



ORACLE

Oracle CPQ and Oracle Enterprise Contracts Integration Guide

Release 18A and Later

Feb 2022
Copyright © 2022, Oracle and/or its affiliates

TABLE OF CONTENTS

Table of Contents.....	2
Revision History	3
Introduction.....	3
Overview	4
Establish Enterprise Contracts Connection	5
Create Commerce Attributes	6
File Attachment Attribute(s)	6
Attributes for Enterprise Contracts Parameters	8
Optional Attributes.....	9
Create Commerce Contract Management Integrations	10
Contract Integration.....	10
Preview Contract Integration	12
Create Commerce Actions.....	15
Add Contracts Enterprise Components to Transaction UI:.....	16
REST APIs to Initiate Contract Modify Actions.....	17
Commerce REST API.....	17
Appendix A: Create Contract XSL File Samples	18
Sample Create Contract SOAP Generator XSL File.....	18
Sample Create Contract Result Parser XSL File	19
Appendix B: Preview Contract XSL File Samples	20
Sample Preview Contract SOAP Generator XSL File	20
Sample Preview Contract Result Parser XSL File	20

REVISION HISTORY

This document will continue to evolve as existing sections change and new information is added. All updates appear in the following table:

DATE	WHAT'S CHANGED	NOTES
FEB 2022		
DEC 2020	Product name updates	Oracle Branding Compliance
FEB 2018		Initial Document Creation

INTRODUCTION

Oracle CPQ currently allows customers to perform certain contract management activities such as contract generation, negotiation of terms, and electronic signature for contract documents. Beginning in Oracle CPQ 18A, customers can integrate with Oracle Enterprise Contracts to provide a comprehensive Contract Lifecycle Management solution. Contracts Enterprise has extensive functionality that supports the legal definition, negotiation redlining, execution, and ongoing management of contracts.

The Oracle CPQ - Contracts Enterprise integration allows customers to create and manage legal aspects of a transaction and maintain the contract throughout its lifecycle, long after the initial transaction that created the contract is completed. The Oracle CPQ - Contracts Enterprise integration provides the ability to create and preview the contracts generated in Contracts Enterprise seamlessly within the Oracle CPQ application via web service calls.

Purpose

The purpose of this implementation guide is to provide the steps that administrators must complete in Oracle CPQ to prepare for an Oracle Enterprise Contracts and Oracle CPQ integration.

Audience

This implementation guide is for administrators who are setting up and configuring the integration. The guide assumes administrators have prior Oracle CPQ administration experience.

Prerequisites

The following is a list of integration prerequisites:

- Oracle Enterprise Contracts is an independently licensed product. Please contact your Oracle Sales Representative to learn more about acquiring an Oracle Enterprise Contracts license.
- An Oracle Enterprise Contracts site setup with Release 13 18.02 and above.
- An Oracle CPQ 18A or later Base Ref App site set up as described in this implementation guide. The integration between Contracts Enterprise and Oracle CPQ adds attributes to the Base Ref App site that correspond to required Commerce Cloud order data.

OVERVIEW

The integration allows customers to create and preview the contracts generated in Oracle Enterprise Contracts. XSL files are used during the integration to exchange data. A total of four XSL files are required for Contract Management: two for creating contracts, and two for previewing contracts. Two XSL files are uploaded to each integration: an Input SOAP Generator XSL file and a Result Parser XSL file.

Administrators perform the following tasks to set up the integration:

1. [Establish Enterprise Contracts Connection](#)

Create a Contracts Management integration in the Oracle CPQ Integration Center to establish a connection with Enterprise Contracts.

2. [Create Commerce Attributes](#)

Create the following Commerce attributes for the integration.

- **File Attachment Attributes** to store the contract document. These attributes are accessed to preview contracts.

Note: If customers want to preview the contract in multiple formats (e.g. RTF, HTML, etc.), File Attachment attributes should be created for each desired format.

- **Attributes for Enterprise Contracts Parameters** to store contract IDs, contract types, and organization IDs.
- **Additional Attributes** can be used for other Enterprise Contracts parameters or to identify if actions have associated integrations.

3. [Create Commerce Contract Management Integrations](#)

Create at least two integrations to support Contract Management.

- **Create Contract Integration:** This integration sends transaction and customer information to Enterprise Contracts. Contracts Enterprise returns contract information for the pending contract.
- **Preview Contract Integration:** This integration sends a request to Contracts Enterprise for the contract document. Contracts Enterprise returns the contract document in the desired format.

Note: If customers want to preview the contract in multiple formats, Preview Contract integrations need to be created for each desired format.

4. [Create Commerce Actions](#)

Administrators can create a single action to initiate Create and Preview Contract integrations or two actions, one for the Create Contract integration and a second for the Preview Contract integration.

Note: If customers want to preview the contract in multiple, modify actions need to be created for each desired format. Adding more than one Preview Contract integration on the same action is not supported. So, if the contract is to be previewed in multiple formats, then an additional action for each corresponding integration has to be created for each format.

