



# CPQ for SOAP API Developers



# Table of Contents

Getting Started as a Developer.....	10
API Supported Packages .....	10
Document Setup.....	10
API Standards and Development Platforms.....	10
Standards .....	11
Development Platforms.....	11
Field Types.....	11
Recommendations .....	12
Integrating Conga with External Systems.....	12
Generating the Conga Web Services WSDL.....	13
Connecting to Conga.....	15
Generic API Operations in Salesforce (SOAP) .....	16
API Reference.....	35
Proposal Web Service.....	35
Retrieving Proposal Field Values from Account.....	36
Retrieving Proposal Field Values from Opportunity.....	41
Retrieving Field Values from Proposal .....	47
Creating Proposal Line Items from Opportunity.....	54
Updating a Proposal .....	57
Cloning Email Template.....	59
Cloning Document Templates .....	61
Getting the Proposal Document Output Format .....	64
Copying Attachments.....	68
Creating a Task After Proposal is Presented .....	69
Retrieving Name of the First Template.....	70
Retrieving Templates in a Proposal.....	71
Deleting an Email Template.....	73
Batch Update Service.....	74
Updating Category Hierarchy for a single Category .....	74

Updating Category Hierarchy for multiple Categories .....	75
CPQ Web Service(Apttus_CPQApi).....	77
Creating a Cart from a Quote .....	78
Retrieving Categories for a Price List .....	82
Retrieving Products and List Prices for a Price List.....	87
Retrieving Products and List Prices for a Price List and Category .....	94
Retrieving Products and List Prices For a Price List and Search Text.....	102
Retrieving Products and List Prices For a Price List Category and Search Text.....	118
Retrieving Option Groups, Options, and List Prices for a Price List Product .....	125
Adding Products to a Cart.....	135
Adding a Bundle to a Cart .....	141
Adding Custom Bundles.....	148
Adding Options to a Bundle.....	156
Cloning Bundle Line Items on the Cart.....	162
Cloning Line Items on the Cart .....	166
Cloning Option Line Items on the Cart .....	171
Comparing Products .....	175
Associating Constraint Rules to a Cart .....	182
Applying Constraint Rules to a Cart .....	185
Applying Constraint Rules to Deleted Products.....	187
Adding Price Ramps to a Cart (CPQ Web Service).....	189
Computing the Net Price for a Bundle .....	194
Updating Price For A Cart.....	198
Price Breakup for a Cart or Specific Line Item.....	203
Removing a Bundle from a Cart .....	210
Removing Multiple Bundles from a Cart.....	213
Removing Line Items from the Cart .....	217
Removing Options from a Bundle .....	220
Retrieving Constraint Rules Results .....	223
Retrieving Incentives on the Cart .....	228
Retrieving Asset Line Items .....	238
Setting Incentives for Cart.....	246
Retrieving Incentives Applied on the Cart.....	247
Abandoning a Cart.....	249

Finalizing a Cart.....	251
Synchronizing a Cart.....	254
Updating Quote Terms.....	257
Modifying Assets (Deprecated).....	264
Computing Shipping for Cart Line Items.....	279
Computing Taxes for Cart Line Items.....	282
Adding a Miscellaneous Item to the Cart.....	287
Creating Coupons for Incentives.....	290
Retrieving Products Included by Auto-inclusion Constraint Rule.....	292
Adding Line Items.....	295
Rebuilds the Category Hierarchy.....	297
CPQ Web Service (Apttus_Config2).....	298
Creating Product Line Items.....	298
Registering Split Cart Parameters.....	302
Finalizing the Configuration Version.....	303
Finalizing the Configuration Version with Synchronization Tasks.....	305
Abandoning a Configuration.....	307
Cloning Primary Line Items.....	308
Creating Bundle Line Items.....	309
Creating and Updating Related Line Items.....	313
Saving Related Line Items.....	316
Splitting Related Line Items.....	319
CPQ Admin Web Service.....	321
Creating New Products.....	322
Creating New Categories.....	326
Associating Products to Category.....	330
Building Hierarchy of Products.....	331
Checking if Product Exists in Hierarchy.....	336
Removing Products from Hierarchy.....	338
Retrieving the List Option Group.....	340
Retrieving Options in an Option Group.....	342
Retrieving of Product Hierarchy.....	344
Retrieves List of Parent Product.....	346
Retrieves List Product IDs.....	348

