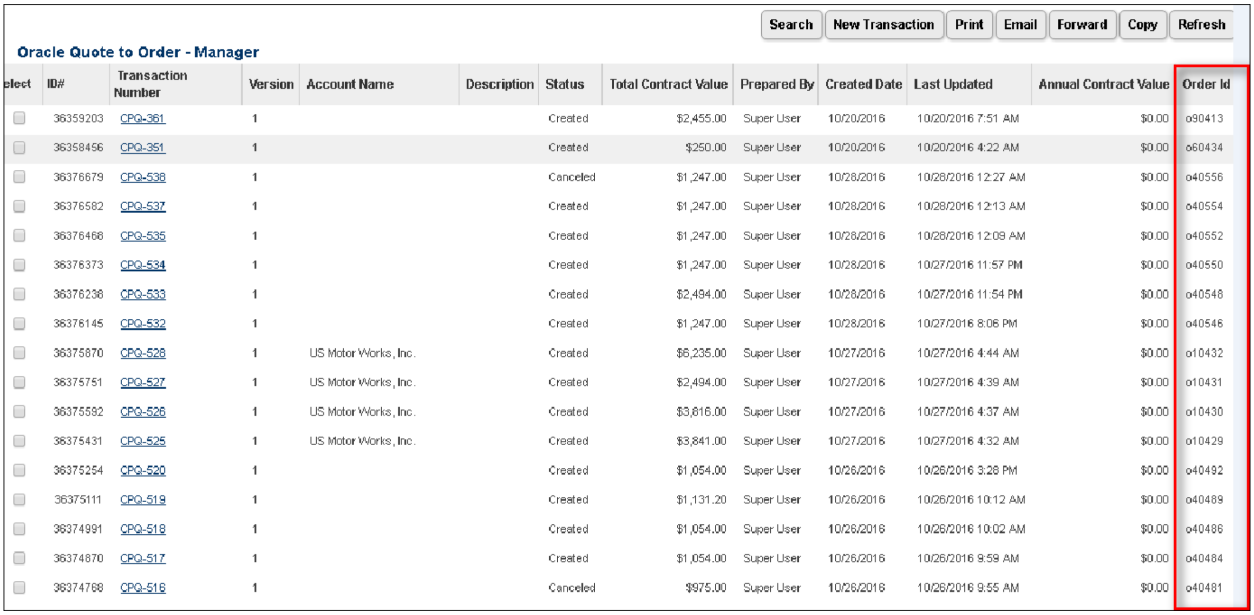
Figure 14: Updated Commerce Cloud Shopping Cart

REQUEST A QUOTE FROM A CPQ CLOUD SALES SPECIALIST

A Commerce Cloud self-service user can select a **Request Quote** option in Commerce Cloud to obtain a quote from a CPQ Cloud sales specialist. The high-level steps related to requesting a quote are provided below. For additional information, refer to the *Integrating Oracle Commerce Cloud Service and CPQ Cloud Service Implementation Guide*.

To request a quote:

1. The Commerce Cloud self-service user selects **Request Quote** from the Commerce Cloud **Create Order** page. The **Request Quote** page opens.
2. From the **Request Quote** page, the Commerce Cloud self-service user has the option of entering a message for the CPQ Cloud sales specialist. For example: The user could enter a message requesting a 15% discount.
3. When the Commerce Cloud self-service user clicks **Request Quote**, an order ID is assigned to the order and the quote request is sent to a CPQ Cloud sales specialist. The CPQ Cloud sales specialist can view the order ID in the last column of the **Oracle Quote to Order – Manager** page.

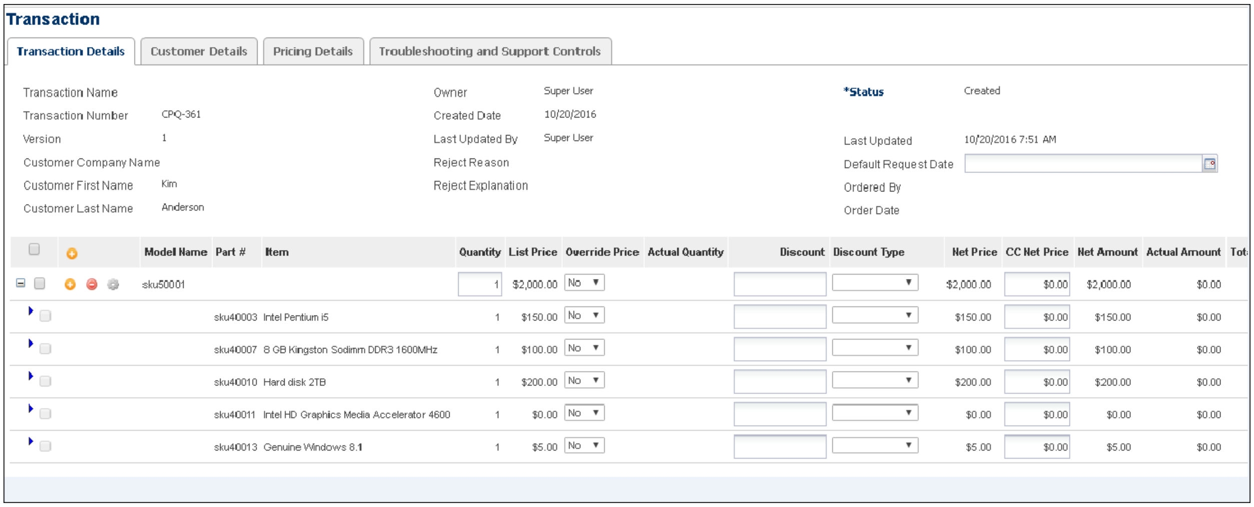


Oracle Quote to Order - Manager

elect	ID#	Transaction Number	Version	Account Name	Description	Status	Total Contract Value	Prepared By	Created Date	Last Updated	Annual Contract Value	Order Id
□	36359203	CPQ-361	1			Created	$2,455.00	Super User	10/20/2016	10/20/2016 7:51 AM	$0.00	o90413
□	36358456	CPQ-351	1			Created	$250.00	Super User	10/20/2016	10/20/2016 4:22 AM	$0.00	o60434
□	36376679	CPQ-538	1			Canceled	$1,247.00	Super User	10/28/2016	10/28/2016 12:27 AM	$0.00	o40556
□	36376582	CPQ-537	1			Created	$1,247.00	Super User	10/28/2016	10/28/2016 12:13 AM	$0.00	o40554
□	36376468	CPQ-535	1			Created	$1,247.00	Super User	10/28/2016	10/28/2016 12:09 AM	$0.00	o40552
□	36376373	CPQ-534	1			Created	$1,247.00	Super User	10/28/2016	10/27/2016 11:57 PM	$0.00	o40550
□	36376238	CPQ-533	1			Created	$2,494.00	Super User	10/28/2016	10/27/2016 11:54 PM	$0.00	o40548
□	36376145	CPQ-532	1			Created	$1,247.00	Super User	10/28/2016	10/27/2016 8:06 PM	$0.00	o40546
□	36375870	CPQ-528	1	US Motor Works, Inc.		Created	$6,235.00	Super User	10/27/2016	10/27/2016 4:44 AM	$0.00	o10432
□	36375751	CPQ-527	1	US Motor Works, Inc.		Created	$2,494.00	Super User	10/27/2016	10/27/2016 4:39 AM	$0.00	o10431
□	36375592	CPQ-526	1	US Motor Works, Inc.		Created	$3,816.00	Super User	10/27/2016	10/27/2016 4:37 AM	$0.00	o10430
□	36375431	CPQ-525	1	US Motor Works, Inc.		Created	$3,841.00	Super User	10/27/2016	10/27/2016 4:32 AM	$0.00	o10429
□	36375254	CPQ-520	1			Created	$1,054.00	Super User	10/26/2016	10/26/2016 3:28 PM	$0.00	o40492
□	36375111	CPQ-519	1			Created	$1,131.20	Super User	10/26/2016	10/26/2016 10:12 AM	$0.00	o40489
□	36374991	CPQ-518	1			Created	$1,054.00	Super User	10/26/2016	10/26/2016 10:02 AM	$0.00	o40486
□	36374870	CPQ-517	1			Created	$1,054.00	Super User	10/26/2016	10/26/2016 9:59 AM	$0.00	o40484
□	36374768	CPQ-516	1			Canceled	$975.00	Super User	10/26/2016	10/26/2016 9:55 AM	$0.00	o40481

Figure 15: Order ID in Oracle Quote to Order – Manager Page

4. The CPQ Cloud sales specialist can then use the **Transaction** page to modify the pricing on the quote. Using the **Sync Quote** action, the CPQ Cloud sales specialist can update the associated Commerce Cloud order.