5. [Add Contract Components to Transaction UI](#)

Administrators need to add required contract components to the Transaction UI.

ESTABLISH ENTERPRISE CONTRACTS CONNECTION

A connection between Oracle CPQ and Contracts Enterprise must be established to use Oracle Enterprise Contracts. To establish this connection, administrators create a Contract Management integration in the Oracle CPQ Integration Center. The Integration Center is a page for managing third party integrations with Oracle CPQ.

Create Contracts Management Integration

Complete the following steps to create a Contracts Management Integration:

1. Log in to Oracle CPQ and open the Admin Home page.
2. Click **Integration Center**, in the Integration Platform section.
3. Click Create Integration.
4. Select **Contract Management** from the Type drop-down.

The screenshot shows the 'Integration Center' interface. On the left is a sidebar menu with options: eSignature, Integration Cloud Service, Remote Approval, DataCube Integration, Contract Management (highlighted), Generic Integration, Authentication Certificate, Digital Assistant, and Data Table Export Sched... At the top right are buttons for 'Back', 'Create Integration', and 'Save'. The main form area has a 'Type' dropdown set to 'Contract Management'. Below this are input fields for: *Name (Contract Management), Request URL (https://fuscdrmsmc47-fa-ext.us.oracle.com/fscmService/), Revise URL (https://fuscdrmsmc47-fa-ext.us.oracle.com/fscmService/), Username (contractMgr), and Password (masked with ****). There is a 'Verify Connection' section with a 'Test' button. At the bottom, there is an 'Enable Integration' checkbox which is checked, and a warning icon with text: 'Disabling this integration will update integrations currently using this service.'

5. Enter a **Name** - Use a unique name to identify the integration. (e.g. Enterprise Contracts)
6. Enter the **Request URL** - The Contract Management SOAP URL used to create a contract.
7. Enter the **Revise URL** - The Contract Management SOAP URL used to preview a contract.

Note: For the Oracle Enterprise Contracts integration, the Request and Revise URLs are the same. (e.g. <https://fuscdrmsmc47-fa-ext.us.oracle.com/fscmService/ContractServiceV2?WSDL>)

8. Enter the **Username** and **Password**
Provide Contract Management application user credentials used to create and view contracts.
9. Click **Test** to verify the connection and credentials.
10. Verify the connection is successful.

Note: "Test Connection Passed" will appear next to the Test button for successful connections.

11. Click the **Enable Integration** checkbox.

Note: Uncheck the checkbox to disable Contracts integration functionality.

12. Click on **Save**.

CREATE COMMERCE ATTRIBUTES

File Attachment Attribute(s)

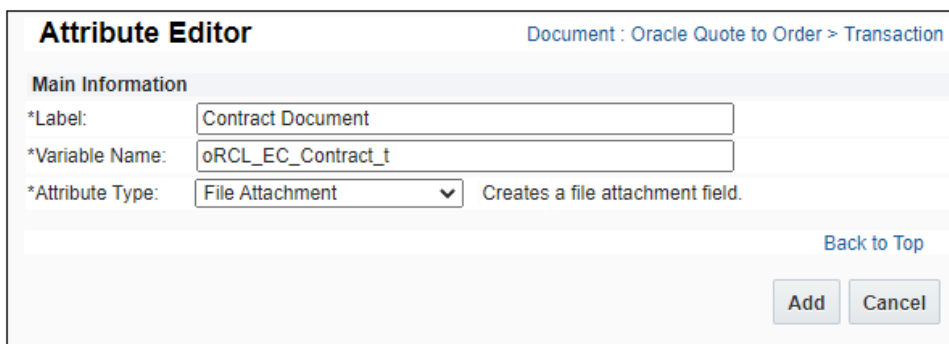
In Enterprise Contracts integrations, a Commerce File Attachment attribute stores the contract document that is previewed using the Preview Contract action. If a File Attachment attribute is not defined, the Contract document is not saved and is only displayed in a new tab or window for preview.

Oracle Enterprise Contracts supports retrieving the Contract document in multiple formats. If multiple Contract document formats are desired, administrators must create a File Attachment attribute for each format. For example, if the customer requests PDF and HTML contracts, two File Attachment attributes must be created: one File Attachment attribute must be created for the PDF document, and a second File Attachment attribute must be created for the HTML document.

Create File Attachment Attribute

Complete the following steps to create a Commerce File Attachment attribute:

1. Log in to Oracle CPQ and open the Admin Home page.
2. Select **Process Definition** in the Commerce and Documents section.
3. Select **Documents** from the Navigation menu, for the applicable process, and then click **List**.
4. Select **Attributes** from the Navigation menu for the main document (i.e. Transaction), and then click **List**.
5. Click **Add**.



The screenshot shows the 'Attribute Editor' interface. At the top right, it says 'Document : Oracle Quote to Order > Transaction'. Below this is a 'Main Information' section with three fields: '*Label:' with the value 'Contract Document', '*Variable Name:' with the value 'oRCL_EC_Contract_t', and '*Attribute Type:' with a dropdown menu set to 'File Attachment'. To the right of the dropdown is the text 'Creates a file attachment field.'. At the bottom right of the form are buttons for 'Add' and 'Cancel', and a 'Back to Top' link.