Retrieves List of Category IDs .....	349
Creating New Attribute Group .....	350
Adding and Removing Attributes from an Attribute Group.....	353
Associating Attribute Group and Products .....	356
Dissociating Attribute Group and Products.....	359
Batch Job Service .....	361
Splitting a Smart Cart.....	362
Updating Price for the Smart Cart.....	363
Finalizing Configuration in Enterprise and Smart Cart .....	364
Retrieving Apex Batch Job Information .....	365
Constraint Web Service 2 .....	368
Associating Constraint Rule to Products Added to the Cart .....	369
Applying Constraint Rule to All Products Added to the Cart.....	370
Applying Constraint Rule to Bundles Added to the Cart.....	372
Retrieving Products Auto-included to the Cart .....	373
Quote Collaboration Service .....	375
Adding Products to a Collaboration Request.....	376
Favorite Configuration Global Service .....	377
Creating Favorite Configuration.....	378
Quote/Proposal Config Web Service .....	381
Accepting a Quote .....	382
Synchronizing Cart.....	384
Copying Product Configuration to the Proposal.....	385
Synchronizing Quote/Proposal with Opportunity .....	386
Asset Service.....	388
Fetching Asset Line Items .....	388
Getting a count of Asset Line Items .....	390
Swapping Assets .....	392
Getting a list of Products to be swapped with Assets.....	394
Terminating Assets .....	396
Renewing Assets .....	397
Incrementing Assets .....	399
Changing Assets.....	402

Merging Duplicate Assets .....	403
Suspending Assets.....	405
Resuming Assets .....	408
Remote CPQ Admin Controller .....	411
Searching Custom Fields in Objects.....	411
Asset Web Services.....	412
Change Assets .....	413
Renew Assets .....	414
Get a List of Products to be Swapped with Assets.....	416
Swap Assets .....	418
Terminate Assets.....	419
Get a Count of Assets.....	421
Get a List of Assets.....	423
Merge Web Service .....	425
Creating Proposal Document.....	426
Creating Proposal Document with Draft Indication .....	428
Generating Documents for Proposal with Large Number of Line Items.....	431
Generating Documents Asynchronously.....	434
Async APIs Using Batch Apex.....	438
Repricing the Cart.....	438
Scenarios.....	440
Working with Products in the Shopping Cart.....	440
Searching Products .....	441
Configuring and Adding Products.....	441
Displaying Recommendations .....	443
Updating Price.....	443
Viewing Price Break-up Details.....	443
Adding Price Ramps to a Cart .....	444
Updating Taxes and Shipping for an Order .....	449
Creating and Updating Quote and Quote Line Items.....	459
Creating and Updating Order and Order Line Items.....	460
Asset Based Ordering.....	461
Finalizing the Cart .....	462

Updating Quote Terms to Modify Dates .....462

Troubleshooting CPQ SOAP APIs..... 464

This section explains the SOAP APIs provided by Conga Configuration, Pricing, and Quoting (CPQ).

Topic	Description
What's Covered	This section walks the API developers through the list of SOAP APIs provided by Conga.
Primary Audience	<ul style="list-style-type: none"> <li>• API developers</li> </ul>
IT Environment	Refer to the latest <i>Conga CPQ Release Notes</i> for information on System Requirements and Supported Platforms.
Updates	For a comprehensive list of updates to this section for each release, see the <a href="#">What's New in CPQ Documentation</a> topic.
Other Resources	Refer to <i>Conga CPQ Release Notes</i> for information on system requirements and supported platforms, new features and enhancements, resolved issues, and known issues for a specific release.