Transaction

Transaction Name		Owner	Super User	*Status	Created
Transaction Number	CPQ-361	Created Date	10/20/2016		
Version	1	Last Updated By	Super User	Last Updated	10/20/2016 7:51 AM
Customer Company Name		Reject Reason		Default Request Date	
Customer First Name	Kim	Reject Explanation		Ordered By	
Customer Last Name	Anderson			Order Date	

□		Model Name	Part #	Item	Quantity	List Price	Override Price	Actual Quantity	Discount	Discount Type	Net Price	CC Net Price	Net Amount	Actual Amount	Tot
⊟ □		sku50001			1	$2,000.00	No ▼			▼	$2,000.00	$0.00	$2,000.00	$0.00	
□			sku40003	Intel Pentium i5	1	$150.00	No ▼			▼	$150.00	$0.00	$150.00	$0.00	
□			sku40007	8 GB Kingston Sodimm DDR3 1600MHz	1	$100.00	No ▼			▼	$100.00	$0.00	$100.00	$0.00	
□			sku40010	Hard disk 2TB	1	$200.00	No ▼			▼	$200.00	$0.00	$200.00	$0.00	
□			sku40011	Intel HD Graphics Media Accelerator 4600	1	$0.00	No ▼			▼	$0.00	$0.00	$0.00	$0.00	
□			sku40013	Genuine Windows 8.1	1	$5.00	No ▼			▼	$5.00	$0.00	$5.00	$0.00	

Figure 16: Update the Pricing of Quote Using the Transaction Page

5. The order is updated in Commerce Cloud, and the Commerce Cloud self-service user has the ability to accept or reject the proposal or request another quote.

STEPS TO ENABLE

Refer to the *Integrating Oracle Commerce Cloud Service and CPQ Cloud Service Implementation Guide* for detailed instructions on how to prepare CPQ Cloud, Commerce Cloud, and ICS for the integration.

TIPS AND CONSIDERATIONS

Consider the following tip when using the 2016 R2 CPQ Cloud – Commerce Cloud integration:

- The Oracle Commerce Cloud Service and CPQ Cloud Service integration does not support BOM Mapping rules. Administrators should instead use CPQ Cloud Recommended Item Rules, which are used to associate parts and models with products based on user-configured values.

KEY RESOURCES

Refer to the following resources for additional information:

- *Integrating Oracle Commerce Cloud Service and CPQ Cloud Service Implementation Guide*: Provides detailed information about how to prepare CPQ Cloud, Commerce Cloud, and ICS for the integration.
- CPQ Cloud Administration Online Help: Refer to the "Recommended Items" topic and the Integration Guides section of the Online Help.

EXTERNAL CONFIGURATOR INTEGRATION

The External Configurator Integration feature available in CPQ Cloud 2016 R2 is part of an ongoing effort to integrate CPQ Cloud with other products both internal and external to Oracle. Customers can now integrate and leverage the use of an external configurator while using the pricing and quoting capabilities of CPQ Cloud. CPQ Cloud sales specialists can access the external configurator from the CPQ Cloud Home page and then return to CPQ Cloud to price, discount, propose, and order the configured product.

The following External Configurator Integration functionality is available in CPQ Cloud 2016 R2:

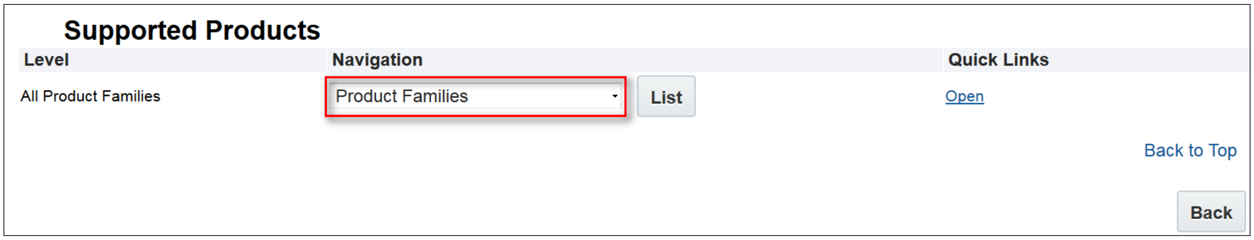
- Define an external configurator integration
- Access an external configurator via Configuration Quick Links
- Use an external configurator Commerce attribute in standard BML functions
- Create or update CPQ Cloud transactions using the External Configurator REST API

DEFINE AN EXTERNAL CONFIGURATOR INTEGRATION

Administrators can define an integration with external configurators such as the Oracle E-Business Suite (EBS) or Fusion. For example: Customers who are using EBS Order Management for order fulfillment can now use the EBS configurator instead of the CPQ Cloud configurator.

To define an external configurator integration:

1. Go to **Admin** > **Products** > **Catalog Definition**.
   The **Supported Products** page opens with **Product Families** selected by default in the **Navigation** drop-down menu.

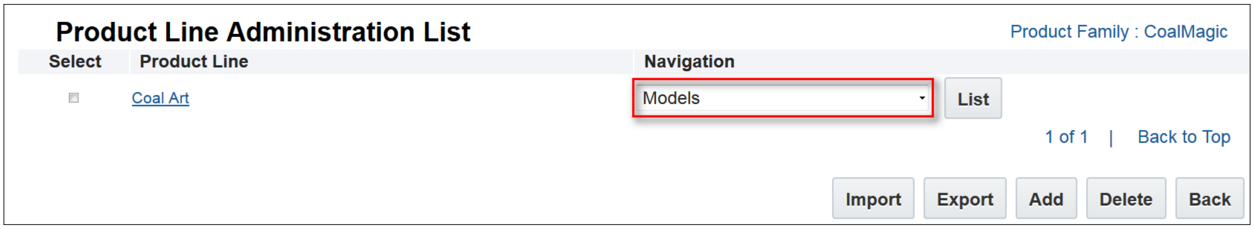


Supported Products

Level	Navigation	Quick Links
All Product Families	Product Families	Open

Figure 17: Supported Products Page

2. Click List.
   The Supported Product Families page opens.
3. Click **List** next to the product family that contains the model you want to integrate with an external configurator. The **Product Line Administration List** page opens with **Models** selected by default for all of the available product lines.



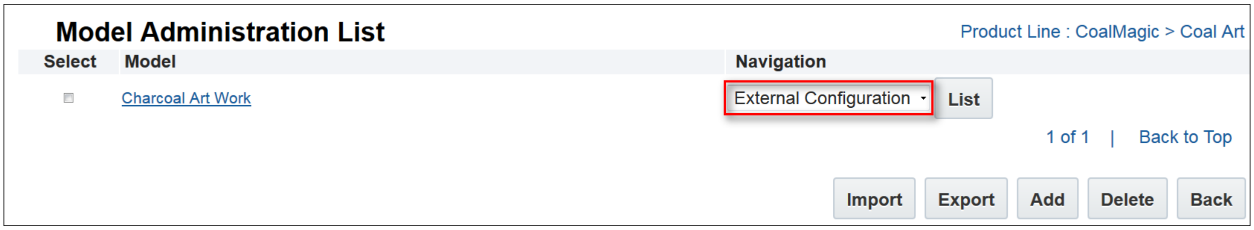
Product Line Administration List

Product Family : CoalMagic

Select	Product Line	Navigation
□	Coal Art	Models

Figure 18: Product Line Administration List

4. Click **List** next to the model you want to integrate with an external configurator.
   The **Model Administration List** page opens.
5. Select **External Configuration** from the **Navigation** drop-down menu.



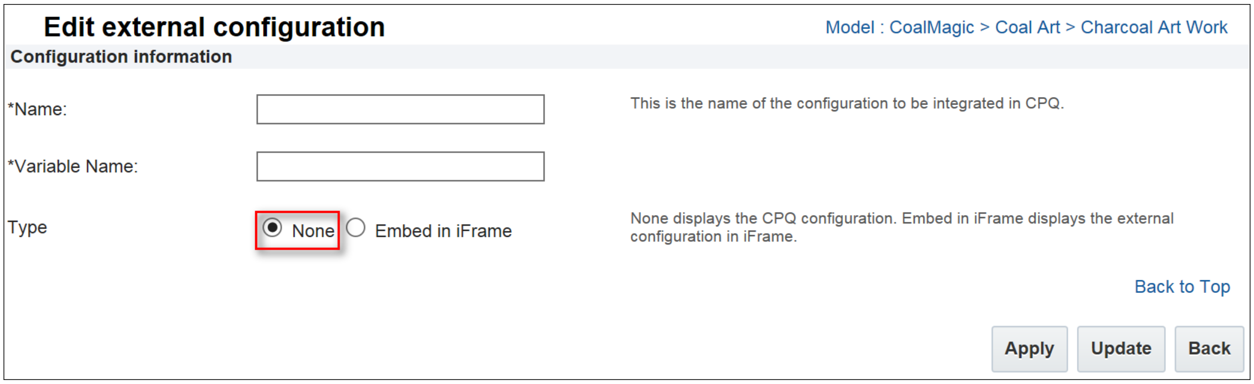
Model Administration List

Product Line : CoalMagic > Coal Art

Select	Model	Navigation
□	Charcoal Art Work	External Configuration

Figure 19: Model Administration List Page

6. Click **List**.
   The **Edit external configuration** page opens with the **Type** option set by default to **None**. This indicates there are no external configurators defined for the specified model.