6. Enter the **Label** (e.g Contract Document)
7. For the **Variable Name**, enter oRCL_EC_Contract_t

Note:

- "oRCL_EC_Contract_t" is the default variable name for the File Attachment attribute referenced in the integration XSLs. If you use another variable name or you want to include additional File Attachment attributes you must be update the integration XSLs accordingly.
- The Variable Name field populates automatically. Variable names can only contain alpha-numeric characters and underscores. You can change the entry before saving. After saving, the value is read-only.

8. Select **File Attachment** from the Attribute Type menu.
9. Click **Add**, the File Attachment Attribute Editor opens.

File Attachment Attribute Editor (oRCL_EC_Contract_t) Document : Oracle Quote to Order > Transaction

General **Document Views** **Mapping**

Main Information

*Label: Contract Document

Variable Name: oRCL_EC_Contract_t

Required:

Description: [Edit](#)

Exclude From XML:

Exclude from PDF Merge:

JET Layout Path: Unassigned [Save and Edit JET Responsive Layout](#)

Mobile Layout Path: Unassigned [Save and Edit Mobile Layout](#)

Validation Fields

Allowed Extensions:

- bmp
- csv
- dat
- doc

[Back to Top](#)

[Translations](#) [Apply](#) [Update](#) [Update and New](#) [Back](#)

10. If necessary, define the following attribute options:

- **Required:** If this option is selected, the user will be required to enter a value before proceeding.
- **Exclude from XML:** This selection determines if an attribute is available to include in Email Designer and Document Designer documents. The attribute is available when this option is unselected.
- **Exclude from PDF Merge:** This checkbox determines if the file uploaded to this attribute can be merged with the PDF proposal.
- **Allowed Extensions:** This menu selection specifies that only selected extensions will be supported in the file attachment attribute.

11. Click **Apply** to remain on the current page or **Update** to save changes and return to the previous page.

Note: To provide user access to the saved Contract document, administrators must add the File Attachment attribute to the Transaction Layout UI.

Attributes for Enterprise Contracts Parameters

Create the following Commerce main document attributes that are referenced in the integration XSL files.

Note: These attributes are the default attributes referenced in the integration XSLs. If you modify the variable names or you want to include additional attributes you must be update the integration XSLs accordingly.

ENTERPRISE CONTRACTS INTEGRATION ATTRIBUTES			
ATTRIBUTE			NOTES
LABEL	VARIABLE NAME	TYPE	
Contract ID	oRCL_EC_ContractId_t	Text	
Contract Number	oRCL_EC_ContractNumber_t	Text	
Contract Type ID	oRCL_EC_ContractTypeId_t	Text	
Contract Type Name	oRCL_EC_ContractTypeName_t	Text	
Primary Organization ID	oRCL_OSC_PrimaryOrgID_t	Text	The primary organization in OSC associated with this transaction or owner. This attribute is included with the Oracle Sales Reference Application.

Create Commerce Text Attributes

Complete the following steps to create Commerce attributes for Enterprise Contract parameters. This process is repeated for each attribute in the Enterprise Contracts Integration Attributes table.

1. Log in to Oracle CPQ and open the Admin Home page.
2. Select **Process Definition** in the Commerce and Documents section.
3. Select **Documents** from the Navigation menu, for the applicable process, and then click **List**.
4. Select **Attributes** from the Navigation menu for the main document (i.e. Transaction), and then click **List**
5. Click **Add**.
6. Enter the **Label** (e.g. Contract ID)
7. For the **Variable Name**, enter oRCL_EC_ContractID_t

Notes:

- "oRCL_EC_ContractID_t" is the default variable name for the Contract ID attribute referenced in the integration XSLs. If you use another variable name you must be update the integration XSLs accordingly.
- The Variable Name field populates automatically. Variable names can only contain alpha-numeric characters and underscores. You can change the entry before saving. After saving, the value is read-only.

8. Select **Text** (or Integer, if applicable) from the Attribute Type menu.

For information about other attribute types, refer to Oracle CPQ Administration Online Help – [Commerce Document Attributes](#).

9. Click **Add**, the Attribute Editor opens.

Text Attribute Editor

(oRCL_EC_ContractID_t)

Document : Oracle Quote to Order > Transaction

General
Default
Modify
Document Views
Mapping

Main Information

*Label:

Variable Name: oRCL_EC_ContractID_t

Required:

Description: Edit

Exclude From XML:

JET Layout Path: Unassigned Save and Edit JET Responsive Layout

Mobile Layout Path: Unassigned Save and Edit Mobile Layout

Validation Fields

Maximum Field Length:

[Back to Top](#)

Translations
Apply
Update
Update and New
Back

10. If necessary, define the following attribute options:

- **Required:** If this option is selected, the user will be required to enter a value before proceeding.
- **Exclude from XML:** This selection determines if an attribute is available to include in Email Designer and Document Designer documents. The attribute is available when this option is unselected.