This section describes the following topics:

- [CPQ Web Service \(Apttus\\_Config2\)](#)
- [CPQ Web Service\(Apttus\\_CPQApi\)](#)
- [Batch Update Service](#)
- [Quote/Proposal Config Web Service](#)
- [Favorite Configuration Global Service](#)
- [Quote Collaboration Service](#)
- [Asset Service](#)
- [Batch Job Service](#)
- [Proposal Web Service](#)

Before using Conga CPQ, you must be familiar with the following:

- Basic knowledge of Salesforce
- Basic knowledge of SOAP APIs
- Salesforce and Conga terms and definitions

Select one of the following topics for more information:

- [Getting Started as a Developer](#)
- [API Reference](#)



- Async APIs Using Batch Apex
- Scenarios
- Troubleshooting CPQ SOAP APIs

# Getting Started as a Developer

This section describes the following topics:

- [API Supported Packages](#)
- [Document Setup](#)
- [API Standards and Development Platforms](#)
- [Field Types](#)
- [Recommendations](#)
- [Integrating Conga with External Systems](#)

## API Supported Packages

Refer to the *Conga CPQ Release Notes* for the package details of the latest release.

## Document Setup

The CPQ API reference Guide is divided into two sections: *API Reference* and *Scenarios*.

<a href="#">API Reference</a>	The <i>API reference</i> section details the APIs that you can use to manipulate Conga objects through API calls and passing parameters. The API reference section also includes some code samples.
<a href="#">Scenarios</a>	The <i>Scenarios</i> section details examples of the APIs you require to complete a specific task such as, adding products and constraint rules to the cart. The scenarios are classified by theme. You can refer to the generic examples of scenarios to identify the calls you can use to achieve your objective.

## API Standards and Development Platforms

CPQ APIs are based on Salesforce APIs and use the same standards and platforms

## Standards

Name	Reference
Simple Object Access Protocol (SOAP) 1.1	<a href="http://www.w3.org/TR/2000/NOTE-SOAP-20000508">http://www.w3.org/TR/2000/NOTE-SOAP-20000508</a>
Web Service Description Language (WSDL) 1.1	<a href="http://www.w3.org/TR/2001/NOTE-wsdl-20010315">http://www.w3.org/TR/2001/NOTE-wsdl-20010315</a>
WS-I Basic Profile 1.1	<a href="http://www.ws-i.org/Profiles/BasicProfile-1.1-2004-08-24.html">http://www.ws-i.org/Profiles/BasicProfile-1.1-2004-08-24.html</a>

## Development Platforms

CPQ SOAP API works with standard SOAP development environments. For a list of compatible development platforms, see [Salesforce Developer Force API](#) details.

## Field Types

CPQ APIs use a subset of the supported data and field types on Salesforce.

The following table lists the APIs that Conga provides. For a comprehensive list of all field types supported by Salesforce, see [Salesforce Data Types](#).

Type	Description
Boolean	The <b>Boolean</b> field has a <i>true</i> (or 1) or <i>false</i> (or 0) value.
Data object	The <b>Data Object</b> field is an ID type and is represented by <i>CPQ.nmDO</i> in this document.
Date	The <b>Date</b> field contains date values only and do not contain relevant time values. Time in a date field is always set to midnight in the UTC time zone. If you want a timestamp you must use a <b>dateTime</b> field.
Decimal	The <b>Decimal</b> field provides an exact numeric value and you can arbitrarily size the precision and scale of the value.

Type	Description
ID	<p>The <b>ID</b> field is an alphanumeric field that acts like the primary key for a specific record associated with an object. The ID value includes a three-character code that identifies which object the record is associated with. The ID for a specific record does not change.</p> <p>For some objects, this field may also be a <code>referenceType</code> value, which contains the ID value for a related record. They are identified by field names ending in 'Id', such as <code>priceListId</code>. The ID field acts like foreign keys and their values can be changed using an <code>update()</code> call.</p>
Integer	The <b>Integer</b> field contains whole numbers only. There are no digits after the decimal.
List	The <b>List</b> field includes a fixed set of values from which you must select a single value. Picklists are available as drop-down lists. If a picklist is unrestricted, the API does not limit entries to only currently active values.
String	The <b>String</b> field contains text and may have differing length restrictions based on the data you store in the specific field. For instance, <code>city</code> may be limited to 50 characters, while <code>addressLine1</code> is limited to 255 characters.

## Recommendations

Following are the recommendations for CPQ SOAP APIs on the Salesforce Platform:

- You must use the Public Admin APIs to integrate the CPQ master data.  
In a nutshell, the master data includes products, rules, and pricing setup as a whole.
- You cannot use generic APIs for any of the master datasets.  
CPQ does not provide support for any data that is integrated through the generic API mechanism for client implementation.