Edit external configuration

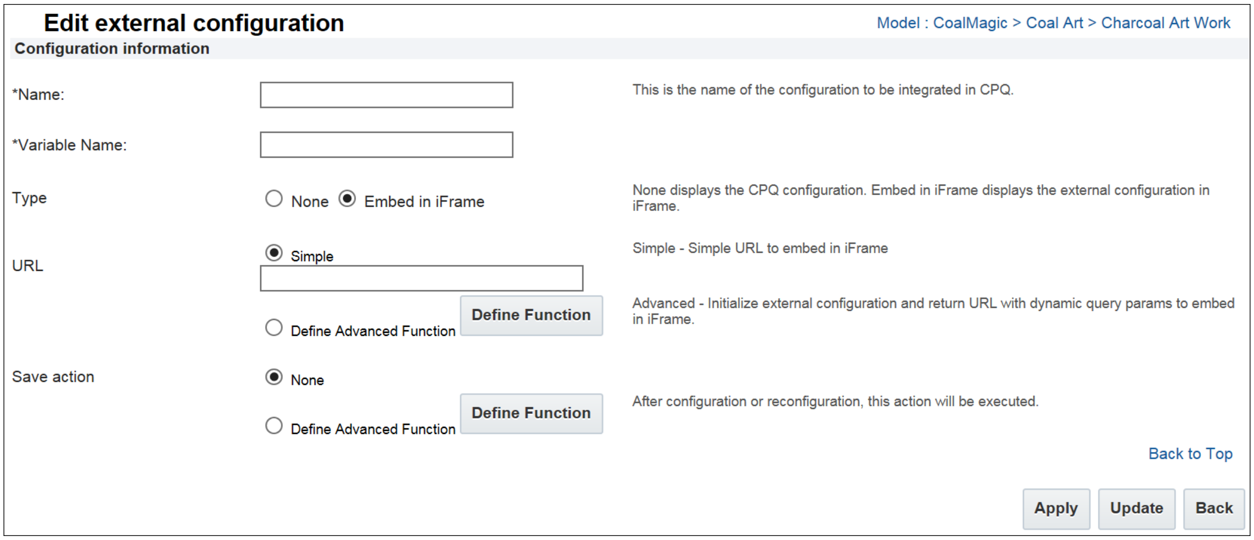
Model : CoalMagic > Coal Art > Charcoal Art Work

Configuration information

*Name:		This is the name of the configuration to be integrated in CPQ.
*Variable Name:		
Type	● None ○ Embed in iFrame	None displays the CPQ configuration. Embed in iFrame displays the external configuration in iFrame.

Figure 20: Edit External Configuration Page with None Option Selected

7. Set the **Type** option to **Embed in iFrame**.
   Additional options are displayed in the **Edit external configuration** page.



Edit external configuration

Model : CoalMagic > Coal Art > Charcoal Art Work

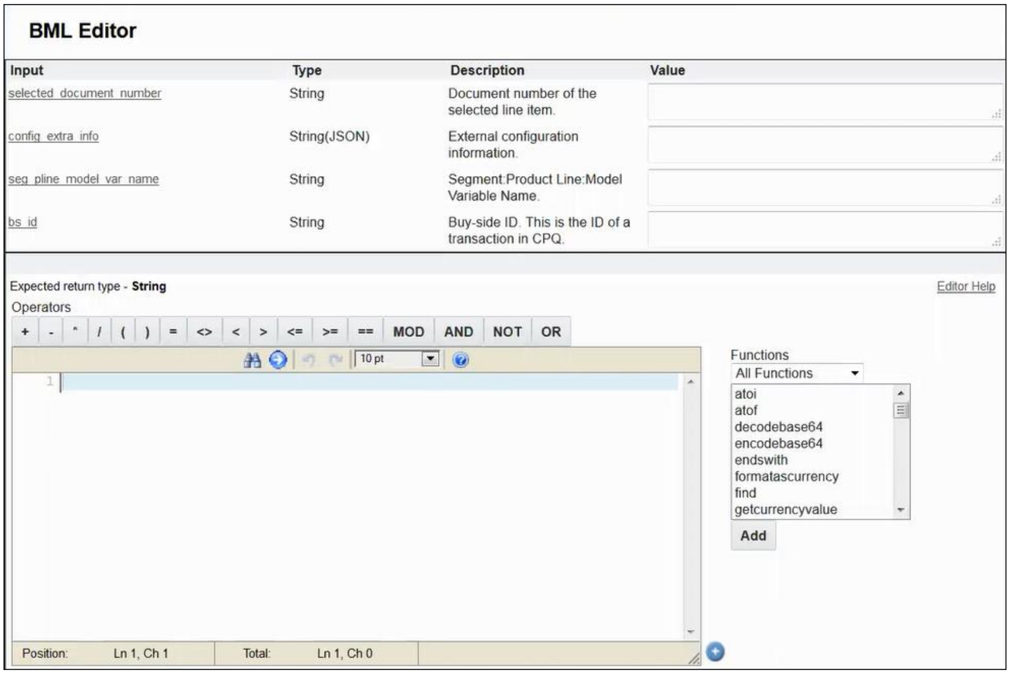
Configuration information

*Name:		This is the name of the configuration to be integrated in CPQ.
*Variable Name:		
Type	○ None ● Embed in iFrame	None displays the CPQ configuration. Embed in iFrame displays the external configuration in iFrame.
URL	● Simple	Simple - Simple URL to embed in iFrame
	○ Define Advanced Function	Advanced - Initialize external configuration and return URL with dynamic query params to embed in iFrame.
Save action	● None	
	○ Define Advanced Function	After configuration or reconfiguration, this action will be executed.

Figure 21: Edit External Configuration Page with Embed in iFrame Option Selected

8. Set the **URL** option to one of the following:

   - **Simple** – Enter the URL for the external configurator, which is viewable via an iFrame to users. If an invalid URL is entered, users see an error message in an iFrame when attempting to access the external configurator.
   - **Define Advanced Function** – Click **Define Function** to open the **BML Editor**. The **BML Editor** is used to define the URL for the external configurator, which is viewable to users via an iFrame.
   - Administrators can use the text box that displays to the left of the list of functions to define the BML code that returns the URL for the iFrame that displays the external configurator. When an error occurs in the BML, the iFrame is not generated.
   - To debug the BML, administrators can specify runtime values of input arguments using the **Value** fields. The runtime values are used to validate the output of the corresponding BML function.



Input	Type	Description	Value
selected_document_number	String	Document number of the selected line item.	
config_extra_info	String(JSON)	External configuration information.	
seg_pline_model_var_name	String	Segment:Product Line:Model Variable Name.	
bs_id	String	Buy-side ID. This is the ID of a transaction in CPQ.	

Expected return type - **String**

Position: Ln 1, Ch 1 Total: Ln 1, Ch 0

Figure 22: BML Editor

NOTE: Administrators must use the **Define Advanced Function** option when using EBS or Fusion as the external configurator.

9. Set the **Save** action to one of the following options:
   - **None** – The save action is not invoked.
   - **Define Advanced Function** – Click **Define Function** to open the **BML Editor**. Using the BML Editor, administrators can set up the BML for the save action invoked when a user configures or reconfigures the product. Use the **Value** fields to specify the runtime values used by the BML code to validate the output of the corresponding BML function.
10. Click **Apply** to enable the integration.

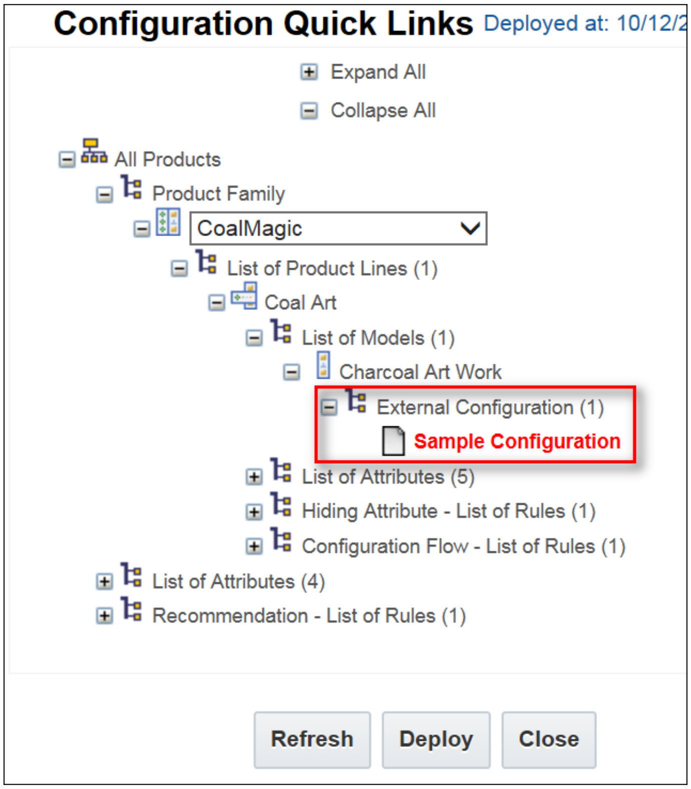
NOTE: Administrators can disable an external configurator integration by returning to the **Edit external configuration** page, setting the **Type** option to **None**, and clicking either **Apply** or **Update**. If the integration the administrator is disabling uses the **Define Advanced Function** for the URL or the **Save** action, the BML is deleted from both the **BML Editor** and the database.