11. Click **Apply** to remain on the current page or **Update** to save changes and return to the previous page.

12. Repeat this procedure for the remaining attributes in the Enterprise Contracts Integration Attributes table.

Note: If you use variable names other than the default values, you must update the integration XSLs.

Optional Attributes

- Optional, administrators can create additional Commerce attributes for other Enterprise Contracts parameters, such as Contract creation date, contract expiration date, etc.

Note: If additional attributes are created, they also need to be added to the integration XSLs.

- Additionally, when customers need to identify if actions have associated integrations based upon the REST API response, the administrator must create Commerce attributes and populate them from the Preview Contract Result Parser XSL results.

For example: In the Preview Contract Integration the [Sample Preview Contract Result Parser XSL File](#), the "isActionOfTypePreviewContract" and "contractDocumentAttributeName" attributes were populated.

When additional Commerce attributes are set up and defined, the corresponding attribute values will be returned in the REST API response.

Refer to the following resources to create additional attributes:

- [Create Commerce Text Attributes Procedure](#)
- Oracle CPQ Administration Online Help – [Commerce Document Attributes](#).

CREATE COMMERCE CONTRACT MANAGEMENT INTEGRATIONS

The following table describes different set up options to define how the Contracts documents are stored and displayed in Oracle CPQ.

INTEGRATION "PREVIEW CONTRACT"	FILE ATTACHMENT ATTRIBUTE	BEHAVIOR
Enabled	Defined	The Contract document is stored in the specified File Attachment attribute and is immediately displayed in a new tab or window for preview.
Not Enabled	Defined	The Contract document is stored in the specified File Attachment attribute, but not previewed immediately. If the File Attachment attribute is added to the layout, the user can click the File Attachment attribute link to preview the Contract document. Note: When using the JET Transaction UI, the Integration "Preview Contract" must be disabled and the File Attachment attribute must be defined.
Enabled	Not Defined	The Contract document is not stored in a File Attachment attribute and is only displayed in a new tab or window for preview. Note: The JET Transaction UI does not support "Preview Contract" functionality.

Contract Integration

When a transaction is created in Oracle CPQ, users can request contract information before the actual contract is generated from the transaction. Transaction and customer information is sent to Contracts Enterprise via a web service request. The Contracts Enterprise Expert Service module will generate the associated terms and conditions, based on the rules created and executed within Contracts Enterprise and the information from the Oracle CPQ transaction. Contracts Enterprise will return contract attributes (e.g. Contract ID) to Oracle CPQ. The contract attributes are then saved and displayed on the Transaction UI.

Create Contract Integration

Complete the following steps to set up the Create Contract integration:

1. Log in to Oracle CPQ and open the Admin Home page.
2. Click **Process Definition** in the Commerce and Documents section.
3. Select **Integrations** from the Navigation menu, for the applicable process, and then click **List**.
4. Click **Add**, the Select Integration Types page opens.

Note: After an Oracle CPQ - Contracts Enterprise integration is established, the Contract Management type integration is available across all existing and newly created Commerce processes on the site.

5. Select **Contract Management**, and then click **Next**. The Edit Integration page opens.

Edit Integration Process : Quotes

Integration Information

*Name:	<input type="text" value="Create Contract"/>	
*Variable Name:	<input type="text" value="createContract"/>	
Description:	<input style="width: 100%;" type="text" value="This integration is used to upload XSL files for creating a contract"/>	
Timeout:	<input style="width: 50%;" type="text"/>	Defined in milliseconds
Preview Contract::	<input type="checkbox"/>	
File Attachment Attribute:	<input style="width: 60%;" type="text"/>	File Attachment Attribute should be in the form of <document_var_name>.<attr_var_name>
*Integration Type:	Contract Management	
*SOAP Generator XSL:	<input type="button" value="Choose File"/> No file chosen	
*Result Parser XSL:	<input type="button" value="Choose File"/> No file chosen	

[Back to Top](#)

- Enter a Label and Variable Name.
Use a unique name to identify the integration (e.g. Create Contract).

Note: The Variable Name field populates automatically. Variable names can only contain alpha-numeric characters and underscores. The Variable Name can be modified before saving. After saving the value is read-only.

- Enter a **Description** to describe the integration.
- Enter a **Timeout** value.

Enter a value between 0 and 600,000 milliseconds. This value is used when calling the Contract SOAP services. If the Contracts service does not respond within the specified time, the CPQ application will abort the web service call and invoke an error.

- Uncheck the **Preview Contract** checkbox.
This option is not used for Create Contract integrations.
- The **File Attachment Attribute** is not used for Create Contract integrations
- Click the SOAP Generator XSL **Choose File** button to select an XSL file to upload for the SOAP Generator input.

Note: XSL files should be validated prior to upload. File format and syntax are not validated when XSL files are uploaded. Files that contain format or syntax errors will not function correctly when activated.