## Integrating Conga with External Systems

Additional steps are required when you choose to integrate CPQ on Salesforce with external applications, customer portals, or other critical business systems. Because CPQ Web Services are hosted on Salesforce, you should familiarize yourself with the Salesforce

SOAP API and processes surrounding integration and best practices detailed here: [https://developer.salesforce.com/docs/atlas.en-us.api.meta/api/sforce\\_api\\_quickstart\\_intro.htm](https://developer.salesforce.com/docs/atlas.en-us.api.meta/api/sforce_api_quickstart_intro.htm)

### **Best Practices**

It is recommended that you work with Professional Services to design and implement your integration. Use this documentation for basic integration steps and to reference CPQ Web Service calls.


The following basic steps are required to get started using the CPQ Web Services API.

1. **Generate the Enterprise or Partner WSDL** – Integration with data stored in Salesforce requires you to first point your browser to the Salesforce Enterprise or Partner WSDL. This WSDL is generally provided by Professional Services. Refer to [Salesforce Documentation](#) for complete instructions on generating the Web Service WSDL.
2. **Generate the Web Services WSDL** – After you have connected to Web Service, go to your organization and download the WSDL for the appropriate Web Service (CPQ Web Service, Batch Update, Proposal Web Service, and so on).
3. **Import the WSDL Files Into Your Development Platform** – After you have generated the WSDL files, you can import them into your development platform. Conga does not provide instructions for the import process. Refer to Salesforce documentation or documentation related to your development platform.
4. **Connect to Conga** – Before you can begin using CPQ Web Services, you must first authenticate to Conga using the `login()` API.

## Generating the Conga Web Services WSDL

Before you can import Conga SOAP Web Service into your development or testing platform, you must generate and download the Conga WSDL for the appropriate Web Service.

The example provided here uses SoapUI.

-  There is a known bug in the WSDL Generator on Salesforce that does not include several field types, so it is recommended to update the WSDL file after you have generated it but *before* importing it into your development platform. You can find the details for any workaround tasks here:
- [https://success.salesforce.com/issues\\_view?id=a1p3A000000eatxQAA&title=generated-wsdl-for-apex-webservices-is-malformed](https://success.salesforce.com/issues_view?id=a1p3A000000eatxQAA&title=generated-wsdl-for-apex-webservices-is-malformed)
  - [https://success.salesforce.com/issues\\_view?id=a1p300000008XKUA2](https://success.salesforce.com/issues_view?id=a1p300000008XKUA2)


When updating generated WSDL, make sure that the target namespace for any schema you add points to the correct Web Service (for example, schemas/class/Apttus\_QPConfig/QPConfigWebService). If you are still having trouble, please ask Conga Professional Services for a modified WSDL for the Web Services you are using.

## To generate the Conga Web Services WSDL

1. Log in to the Salesforce organization that contains your Conga records and data (sandbox or production).
2. Go to **Setup > Develop > Apex Classes** (on Lightning, go to **Setup > Custom Code > Apex Classes**).
3. Find the Web Service you want to generate the WSDL for (for example, **CPQWebService**).
4. Click the **WSDL** link to generate the WSDL. The WSDL XML is generated and displayed in a new tab.

<a href="#">Edit</a>	 <a href="#">CPQSystemOverviewController</a>	Apttus_Config2
<a href="#">Edit</a>	 <a href="#">CPQSystemOverviewControllerTest</a>	Apttus_Config2
<a href="#">Edit</a>   <a href="#">Security</a>	 <a href="#">CPQWebService</a>	Apttus_CPQApi
<a href="#">Edit</a>   <b>WSDL</b>   <a href="#">Security</a>	 <a href="#">CPQWebService</a>	Apttus_Config2
<a href="#">Edit</a>	 <a href="#">CPQWebServiceSupport</a>	Apttus_CPQApi
<a href="#">Edit</a>	 <a href="#">CPQWebServiceTest</a>	Apttus_CPQApi
<a href="#">Edit</a>	 <a href="#">CPQWebServiceTest</a>	Apttus_Config2

5. Right-click on the page and select **View Page Source**. Copy the XML content to any text editor.
6. Save the file with the extension **.wsdl**.
7. Open SoapUI (or wherever is required on your development platform).
8. Create a new SOAP project and import the Conga Web Services WSDL. All methods under that Web Service are now available to call.