USE CONFIGURATION QUICK LINKS TO ACCESS AN EXTERNAL CONFIGURATOR

After defining an external configurator integration, the external configurator automatically displays within Configuration Quick Links.

To access the external configurator via Configuration Quick Links:

1. From the **Configuration Quick Links** pane, drill-down to the associated product family and model.



Configuration Quick Links Deployed at: 10/12/2

- Expand All
- Collapse All
- All Products
  - Product Family
    - CoalMagic
      - List of Product Lines (1)
        - Coal Art
          - List of Models (1)
            - Charcoal Art Work
              - External Configuration (1)
                - Sample Configuration
          - List of Attributes (5)
          - Hiding Attribute - List of Rules (1)
          - Configuration Flow - List of Rules (1)
  - List of Attributes (4)
  - Recommendation - List of Rules (1)

Figure 23: Configuration Quick Links

NOTE: Any un-deployed changes are displayed in red until the changes are deployed. Administrators can click **Deploy** from the **Configuration Quick Links** page to deploy any un-deployed changes.

2. Click the **External Configuration** link shown in the above figure. The **Edit external configuration** page opens and allows the administrator to view, modify, or disable the external configurator.

USE AN EXTERNAL CONFIGURATOR COMMERCE ATTRIBUTE IN STANDARD BML FUNCTIONS

A new Commerce attribute is available in 2016 R2 and supports the external configurator feature. Administrators can use the attribute in standard BML functions, which include library functions, Commerce functions, and Commerce rules.

Name	Variable Name	Type	Description
`_config_extra_info`	`Config Extra Info`	Text	The attribute is available as part of transaction line attributes. The variable is a field that stores the external configuration information in JSON format.

STEPS TO ENABLE

To enable this feature, follow the steps in the "Define an External Configurator Integration" topic.

TIPS AND CONSIDERATIONS

Consider the following tips when setting up and using an external configurator:

- SSO is required to be set up for both the external configurator and CPQ Cloud. The CPQ Cloud sales specialist must only log in to CPQ Cloud once to use both CPQ Cloud and the external configurator.
- The URL of an EBS or Fusion configurator is rendered in an iFrame by invoking the EBS or Fusion REST Web Service.

KEY RESOURCES

Refer to the following resources for additional information:

- CPQ Cloud Administrator Online Help

OTHER ENHANCEMENTS

In 2016 R2, CPQ Cloud continues our commitment to improving the customer experience by providing a more consistent look and feel throughout CPQ Cloud's Administrative user interface. In 2016 R1, CPQ Cloud provided modern style updates for the Administration Platform. In this release, CPQ Cloud delivers Administration styling enhancements for the Product Pricing and Formula Management sections. Several new font families are also available in CPQ Cloud 2016 R2. These font families expand the options available to customers for creating aesthetically pleasing documents in Document Engine and Document Designer.

The following enhancements are available in CPQ Cloud 2016 R2:

- Administration Styling Enhancements
- New Font Families

ADMINISTRATION STYLING

In 2016 releases, CPQ Cloud adopted a design strategy to improve the customer experience throughout the administrative UI. This design strategy incorporates the following design enhancements:

- A new UI provides a unified, compelling, and comfortable experience across applications deployed to the Cloud and Mobile clients.
- A simpler page construction and less page clutter allow critical content to load faster.
- A consistent design with common design elements provides an intuitive interface. New design enhancements include updates to navigation, buttons, selectors, tabs, and tool tips.

COMMON DESIGN ELEMENTS

ADMINISTRATION NAVIGATION

The administration top navigation bar has a clean and modern appearance for icons, labels, and links.

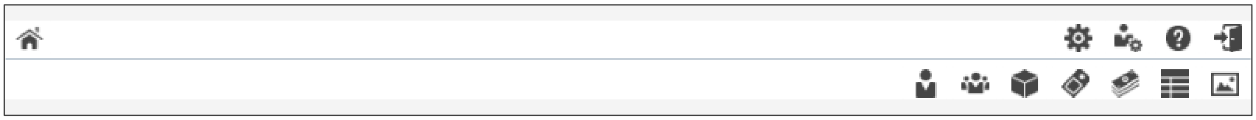
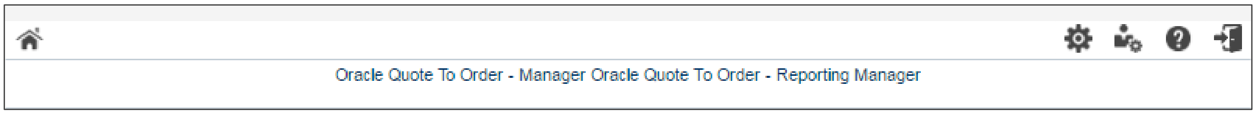


Figure 24: Administration Navigation



Oracle Quote To Order - Manager Oracle Quote To Order - Reporting Manager

Figure 25: Administration Navigation Links

TABS

A consistent presentation is now available for Selected Tabs, Disabled Tabs, Unselected Tabs, and Tabs with Hover.

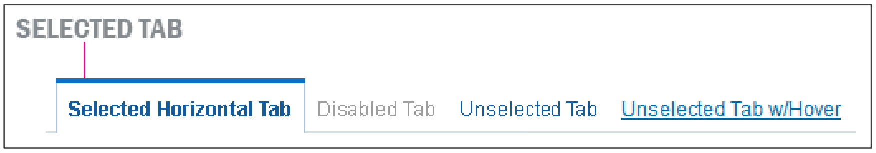


Figure 26: Tab Appearance

BUTTONS

The new design includes consistent button styles and behaviors.

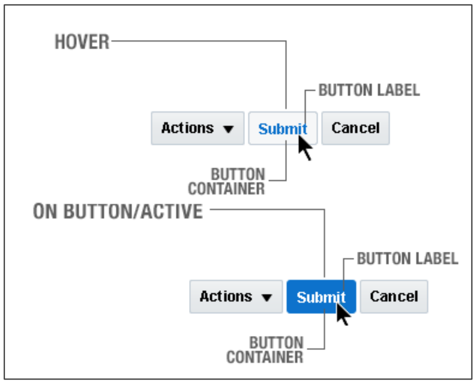
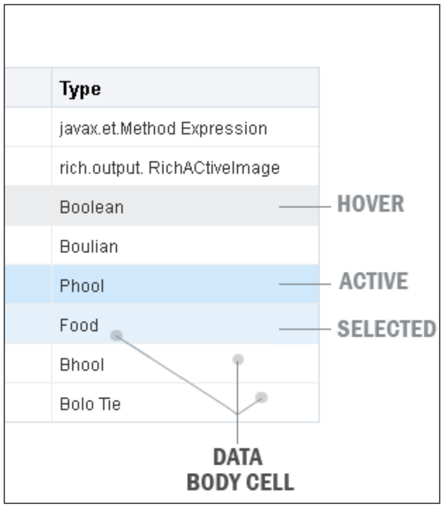


Figure 27: Button Appearance

ITEM SELECTION

Active, Inactive, Hover, and Selected states now have a consistent appearance.

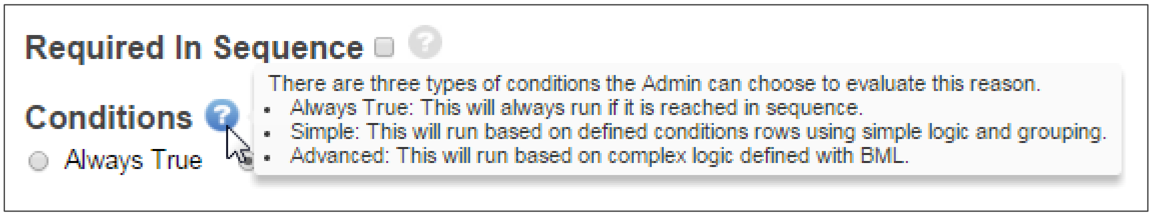


Type
javax.et.Method Expression
rich.output. RichACtiveImage
Boolean
Boulian
Phool
Food
Bhool
Bolo Tie

Figure 28: Item Selection Appearance

TOOL TIPS

Tool tips and notes now have a consistent look-and-feel.