To view a sample XSL file, refer to Appendix A: [Sample Create Contract SOAP Generator XSL File](#). This sample file is used to create the SOAP Generator XSL input for the "Create Contract" web service.

- This example XSL creates a "Buy:Project Lines" type Contract. If required, this XSL should be modified accordingly to pass different sub-document attributes to Contracts Enterprise.
- In this example XSL, sub-document attributes (line level) are passed to create corresponding lines in a contract.

Create Contract SOAP Generator XSL variables:

- \$bmTransId** - This is the unique ID for the CPQ buyside transaction. It can be passed to contracts service to associate a contract with a particular CPQ transaction.
- \$mainDoc** - This variable contains the information for the Commerce main document. It is used to pass main document Commerce attribute values to the Contracts application.
- \$sub_doc** - This variable contains the information for the Commerce sub-document. It is used to pass sub-document Commerce attribute values to the Contracts application.

12. Click the Result Parser XSL **Choose File** button to select an XSL file to upload for the Result Parser output.

Note: XSL files should be validated prior to upload. File format and syntax are not validated when XSL files are uploaded. Files that contain format or syntax errors will not function correctly when activated.

To view a sample XSL file, refer to Appendix A: [Sample Create Contract Result Parser XSL File](#). This sample file is used to process the Result Parser XSL output for the "Create Contract" web service.

In this example, the value of "ns1:ID", which is present in the response from Contract service, is mapped to and stored in the "oRCL_EC_ContractId_t" Commerce attribute.

Notes:

- Administrators must create the "oRCL_EC_ContractId_t" Commerce attribute prior to invoking the Create Contract web service. Similarly, administrators can edit or create new XSLs to map the output from the Create Contract service to Commerce attributes.
- If there are errors in this XSL, there will be corresponding run time errors.

13. Click **Add**.

Preview Contract Integration

Depending on the customer requirements, the Preview Contract Integration can be invoked using one of the following actions: a modify action designated to preview contracts, or the modify action designated to create and preview contracts.

Note: When the Preview Contract Integration is invoked during contract creation, the Preview Contract integration must be initiated after the Create Contract Integration.

When Preview Contract is initiated, a request for the contract document is sent to Contracts Enterprise. Contracts Enterprise sends the contract document back to Oracle CPQ in the desired format.

- When the Preview Contract option for the Preview Contract Integration is enabled and the user's browser is capable of displaying files, a new tab or window will open to display the contract document. If the user's browser does not display files, the user will be prompted to download the contract document.
- The contract document is also saved to a designated File Attachment Attribute when the attribute is defined in the Preview Contract Integration.

Create Preview Contract Integration

Complete the following steps to set up the Commerce Preview Contract integration:

1. Log in to Oracle CPQ and open the Admin Home page.
2. Click **Process Definition** in the Commerce and Documents section.
3. Select **Integrations** from the Navigation menu, for the applicable process, and then click **List**.
4. Click **Add**, the Select Integration Types page opens.

Note: After an Oracle CPQ - Contracts Enterprise integration is established, the Contract Management type integration is available across all existing and newly created Commerce processes on the site.

5. Select **Contract Management**, and then click **Next**.
The Edit Integration page opens.

Edit Integration
Process : Quotes

Integration Information

*Name:

*Variable Name:

Description:

Timeout: Defined in milliseconds

Preview Contract:

File Attachment Attribute: File Attachment Attribute should be in the form of <document_var_name>.<attr_var_name>

*Integration Type:

*SOAP Generator XSL: No file chosen

*Result Parser XSL: No file chosen

[Back to Top](#)

6. Enter a **Name** and **Variable Name**.

Use a unique name to identify the integration (e.g. Preview Contract).

The Variable Name field populates automatically. Variable names can only contain alpha-numeric characters and underscores. The Variable Name can be modified before saving. After saving, the value is read-only.

7. Enter a **Description** to describe the integration.

8. Enter a **Timeout** value. Enter a value between 0 and 600,000 milliseconds.

This value is used when calling the Contract SOAP services. If the Contracts service does not respond within the specified time, the CPQ application will abort the web service call and invoke an error.

9. Check the **Preview Contract** checkbox.

This checkbox is used to indicate Preview Contract integrations. Enabling this option will automatically open the contract when the Preview action is invoked.

Note: The JET Transaction UI does not support "Preview Contract" functionality. This option should not be used when using the JET Transaction Layout

10. Enter the **File Attachment Attribute** value.

This is the file attachment attribute for the previewed contract. The following syntax is used to provide the document and attribute variable name: <document_var_name>.<attribute_var_name>
(e.g. transaction.oRCL_EC_Contract_t)

11. Click the SOAP Generator XSL **Choose File** button to select an XSL file to upload for the SOAP Generator input.

Note: XSL files should be validated prior to upload. File format and syntax are not validated when XSL files are uploaded. Files that contain format or syntax errors will not function correctly when activated.

To view a sample XSL file, refer to Appendix B: [Sample Preview Contract SOAP Generator XSL File](#). This files is used to create the input for the "Preview Contract" web service.