 Refer to the **Request/Response XML** section for any API in this reference to get the structure of the request and any prerequisite calls required for any API.

## Connecting to Conga


After you have downloaded the Enterprise or Partner WSDL, call the **login()** method to obtain a session ID from your org that you can use when calling CPQ Web Services. After authenticating, you can use the same session ID until it either expires or your logout or login again.

The example provided here uses SoapUI, an API testing tool which can be downloaded for free here: <https://www.soapui.org/>.

**Prerequisite:** To authenticate with Conga, please make sure to have your production or test org credentials on hand (username and password).

## To connect to Conga Web Service using SoapUI

1. Open SoapUI. Go to **File > New SOAP Project**.
2. Enter a name for the project.
3. Click **Browse**. Navigate to the saved Enterprise or Partner WSDL file that you downloaded and click **Open**.
4. Click **OK** to close the project window.
5. From the Navigation panel to the left, highlight the project folder and click to expand. Click to expand the **SoapBinding**. The list of methods that comprise the Enterprise or Partner services are displayed.
6. Scroll down and right click on **login**. Double-click on an existing **Request**. The request window opens in the SoapUI interface.

 If you are doing this for the first time, you need to right-click on the **login** method and select **New Request**.

7. Select and delete all content following the **<soapenv:Header>** tag and the **</soapenv:Header>** tag.
8. Enter the username for your org (must have appropriate privileges) between the **<urn:username>** and **</urn:username>** tags.
9. Enter the password for your org (must have appropriate privileges) between the **<urn:password>** and **</urn:password>** tags.

The request should look like the following:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:urn="urn:partner.soap.sforce.com">
  <soapenv:Header>
  </soapenv:Header>
  <soapenv:Body>
    <urn:login>
      <urn:username>username@example.com</urn:username>
      <urn:password>password</urn:password>
    </urn:login>
  </soapenv:Body>
</soapenv:Envelope>
```

10. From the upper-left corner of the window, click the **Run** (▶) icon. The response is generated in the right-hand window.
11. Make note of the **serverURL** and the **sessionId** returned by the server. You will use the information to make Conga Web Service calls.

## Generic API Operations in Salesforce (SOAP)

The following methods are provided in the Enterprise or Partner WSDL as generic core API calls that allow you primarily to perform CRUD actions on standard and custom Salesforce objects (also referred to as "data setup"). Refer to [Salesforce Developer documentation](#) for complete information on calling these methods. The examples provided on this page will aid you in performing CRUD actions and querying information from the Salesforce database to use in Conga API calls.

### Retrieving the Server URL and Session Id (Login)

All API calls to your organization require you to authenticate with a valid Session Id.

Method: `login()`

Request: Pass the **username** and **password** for the user providing authentication.

Example Request
<pre>&lt;soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:urn="urn:partner.soap.sforce.com"&gt;   &lt;soapenv:Header&gt;</pre>



```

</soapenv:Header>
<soapenv:Body>
  <urn:login>
    <urn:username>username</urn:username>
    <urn:password>password</urn:password>
  </urn:login>
</soapenv:Body>
</soapenv:Envelope>

```

Response: Returns the **serverURL** and **session Id**.

### Example Response

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="urn:partner.soap.sforce.com" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <loginResponse>
      <result>
        <serverUrl>serverURL</serverUrl>
        <sessionId>sessionId</sessionId>
      </result>
    </loginResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Retrieving an Opportunity

Method: `query()`

Request: Pass the **session Id** and **Opportunity Name**.

### Example Request

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:enterprise.soap.sforce.com">
  <soapenv:Header>
    <urn:SessionHeader>
      <urn:sessionId>session Id</urn:sessionId>
    </urn:SessionHeader>
  </soapenv:Header>

```

```

<soapenv:Body>
  <urn:query>
    <urn:queryString>select id from opportunity where
name='opportunityName'</urn:queryString>
  </urn:query>
</soapenv:Body>
</soapenv:Envelope>