Required In Sequence □

Conditions

○ Always True

There are three types of conditions the Admin can choose to evaluate this reason.

- Always True: This will always run if it is reached in sequence.
- Simple: This will run based on defined conditions rows using simple logic and grouping.
- Advanced: This will run based on complex logic defined with BML.

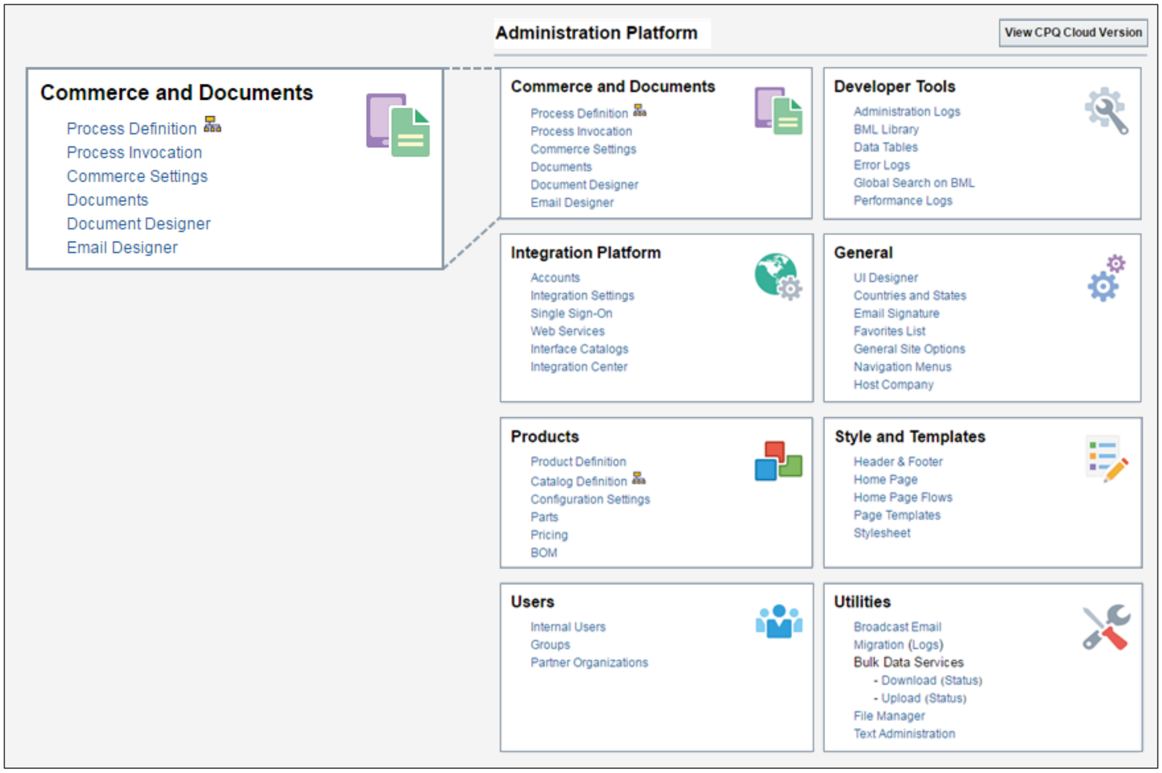
Figure 29: Tool Tips Appearance

NEW ADMINISTRATION PAGE DESIGNS

Administration pages now have a new modern scheme with a consistent style and behavior for buttons, selectors, tabs, notes, and navigation elements. The following images display the updated design for the Administration Platform, Product Home page, Transaction page, and Configurator page.

ADMINISTRATION PLATFORM

The CPQ Cloud **Administration Platform**, referred to as the Admin Home page, is a frequently used destination for CPQ Cloud admins. Each link on the Admin Home page is the gateway to a different administration module. Either directly, or through a navigation path that originates on the Admin Home page, every admin page in CPQ Cloud can be accessed from the Admin Home page.



Administration Platform

Commerce and Documents
- Process Definition
- Process Invocation
- Commerce Settings
- Documents
- Document Designer
- Email Designer

Commerce and Documents
- Process Definition
- Process Invocation
- Commerce Settings
- Documents
- Document Designer
- Email Designer

Developer Tools
- Administration Logs
- BML Library
- Data Tables
- Error Logs
- Global Search on BML
- Performance Logs

Integration Platform
- Accounts
- Integration Settings
- Single Sign-On
- Web Services
- Interface Catalogs
- Integration Center

General
- UI Designer
- Countries and States
- Email Signature
- Favorites List
- General Site Options
- Navigation Menus
- Host Company

Products
- Product Definition
- Catalog Definition
- Configuration Settings
- Parts
- Pricing
- BOM

Style and Templates
- Header & Footer
- Home Page
- Home Page Flows
- Page Templates
- Stylesheet

Users
- Internal Users
- Groups
- Partner Organizations

Utilities
- Broadcast Email
- Migration (Logs)
- Bulk Data Services
  - Download (Status)
  - Upload (Status)
- File Manager
- Text Administration

Figure 30: Administration Platform Appearance

PRODUCT PRICING AND FORMULA MANAGEMENT PAGE ENHANCEMENTS

In addition to the common style elements, the **Product Pricing** and **Formula Management** pages now have a new left-hand panel design. This release also introduces filter improvements and easier access to update profile and rule names.

COLLAPSIBLE NAVIGATION PANES

The left-hand navigation panels now have consistent collapse and expand control buttons.

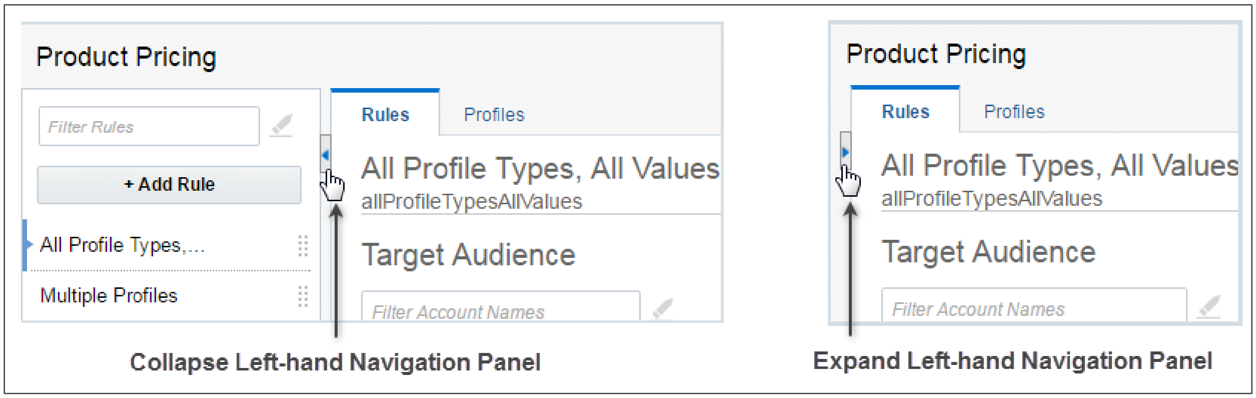


Figure 31: Collapsible Panels

FILTER IMPROVEMENTS

To erase the filter criteria, click the **Clear** icon to the right of the filter box.

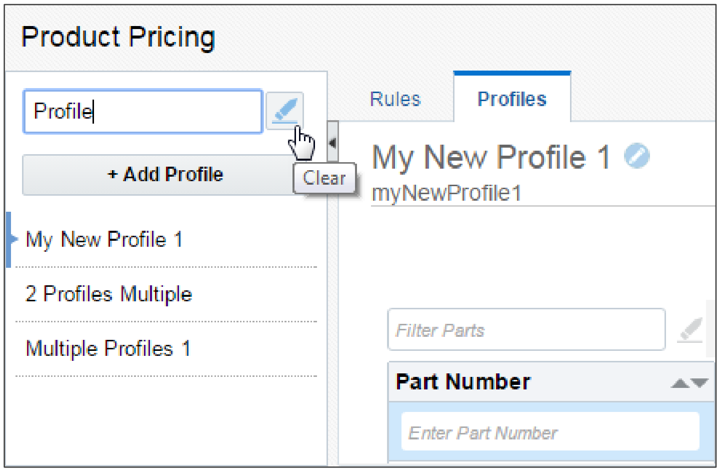


Figure 32: Clear Filter

UPDATE PROFILE OR RULE NAME

To update pricing profile or rule names click the **Edit** icon, enter the new name, and click outside of the edit box to apply changes.