Preview Contract SOAP Generator XSL variables:

- **\$mainDoc** - This variable contains the information for the main Commerce document and is used to pass Commerce attribute values to the Contracts application.

12. Click the Result Parser XSL **Choose File** button to select an XSL file to upload for the Result Parser output.

Note: XSL files should be validated prior to upload. File format and syntax are not validated when XSL files are uploaded. Files that contain format or syntax errors will not function correctly when activated.

To view a sample XSL file, refer to Appendix B: [Sample Preview Contract Result Parser XSL File](#). This file is used to process the output for the "Preview Contract" web service.

Preview Contract Result Parser XSL variables:

- **\$mainDoc** - This variable contains the information for the main Commerce document. It is used to pass Commerce attribute values to the Contracts application.

The following three tags are required in this XSL:

- **attachment_field_value:** The value returned from the contract service will be stored in the attribute specified for the File Attachment attribute field.
- **mime_type:** This should be mapped to the mime type returned from the contract service.
- **file_name:** This should be mapped to the file name returned back by contract service.

Other Commerce document attributes can also be used in this XSL. They can be mapped to values returned back from service or can be populated with hard coded values.

Note: The Data types of Commerce attribute and corresponding field in the result must be same. Errors will occur if these Data types do not match.

CREATE COMMERCE ACTIONS

Commerce Modify actions are used to initiate the Contract SOAP integrations. Administrators create two Modify actions for Enterprise Contract integrations: "createContract" and "previewContract".

Notes:

- Modify actions are the only supported action type for Contract integrations.
- The Modify actions used for Contracts integration must belong to Commerce Main Document.

Create Commerce Modify Action

Complete the following steps to create a Commerce Modify action.

1. Log in to Oracle CPQ and open the Admin Home page.
2. Click **Process Definition** in the Commerce and Documents section.
3. Select **Documents** from the Navigation menu, for the applicable process, and then click **List**.
4. Select **Actions** from the Navigation menu for the Main Document (i.e. Transaction), and then click **List**.
5. Click **Add**.
6. Enter the Label and Variable Name.
Note: The Variable Name field populates automatically. The entry can be changed before saving. After saving, the value is read-only.
7. Select **Modify** from the Action Type menu.
8. Click **Add**.
9. Click the **Integration** tab.
10. Select the Contract integrations to apply to the action from the **Integration List**. Administrators can select and deselect any of the listed integrations.
11. Click the > button to move the Contract integrations to the **Selected Integration** list.

The screenshot shows the 'Admin Action' configuration page for 'createContract'. The page has a breadcrumb trail: 'Document : Oracle Quote to Order > Transaction'. There are five tabs: 'General Administration', 'Modify', 'Destination', 'Integration', and 'Document Views'. The 'Integration' tab is active. It features two lists: 'Integration List' on the left and 'Selected Integration' on the right. The 'Integration List' contains 'Preview Contract (RTF Format)' and 'Preview Contract (HTML Format)'. The 'Selected Integration' list contains 'Apply Modify Functions', 'Create Contract', and 'Preview Contract (PDF Format)'. The 'Create Contract' item is highlighted in blue. There are arrow buttons between the lists to move items. At the bottom right of the 'Selected Integration' list is a 'Back to Top' link. At the bottom of the page are buttons for 'Translations', 'Apply', 'Update', 'Update and New', and 'Back'.

12. Use the ▲ and ▼ buttons to change the order of contract integrations in the **Selected Integration** list.

Notes:

- If a single action is used for both creating and previewing the contract the order of selected "integrations" is important. The Create Contract integration must be listed before the Preview Contracts integration.
- Adding more than one integration for previewing the contract on the same action is not supported. There should only be one integration for previewing the contract, per action.
- If the contract is to be retrieved in multiple formats, then multiple Modify actions with corresponding "integrations" need to be created.

13. Click **Apply**.

ADD CONTRACTS ENTERPRISE COMPONENTS TO TRANSACTION UI:

Administrators need to add contract components to the Transaction UI.

- **Modify Action(s):** The number of items that need to be added depends upon the number of integration Modify actions.
 - **One component**, if the Create and Preview Contracts integrations are invoked using the same action.
 - **Two components**, if the Create Contract and Preview Contract integrations are invoked using individual actions.

Note: If multiple actions were created to preview the contract in multiple formats, the corresponding actions for each format should be added to the Transaction UI.

- **File Attachment Attribute(s):** That store contract documents can be added to the Transaction UI to allow users access to the saved Preview Contract document(s).
- **Attributes for Enterprise Contract Parameters:** That store contract IDs, contract types, and organization IDs from Enterprise Contracts can also be added to the Transaction UI if needed.
- **Optional Attributes:** That store other Enterprise Contracts parameters or Preview Contract Result Parser XSL results can also be added to the Transaction UI if needed.

Refer to Oracle CPQ Administration Online Help – [Commerce Layout Editor](#) for instructions to add actions and attributes to the Transaction UI.