```

Response: Returns the **Opportunity Id** and **Size** (indicates number of Ids)

### Example Response

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="urn:enterprise.soap.sforce.com"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:sf="urn:subject.enterprise.soap.sforce.com">
  <soapenv:Header>
    <LimitInfoHeader>
      <limitInfo>
        <current>4973</current>
        <limit>5000000</limit>
        <type>API REQUESTS</type>
      </limitInfo>
    </LimitInfoHeader>
  </soapenv:Header>
  <soapenv:Body>
    <queryResponse>
      <result>
        <done>true</done>
        <queryLocator xsi:nil="true"/>
        <records xsi:type="sf:Opportunity">
          <sf:Id>006Z000000Gr90DIAZ</sf:Id>
        </records>
        <size>1</size>
      </result>
    </queryResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

# Creating a Price List

Method: `create()`

Request: Pass the **session Id** and **Price List Name**.

## Example Request

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:enterprise.soap.sforce.com"
  xmlns:urn1="urn:subject.enterprise.soap.sforce.com">
  <soapenv:Header>
    <urn:SessionHeader>
      <urn:sessionId>session Id</urn:sessionId>
    </urn:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <urn:create>
      <urn:sObjects xsi:type="Apttus_Config2__PriceList__c"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
        <Name>Price List Name</Name>
      </urn:sObjects>
    </urn:create>
  </soapenv:Body>
</soapenv:Envelope>
```

Response: Returns the **Price List Id** with status = 'true' if the Price List was successfully created.

## Example Response

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="urn:enterprise.soap.sforce.com">
  <soapenv:Header>
    <LimitInfoHeader>
      <limitInfo>
        <current>4983</current>
        <limit>50000000</limit>
        <type>API REQUESTS</type>
      </limitInfo>
    </LimitInfoHeader>
  </soapenv:Header>
  <soapenv:Body>
    <urn:PriceListId>PriceListId</urn:PriceListId>
  </soapenv:Body>
</soapenv:Envelope>
```

```

        </LimitInfo>
    </LimitInfoHeader>
</soapenv:Header>
<soapenv:Body>
    <createResponse>
        <result>
            <id>a1DZ0000002i9IWMAY</id>
            <success>>true</success>
        </result>
    </createResponse>
</soapenv:Body>
</soapenv:Envelope>

```

## Retrieving a Price List

Method: `query()`

Request: Pass the **session Id** and **Price List Name**.

### Example Request

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:enterprise.soap.sforce.com">
  <soapenv:Header>
    <urn:SessionHeader>
      <urn:sessionId>${#Project#SessionID}</urn:sessionId>
    </urn:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <urn:query>
      <urn:queryString>select Id from Apttus_Config2__PriceList__c where
        name='priceListName'
      </urn:queryString>
    </urn:query>
  </soapenv:Body>
</soapenv:Envelope>

```

Response: Returns the **Price List Id**.

**Example Response**

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="urn:partner.soap.sforce.com" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:sf="urn:subject.partner.soap.sforce.com">
  <soapenv:Header>
    <LimitInfoHeader>
      <limitInfo>
        <current>637</current>
        <limit>5000000</limit>
        <type>API REQUESTS</type>
      </limitInfo>
    </LimitInfoHeader>
  </soapenv:Header>
  <soapenv:Body>
    <queryResponse>
      <result xsi:type="QueryResult">
        <done>true</done>
        <queryLocator xsi:nil="true"/>
        <records xsi:type="sf:sObject">
          <sf:type>Apttus_Config2__PriceList__c</sf:type>
          <sf:Id>a1DZ00000002mg5nMAA</sf:Id>
        </records>
        <size>1</size>
      </result>
    </queryResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Creating a Product Category

Method: `create()`

Request: Pass the **session Id**, **Category Name**, and **Category Label**.

**Example Request**

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:urn="urn:enterprise.soap.sforce.com" xmlns:urn1="urn:subject.enterprise.soap.sforce.com">
  <soapenv:Header>
    <urn:SessionHeader>

```

```

    <urn:sessionId>session Id</urn:sessionId>
  </urn:SessionHeader>
</soapenv:Header>
<soapenv:Body>
  <urn:create>
    <urn:sObjects xsi:type="Apttus_Config2__ClassificationName__c" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <Name>Category Name</Name>
      <Apttus_Config2__HierarchyLabel__c>Category Label</Apttus_Config2__HierarchyLabel__c>
    </urn:sObjects>
  </urn:create>
</soapenv:Body>
</soapenv:Envelope>

```

Response: Returns the **Category Id** with status = 'true' if the Category was successfully created.