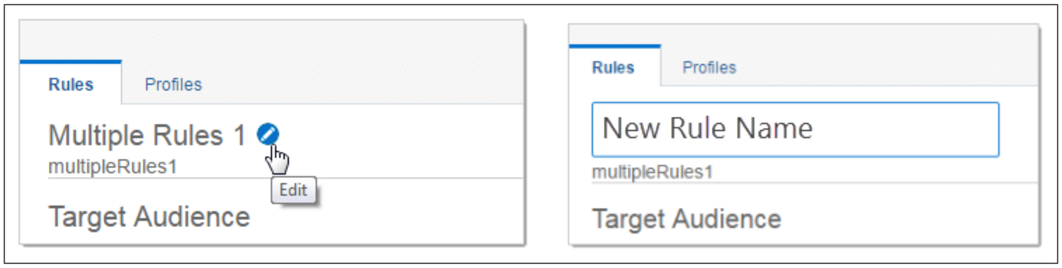


Figure 33: Update Profile Name

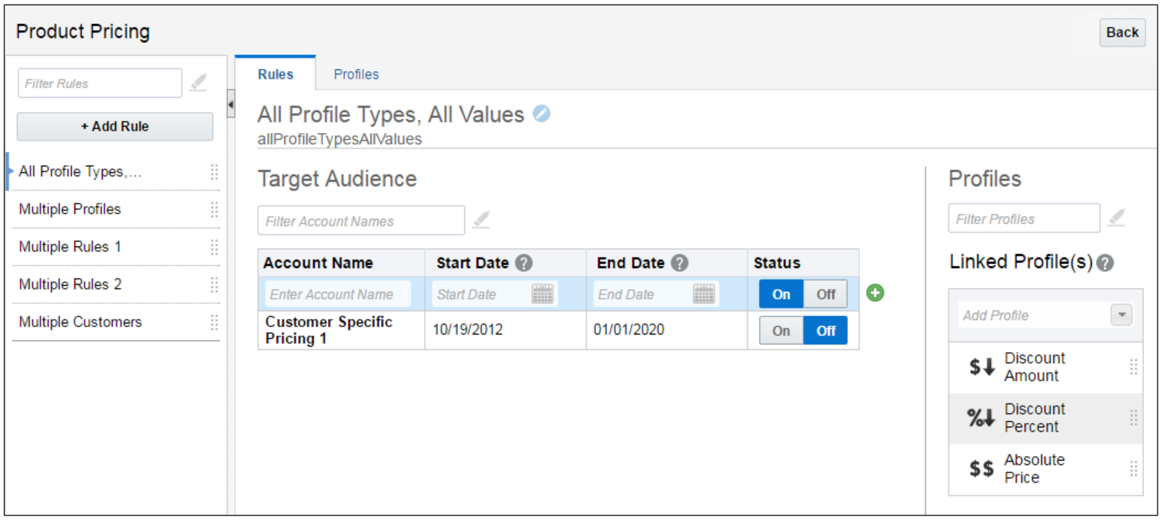
PRODUCT PRICING PAGE

The **Product Pricing** page includes the following design elements:

- Common page layouts, fonts, and backgrounds
- Updated headers with action buttons moved to the top of the page
- Updated styles for left-hand navigation panel
- Edit icons for applicable fields
- Consistent Tool Tip behavior and appearance
- Consistent behavior and appearance for navigation tabs

PRODUCT PRICING – RULES TAB

The **Rules** tab is used by administrators to create rules and add Accounts.

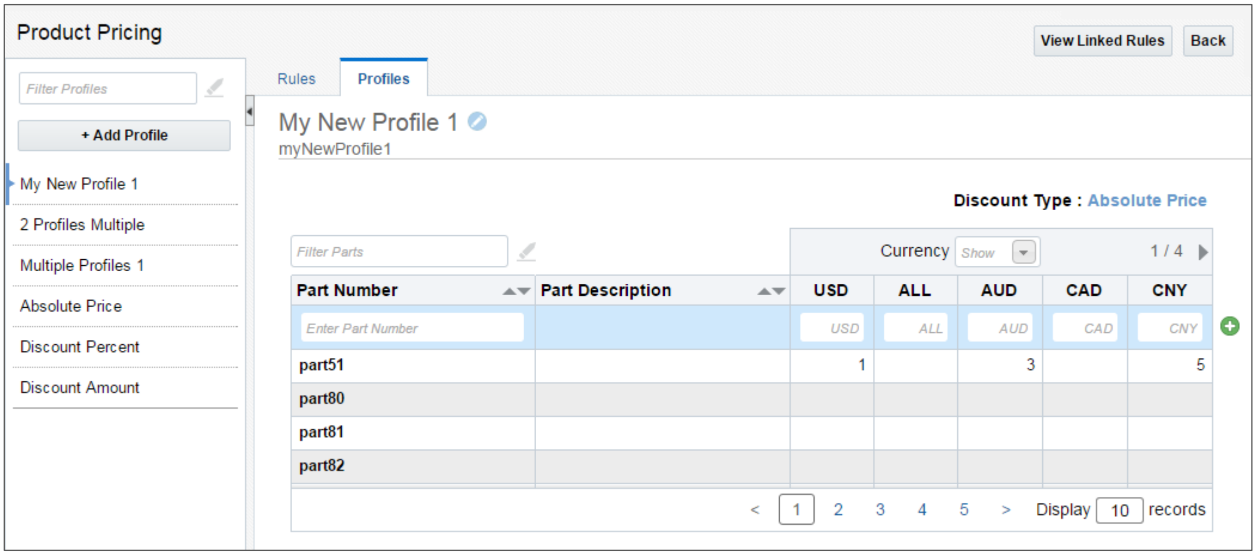


Account Name	Start Date	End Date	Status
Enter Account Name	Start Date	End Date	On Off
Customer Specific Pricing 1	10/19/2012	01/01/2020	On Off

Figure 34: Pricing - Rules Page Appearance

PRODUCT PRICING – PROFILES TAB

The **Profiles** tab is used by administrators to create a Profile and add parts and pricing. This page features an updated pagination menu at the bottom of the parts lists to view long lists of records.



Product Pricing

- My New Profile 1
- 2 Profiles Multiple
- Multiple Profiles 1
- Absolute Price
- Discount Percent
- Discount Amount

My New Profile 1
myNewProfile1

Discount Type : Absolute Price

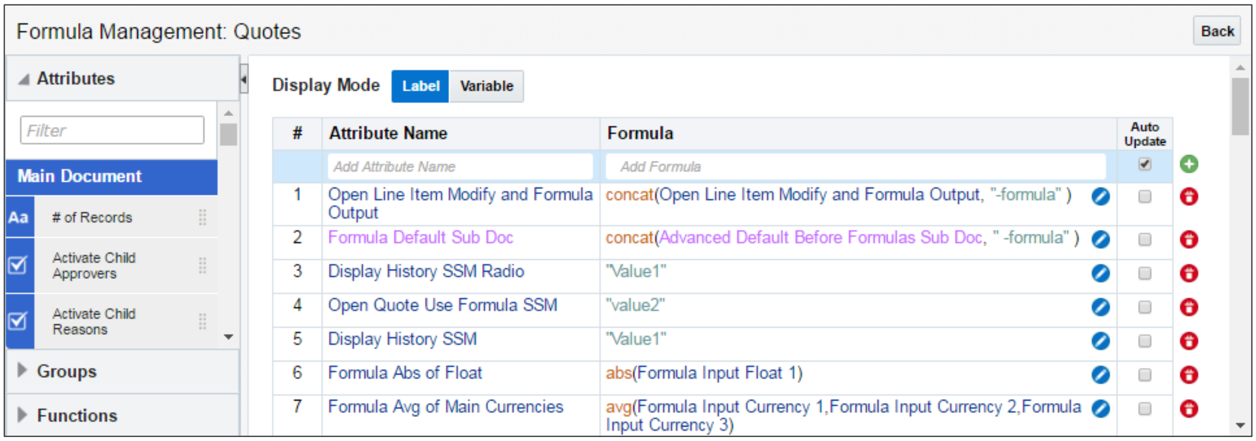
Part Number	Part Description	USD	ALL	AUD	CAD	CNY
Enter Part Number		USD	ALL	AUD	CAD	CNY
part51		1		3		5
part80						
part81						
part82						

Figure 35: Pricing - Profiles Page Appearance

FORMULA MANAGEMENT PAGE

The **Formula Management** page includes the following design elements:

- Common page layouts, fonts, and backgrounds
- Updated headers with action buttons moved to the top of the page
- Updated styles for left-hand navigation panel
- Consistent Tool Tip behavior and appearance