After adding components to the Transaction UI deploy all the changes to the Commerce process.

REST APIS TO INITIATE CONTRACT MODIFY ACTIONS

REST API services are available for Oracle CPQ Commerce Modify actions. Accordingly, the Create Contract and Preview Contract functionality can be initiated via REST APIs for the corresponding Modify actions.

Commerce REST API

COMMERCE REST API FOR MODIFY ACTIONS		
Description	This REST API invokes the specified Commerce Modify action.	
URI Endpoint	<code>https://<sitename>/rest/v14/commerceDocuments{ProcessVarName}{MainDocVarName}/{_id}/actions/{action_name}</code>	
Endpoint Parameters	ProcessVarName	The variable name of the Commerce process, the first letter must be capitalized
	MainDocVarName	The variable name of the main document, the first letter must be capitalized
	_id	The Commerce Transaction ID
	action_name	The Modify action name used to invoke Contract services. For example: "createContract" or "previewContract"
HTTP Method	POST	
Request Body Parameters	None Request Body Parameters are not required when invoking Contract integration Modify actions.	
Response	JSON data containing all Main Document and Sub Document attributes for the specified Transaction	

URI Endpoint Samples

Example 1: The following endpoint invokes the previously created Create Contract action.

```
https://sitename.oracle.com/rest/v14/commerceDocumentsOraclecpqoTransaction/18103143/actions/createContract
```

Example 2: The following endpoint invokes the previously created Preview Contract action.

```
https://sitename.oracle.com/rest/v14/commerceDocumentsOraclecpqoTransaction/18103143/actions/previewContract
```

APPENDIX A: CREATE CONTRACT XSL FILE SAMPLES

Sample Create Contract SOAP Generator XSL File

This example XSL creates a "Buy:Project Lines" type Contract. If required, this XSL should be modified accordingly to pass different sub-document attributes to Contracts Enterprise.

```
<!-- XSL -->
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
  <xsl:output method="xml"/>
  <!-- data_type=0 : quote-level node -->
  <!-- data_type=2 : model-level node -->
  <!-- data_type=3 : line item node -->
  <xsl:variable select="/transaction/data_xml/document[@data_type=0]" name="main_doc"/>
  <xsl:variable name="startPos" select="0"/>
  <xsl:variable name="endPos" select="200"/>
  <xsl:variable select="/transaction/@id" name="bmTransId"/>
  <xsl:variable select="/transaction/data_xml/document[@data_type=3]" name="line_item"/>
  <xsl:strip-space elements="*" />
  <!-- template: main -->
  <xsl:template match="/">
    <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
      <soap:Body>
        <ns1:createContract
xmlns:ns1="http://xmlns.oracle.com/apps/contracts/coreAuthoring/contractService/types/"
        <ns1:contractHeader
xmlns:ns2="http://xmlns.oracle.com/apps/contracts/coreAuthoring/contractService/">
          <ns2:OrgId>
            <xsl:value-of select="$main_doc/oRCL_OSC_PrimaryOrgID_t"/>
          </ns2:OrgId>
          <ns2:InvOrganizationId>
            <xsl:value-of select="$main_doc/oRCL_OSC_PrimaryOrgID_t"/>
          </ns2:InvOrganizationId>
          <ns2:ContractTypeId>
            <xsl:value-of select="$main_doc/oRCL_EC_ContractTypeId_t"/>
          </ns2:ContractTypeId>
          <ns2:ContractTypeName>
            <xsl:value-of select="$main_doc/oRCL_EC_ContractTypeName_t"/>
          </ns2:ContractTypeName>
          <ns2:ContractNumber>
            <xsl:value-of select="$main_doc/oRCL_EC_ContractNumber_t"/>
          </ns2:ContractNumber>
          <ns2:PartyId>
            <xsl:value-of select="$main_doc/_customer_id"/>
          </ns2:PartyId>
          <ns2:BuyOrSell>B</ns2:BuyOrSell>
          <ns2:StsCode>DRAFT</ns2:StsCode>
          <ns2:CurrencyCode>currency_t</ns2:CurrencyCode>
          <ns2:StartDate>2021-08-23</ns2:StartDate>
          <xsl:for-each select="/transaction/data_xml/document[@data_type=3 or
@data_type=2][position() > $startPos and position() <= $endPos]">
            <xsl:sort select="_sequence_number" data-type="number"/>
            <xsl:variable name="sub_doc" select="."/>
            <ns2:ContractLine>
              <ns2:LineTypeId>300000000099381</ns2:LineTypeId>
Oracle CPQ and Oracle Enterprise Contracts Integration 16
            <ns2:LineTypeName>Free-form, project</ns2:LineTypeName>
              <ns2:StsCode>DRAFT</ns2:StsCode>
              <ns2:LineStatus>Draft</ns2:LineStatus>
            </ns2:ContractLine>
          </xsl:for-each>
        </ns1:contractHeader>
      </soap:Body>
    </soap:Envelope>
  </xsl:template>
</xsl:stylesheet>
```

```

<ns2:ItemNameTxt>
  <!-- <xsl:value-of select="$sub_doc/_part_number"/> -->
  <xsl:choose>
    <xsl:when test="string-length($sub_doc/_part_number) > 1">
      <xsl:value-of select="$sub_doc/_part_number"/>
    </xsl:when>
    <xsl:otherwise>
      <xsl:value-of select="$sub_doc/_model_name"/>
    </xsl:otherwise>
  </xsl:choose>
</ns2:ItemNameTxt>
<ns2:ItemDescription>
  <!-- <xsl:value-of select="$sub_doc/_part_desc"/> -->
  <xsl:choose>
    <xsl:when test="string-length($sub_doc/_part_desc) > 1">
      <xsl:value-of select="$sub_doc/_part_desc"/>
    </xsl:when>
    <xsl:otherwise>
      <xsl:value-of select="$sub_doc/_model_product_line_name"/>
    </xsl:otherwise>
  </xsl:choose>
</ns2:ItemDescription>
<ns2:StartDate>2021-08-23</ns2:StartDate>
<ns2:FixedPriceServiceYn>N</ns2:FixedPriceServiceYn>
</ns2:ContractLine>
</xsl:for-each>
</nsl:contractHeader>
</nsl:createContract>
</soap:Body>
</soap:Envelope>
</xsl:template>
</xsl:stylesheet>