### Example Response

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="urn:enterprise.soap.sforce.com">
  <soapenv:Header>
    <LimitInfoHeader>
      <limitInfo>
        <current>5011</current>
        <limit>5000000</limit>
        <type>API REQUESTS</type>
      </limitInfo>
    </LimitInfoHeader>
  </soapenv:Header>
  <soapenv:Body>
    <createResponse>
      <result>
        <id>a0oZ0000002u94bIAA</id>
        <success>true</success>
      </result>
    </createResponse>
  </soapenv:Body>
</soapenv:Envelope>undefined</soapenv:Body>undefined</soapenv:Envelope>

```

## Retrieving a Category

Method: `query()`

Request: Pass the **session Id** and **Category Name**.

### Example Request

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:enterprise.soap.sforce.com">
  <soapenv:Header>
    <urn:SessionHeader>
      <urn:sessionId>${#Project#SessionID}</urn:sessionId>
    </urn:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <urn:query>
      <urn:queryString>select Id, Name from
Apttus_Config2__ClassificationName__c where
      name='Category Name'
    </urn:queryString>
  </urn:query>
</soapenv:Body>
</soapenv:Envelope>
```

Response: Returns the **Category Id**, **Category Name**, and **Size** (indicates number of Ids).

### Example Response

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="urn:enterprise.soap.sforce.com"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:sf="urn:subject.enterprise.soap.sforce.com">
  <soapenv:Header>
    <LimitInfoHeader>
      <limitInfo>
        <current>5017</current>
        <limit>5000000</limit>
        <type>API REQUESTS</type>
      </limitInfo>
    </LimitInfoHeader>
  </soapenv:Header>
```

```

        </LimitInfo>
    </LimitInfoHeader>
</soapenv:Header>
<soapenv:Body>
    <queryResponse>
        <result>
            <done>true</done>
            <queryLocator xsi:nil="true"/>
            <records xsi:type="sf:Apttus_Config2__ClassificationName__c">
                <sf:Id>a0oe0000005137UAAQ</sf:Id>
                <sf:Name>AutoQuoteOfferings</sf:Name>
            </records>
            <size>1</size>
        </result>
    </queryResponse>
</soapenv:Body>
</soapenv:Envelope>undefined</soapenv:Body>undefined</soapenv:Envelope>undefined</
soapenv:Envelope>

```

## Creating a Standalone Product

Method: `create()`

Request: Pass the **session Id**, **Product Name**, and **Configuration Type**.

### Example Request

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:enterprise.soap.sforce.com"
  xmlns:urn1="urn:subject.enterprise.soap.sforce.com">
  <soapenv:Header>
    <urn:SessionHeader>
      <urn:sessionId>${#Project#SessionID}</urn:sessionId>
    </urn:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <urn:create>
      <urn:sObjects xsi:type="Product2"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
        <Name>${#Project#createStandaloneProductName}</Name>

```



```

        <Apttus_Config2__ConfigurationType__c>Standalone</
Apttus_Config2__ConfigurationType__c>
        </urn:sObjects>
    </urn:create>
</soapenv:Body>
</soapenv:Envelope>

```

Response: Returns the **Product Id** with status = 'true' if the Product was successfully created.

### Example Response

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="urn:enterprise.soap.sforce.com">
  <soapenv:Header>
    <LimitInfoHeader>
      <limitInfo>
        <current>5033</current>
        <limit>5000000</limit>
        <type>API REQUESTS</type>
      </limitInfo>
    </LimitInfoHeader>
  </soapenv:Header>
  <soapenv:Body>
    <createResponse>
      <result>
        <id>01tZ0000004ypoaIAA</id>
        <success>>true</success>
      </result>
    </createResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Retrieving a Standalone Product

Method: `query()`

Request: Pass the **session Id** and **Product Name**.