Formula Management: Quotes

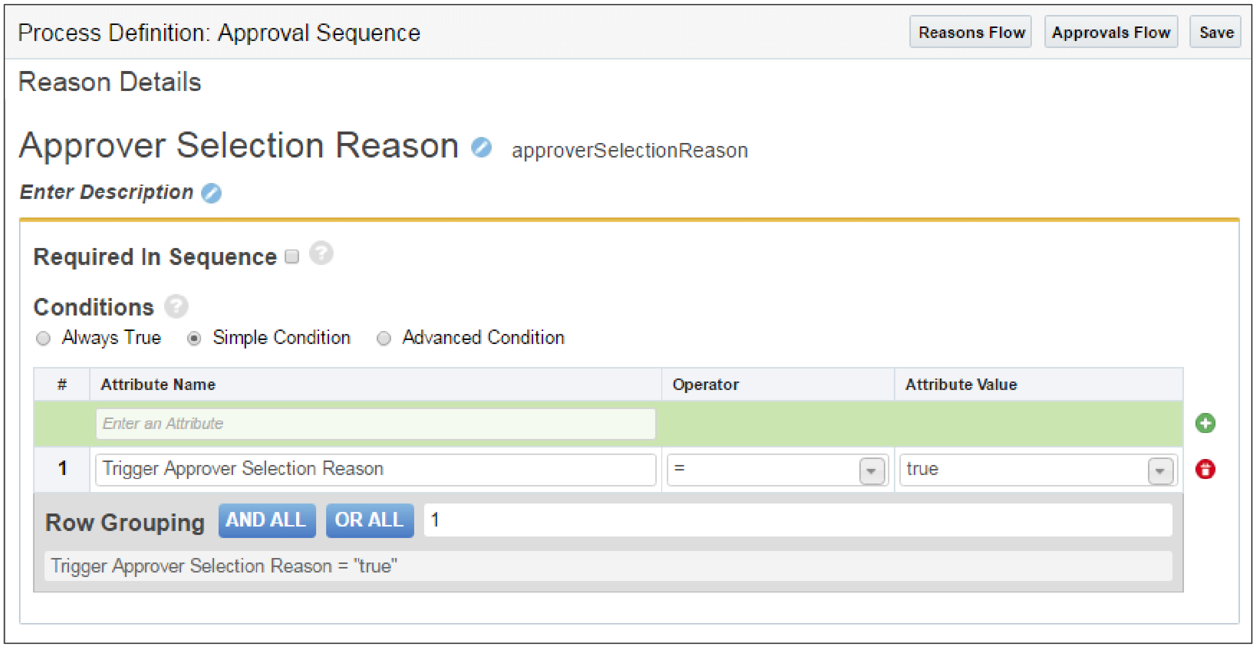
#	Attribute Name	Formula	Auto Update
	Add Attribute Name	Add Formula	☑
1	Open Line Item Modify and Formula Output	concat(Open Line Item Modify and Formula Output, "-formula")	□
2	Formula Default Sub Doc	concat(Advanced Default Before Formulas Sub Doc, " -formula")	□
3	Display History SSM Radio	"Value1"	□
4	Open Quote Use Formula SSM	"value2"	□
5	Display History SSM	"Value1"	□
6	Formula Abs of Float	abs(Formula Input Float 1)	□
7	Formula Avg of Main Currencies	avg(Formula Input Currency 1,Formula Input Currency 2,Formula Input Currency 3)	□

Figure 36: Formula Management Page Appearance

APPROVAL SEQUENCE PAGES

The **Approvals Flow**, **Approval Details**, **Reasons Flow**, and **Reason Details** pages are updated to include the following design elements:

- Common page layouts, fonts, and backgrounds
- Updated headers with action buttons moved to the top of the page
- Edit icons for applicable fields
- Consistent Tool Tip behavior and appearance



Process Definition: Approval Sequence

Reason Details

Approver Selection Reason approverSelectionReason

Enter Description

Required In Sequence □

Conditions

○ Always True ● Simple Condition ○ Advanced Condition

#	Attribute Name	Operator	Attribute Value
	Enter an Attribute		
1	Trigger Approver Selection Reason	=	true

Row Grouping 1

Trigger Approver Selection Reason = "true"

Figure 37: Reason Details Page Appearance

MOBILE LAYOUT EDITORS

The Commerce and Configuration Mobile Layout Editor pages are updated to include the following design elements:

- Common page layouts, fonts, and backgrounds
- Updated headers with action buttons moved to the top of the page
- Updated styles for left-hand navigation panel
- Consistent Tool Tip behavior and appearance
- Consistent behavior and appearance for navigation tabs

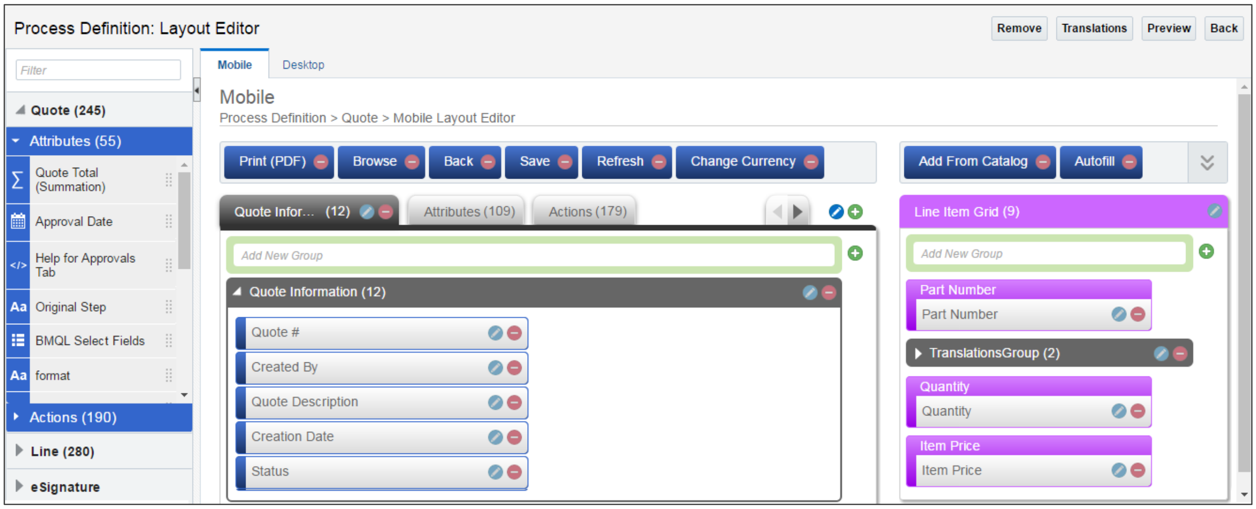


Figure 38: Commerce Mobile Layout Editor Appearance

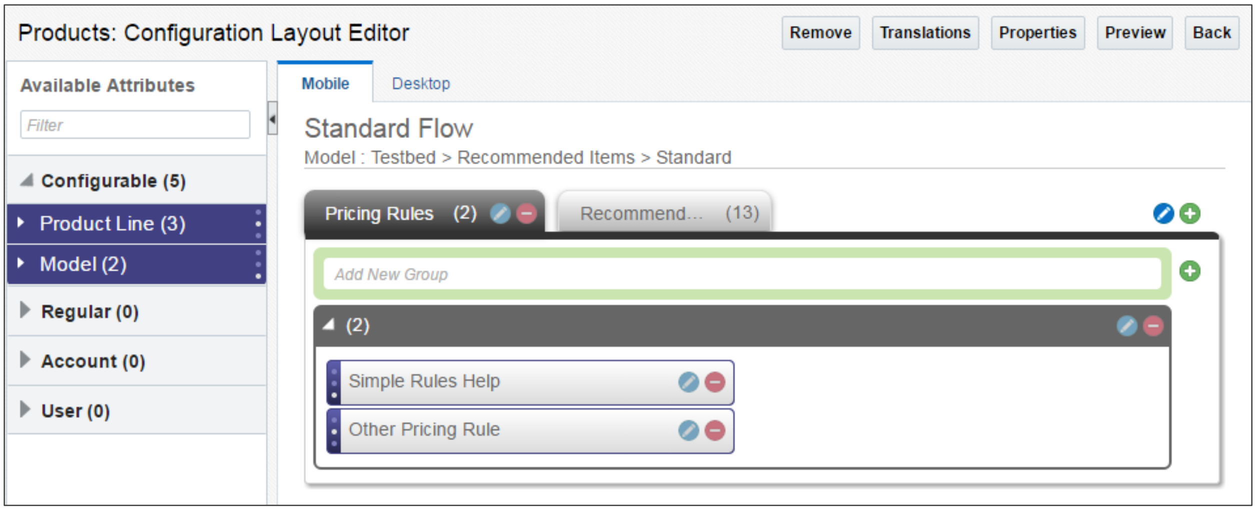


Figure 39: Configuration Mobile Layout Editor Appearance

NEW FONT FAMILIES

Several new font families are available in CPQ Cloud 2016 R2. These font families expand the options available to customers for creating aesthetically pleasing documents in Document Engine and Document Designer.

The following table identifies the font families available in CPQ Cloud 2016 R2.

- Font Family is the name shown in Document Engine and Document Designer.
- Font Type indicates the available styles for each family.
    - Full Family – All styles are available: Normal, **Bold**, *Italic*, ***Bold Italic***.
    - Base Only – Only the Normal style is available.
    - Bold Only – Only the Normal and **Bold** styles are available.
    - Italic Only – Only the Normal and *Italic* styles are available.