```

Sample Create Contract Result Parser XSL File

The sample file is used to process the Result Parser XSL output for the "Create Contract" web service.

```

<xsl:stylesheet
xmlns:ns2="http://xmlns.oracle.com/apps/contracts/coreAuthoring/contractService/types/"
xmlns:ns1="http://xmlns.oracle.com/apps/contracts/coreAuthoring/contractService/"
xmlns:ns0="http://xmlns.oracle.com/adf/svc/types/"
xmlns:ns5="http://xmlns.oracle.com/apps/projects/billing/contracts/contractPublicService/"
xmlns:ns7="http://xmlns.oracle.com/apps/contracts/coreAuthoring/header/flex/header/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
  <xsl:output method="xml"/>
  <xsl:template match="*">
    <data_xml>
      <document document_var_name="transaction">
        <oRCL_EC_ContractId_t>
          <xsl:value-of select="//nsl:Id"/>
        </oRCL_EC_ContractId_t>
      </document>
    </data_xml>
  </xsl:template>
</xsl:stylesheet>

```

APPENDIX B: PREVIEW CONTRACT XSL FILE SAMPLES

Sample Preview Contract SOAP Generator XSL File

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
  <xsl:output method="xml"/>
  <xsl:variable name="main_doc" select="/transaction/data_xml/document[@data_type=0]"/>
  <xsl:variable name="bmTransId" select="/transaction/@id"/>
  <xsl:strip-space elements="*" />
  <!-- template: main -->
  <xsl:template match="/">
    <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
      <soap:Body>
        <nsl:previewContractAsync
xmlns:nsl="http://xmlns.oracle.com/apps/contracts/coreAuthoring/contractService/types/"
          <nsl:contractId>
            <xsl:value-of select="$main_doc/oRCL_EC_ContractId_t"/>
          </nsl:oRCL_EC_ContractId_t>
          <nsl:majorVersion>1</nsl:majorVersion>
          <nsl:externalSourceKey/>
          <nsl:externalReferenceKey/>
          <nsl:language/>
        </nsl:previewContractAsync>
      </soap:Body>
    </soap:Envelope>
  </xsl:template>
</xsl:stylesheet>
```

Sample Preview Contract Result Parser XSL File

This sample file is used to process a PDF format output for the "Preview Contract" web service.

```
<xsl:stylesheet
xmlns:ns0="http://xmlns.oracle.com/apps/contracts/coreAuthoring/contractService/types/"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
  <xsl:variable name="main_doc" select="/transaction/data_xml/document[@data_type=0]"/>
  <xsl:variable name="bmTransId" select="/transaction/@id"/>
  <xsl:output method="xml"/>
  <!-- template: main -->
  <xsl:template match="/">
    <data_xml>
      <document Document_var_name="transaction">
        <attachment_field_value>
          <xsl:value-of select="//ns0:result[3]"/>
        </attachment_field_value>
        <mime_type>
          <xsl:value-of select="//ns0:result[2]"/>
        </mime_type>
        <file_name>
          <xsl:value-of select="//ns0:result[1]"/>
        </file_name>
        <isActionOfTypePreviewContract>true</isActionOfTypePreviewContract>
        <contractDocumentAttributeName>oRCL_EC_Contract_t</contractDocumentAttributeName>
      </document>
    </data_xml>
  </xsl:template>
</xsl:stylesheet>
```

CONNECT WITH US

Call +1.800.ORACLE1 or visit oracle.com.
Outside North America, find your local office at oracle.com/contact.



blogs.oracle.com



facebook.com/oracle



twitter.com/oracle

Copyright © 2022 Oracle and/or its affiliates. All rights reserved.

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

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

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

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

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

Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications. Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

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

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