Font Family	Font Type
Albany Trans Arabic	Bold Only
Albany WT	Base Only
Albany WT J	Base Only
Albany WT K	Base Only
Albany WT SC	Base Only
Albany WT TC	Base Only
Andale Duospace WT	Bold Only
Andale Duospace WT J	Bold Only
Andale Duospace WT K	Bold Only
Andale Duospace WT SC	Bold Only
Andale Duospace WT TC	Bold Only
Andale Mono	Base Only
Arial	Full Family
Arial Black	Italic Only
Arial Narrow	Full Family
Arial Unicode MS	Base Only
Courier New	Full Family
DeJaVu Sans	Full Family
Franklin Comp ITC Std Black	Italic Only
Franklin Comp ITC Std Light	Italic Only
Franklin Comp ITC Std Med	Full Family
Franklin Comp ITC Std Thin	Italic Only
Franklin Comp ITC Std Ultra	Italic Only
Franklin Cond ITC Std Black	Italic Only
Franklin Cond ITC Std Light	Italic Only
Franklin Cond ITC Std Med	Full Family
Franklin Cond ITC Std Thin	Italic Only
Franklin Cond ITC Std Ultra	Italic Only
Franklin ITC Std Black	Italic Only

Font Family	Font Type
Franklin Nrrw ITC Std Black	Italic Only
Franklin Nrrw ITC Std Light	Italic Only
Franklin Nrrw ITC Std Med	Full Family
Franklin Nrrw ITC Std Thin	Italic Only
Franklin Nrrw ITC Std Ultra	Italic Only
HeiT ASC Medium	Base Only
Helvetica Hebrew	Full Family
Helvetica Inserat LT Pro	Base Only
Helvetica LT Pro	Full Family
Helvetica LT Pro Black	Italic Only
Helvetica LT Pro Black Cond	Italic Only
Helvetica LT Pro Compressed	Base Only
Helvetica LT Pro Condensed	Full Family
Helvetica LT Pro ExtraComp	Base Only
Helvetica LT Pro Light	Italic Only
Helvetica LT Pro Light Cond	Italic Only
Helvetica LT Pro Narrow	Full Family
Helvetica LT Pro UltCompressed	Base Only
Helvetica LT Thai Light	Italic Only
Helvetica LT Thai Regular	Full Family
Helvetica Monospaced Pro	Full Family
Helvetica Textbook Com	Full Family
Helvetica World	Full Family
Helvetica Rounded LT Pro Bd	Italic Only
Helvetica Rounded LT Pro BdCn	Italic Only
Helvetica Rounded LT Pro Blk	Italic Only
MHeiSung HK UltraBold	Base Only
MHeiSung PRC UltraBold	Base Only
MingLiU	Base Only

Font Family	Font Type
Franklin ITC Std Light	Italic Only
Franklin ITC Std Medium	Full Family
Franklin ITC Std Thin	Italic Only
Franklin ITC Std Ultra	Italic Only

Font Family	Font Type
MS Gothic	Base Only
Simplified Arabic	Bold Only
Times New Roman	Full Family
Verdana	Full Family

STEPS TO ENABLE

Administrative styling enhancements and new font families are automatically available in CPQ Cloud 2016 R2.

TIPS AND CONSIDERATIONS

Consider the following tips when using the new font families:

- Since the browser uses web fonts that may not include the font families used in Document Engine or Document Designer, the font families may not render in Document Engine and Document Designer. Previewing or printing the documents with the new font families will display the fonts as intended.
- The process for requesting additional customer-specific fonts for customers with custom font requirements remains the same.

KEY RESOURCES

Refer to the following resources for additional information:

- CPQ Cloud Administrator Online Help

PRE-UPGRADE CONSIDERATIONS

BEHAVIOR CHANGES

- The way the system accommodates storage of user inputs of the ~ character storage in configuration data has changed with this release. Now the '~' character is converted to '$_tilda_$' rather than a comma when the data is saved to a Commerce Transaction. This conversion provides the ability to reconfigure a model with this data but requires any BML scripts relying on the previous conversion of the '~' character to a ',' character to be updated.

  Note: This change is only applicable when a '~' is entered in a configuration field.

- Commerce Processes with a document name equal to 'transaction' will now generate a WSDL that contains complexTypes of ResponseCustomDoctransactionType and CustomDoctransactionType in their generated XSD. This change was made to avoid complexType conflicts when validating the WSDL in certain cases.

SECURE URL ADDRESSES

Confirm that all references to your CPQ Cloud URL, such as in customizations or third party tools, use `https://` in the URL.

TRANSLATION

For some system-defined messages and components, some strings were removed and others added in CPQ Cloud 2016 R2. If you have created your own implementation-specific translations of system-defined strings, some of these strings may no longer appear. Other strings may appear in English. The strings appearing in English are new and require translation.

Most of these messages and components are on the Admin side of CPQ Cloud, but you should review both your end user and administration pages before deploying your updated installation to confirm that all strings appear in the desired language.

MIGRATION

When migrating from one site to another using the Migration Center, both sites must use the same major release. Content may only be migrated across minor releases within the same major release. Migration across major releases cannot occur.

- "Major release" = A major product release, e.g. 2016 R2
- "Minor release" = A release update, e.g. 2015 R2 Update 1

RESOLVED KNOWN ISSUES

For information on bugs fixed in 2016 R2, refer to the 2016 R2 Resolved Known Issues document available on My Oracle Support and the CPQ Cloud Online Help.

TRANSLATION STATUS

CPQ Cloud supports the consumption of both single and multi-byte character sets. Submit a service request on My Oracle Support to enable your site for a new language.

For the following languages, a translation of the CPQ Cloud user interface is available for both the platform and the reference application:

- Chinese (Simplified) [China]
- Chinese (Traditional) [Taiwan]
- Czech [Czech Republic]
- Danish [Denmark]
- Dutch [Netherlands]
- English
- Finnish [Finland]
- French
- French [Canada]
- German
- Hungarian [Hungary]
- Italian
- Japanese [Japan]
- Korean [South Korea]
- Norwegian (Bokmål) [Norway]
- Polish [Poland]
- Portuguese [Brazil]
- Romanian [Romania]
- Russian [Russia]
- Spanish (Worldwide)
- Swedish [Sweden]
- Turkish [Turkey]

POST-UPGRADE CONSIDERATIONS

Upgrade and test all test instances on Oracle CPQ Cloud 2016 R2 before upgrading to production.

BROWSER SUPPORT

CPQ Cloud supports all browser versions that meet the criteria of the Oracle Software Web Browser Support Policy.

SUPPORTED BROWSERS

Windows

- Major releases of Google Chrome upon general browser availability and until Google no longer supports the version
- Major releases of Mozilla Firefox upon general browser availability and until Mozilla no longer supports the version
- Major releases of Internet Explorer/Microsoft Edge within nine months of general browser availability and until Microsoft no longer supports the version

Mac OS X

- Major releases of Google Chrome upon general browser availability and until Google no longer supports the browser version
- Major releases of Mozilla Firefox upon general browser availability and until Mozilla no longer supports the version
- Major releases of Safari within nine months of general browser availability and until Apple no longer supports the version

Android

- Major releases of Google Chrome upon general browser availability and until Google no longer supports the browser version

iOS

- Major releases of Safari within nine months of general browser availability and until Apple no longer supports the browser version.

If you experience issues using a supported browser version, open a ticket on My Oracle Support to resolve the issue. If an issue arises when using a supported browser, use a certified browser version until a fix is delivered. Certified browsers are selected based on current market share and are thoroughly tested to work with the current version's standard functionality.

CERTIFIED BROWSERS

Windows

- Google Chrome 54.x
- Mozilla Firefox 50.x
- Internet Explorer 11.x

iOS

- Operating System: iOS 10.x
- Browser: Safari 10.x
- Screen resolution: 2048 x 1536

NOTE: Compatibility issues with the selected browsers may exist when sites contain additional JavaScript, alternate CSS, or other custom functionality. Customizations may require add-on work. Contact My Oracle Support to determine the availability of workarounds and minor fixes.

SALESFORCE MANAGED PACKAGE SUPPORT

CPQ Cloud no longer releases updates to the Salesforce Managed Packages prior to v7.0. With the release of 2016 R2, only Managed Package v7.x is officially supported. Although Salesforce integrations that use a Managed Package prior to v7.0 are still expected to function, new issues that arise in these versions are not addressed by CPQ Cloud.

TRAINING

Please refer to the release documentation for all versions between your current version and the version to which you are upgrading to see all new functionality, resolved known issues, and functional known issues.

Refer to the CPQ Cloud Online Help to become familiar with the new features introduced in Oracle CPQ Cloud 2016 R2. For additional help, see My Oracle Support.

Verify any information not explicitly mentioned in this document as supported by the software against the product help for Oracle CPQ Cloud 2016 R2 or the Oracle CPQ Cloud Consulting team.

ADDITIONAL INFORMATION

For more information on Oracle CPQ Cloud, visit the Oracle CPQ Cloud documentation site.

Integrated Cloud Applications & Platform Services