**Example Request**

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:enterprise.soap.sforce.com">
  <soapenv:Header>
    <urn:SessionHeader>
      <urn:sessionId>${#Project#SessionID}</urn:sessionId>
    </urn:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <urn:query>
      <urn:queryString>select Id, Name from Product2 where name='Standalone
Product Name'</urn:queryString>
    </urn:query>
  </soapenv:Body>
</soapenv:Envelope>

```

Response: Returns the **Standalone Product Id**, **Product Name**, and **Size** (indicates the number of Ids).

**Example Response**

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="urn:partner.soap.sforce.com"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:sf="urn:subject.partner.soap.sforce.com">
  <soapenv:Header>
    <LimitInfoHeader>
      <limitInfo>
        <current>638</current>
        <limit>5000000</limit>
        <type>API REQUESTS</type>
      </limitInfo>
    </LimitInfoHeader>
  </soapenv:Header>
  <soapenv:Body>
    <queryResponse>
      <result xsi:type="QueryResult">
        <done>true</done>

```

```

    <queryLocator xsi:nil="true"/>
    <records xsi:type="sf:sObject">
      <sf:type>Product2</sf:type>
      <sf:Id>01tZ0000004LUXeIAM</sf:Id>
      <sf:Name>Auto_API_StandAloneProduct3</sf:Name>
    </records>
    <size>1</size>
  </result>
</queryResponse>
</soapenv:Body>
</soapenv:Envelope>

```

## Creating a Bundle Product

Method: `create()`

Request: Pass the **session Id**, **Product Name**, and **Configuration Type**.

### Example Request

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:enterprise.soap.sforce.com"
  xmlns:urn1="urn:subject.enterprise.soap.sforce.com">
  <soapenv:Header>
    <urn:SessionHeader>
      <urn:sessionId>${#Project#SessionID}</urn:sessionId>
    </urn:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <urn:create>
      <urn:sObjects xsi:type="Product2"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
        <Name>Bundle Product Name</Name>
        <Apttus_Config2__ConfigurationType__c>Bundle</
Apttus_Config2__ConfigurationType__c>
      </urn:sObjects>
    </urn:create>
  </soapenv:Body>
</soapenv:Envelope>

```

Response: Returns the **Product Id** with status = 'true' if the Bundle Product was successfully created.

**Example Response**

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="urn:enterprise.soap.sforce.com">
  <soapenv:Header>
    <LimitInfoHeader>
      <limitInfo>
        <current>5033</current>
        <limit>5000000</limit>
        <type>API REQUESTS</type>
      </limitInfo>
    </LimitInfoHeader>
  </soapenv:Header>
  <soapenv:Body>
    <createResponse>
      <result>
        <id>01tZ0000004yrtyaIAA</id>
        <success>>true</success>
      </result>
    </createResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Retrieving a Bundle

Method: `query()`Request: Pass the **session Id** and **Product Name**.**Example Request**

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:urn="
urn:partner.soap.sforce.com">
  <soapenv:Header>
    <urn:SessionHeader>
      <urn:sessionId>00DZ000000NAEIA!
ASAAQHmIRqgn4R90i1yQjWTIVk4UZmsDe_.eK0Z9z6qLij7Tu.L_Yo8dRA_p80mhKMeRs4uzCZTadIgQ9fEbD
KciXEQRyaA</urn:sessionId>
    </urn:SessionHeader>

```

```

</soapenv:Header>
<soapenv:Body>
  <urn:query>
    <urn:queryString>select Id, Name from Product2 where name='Bundle Product
Name'
    </urn:queryString>
  </urn:query>
</soapenv:Body>
</soapenv:Envelope>

```

Response: Returns the **Bundle Id** and **Product Name**, and **Size** (indicates the number of Ids).

### Example Response

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="urn:partner.soap.sforce.com" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:sf="urn:subject.partner.soap.sforce.com">
  <soapenv:Header>
    <LimitInfoHeader>
      <limitInfo>
        <current>643</current>
        <limit>5000000</limit>
        <type>API REQUESTS</type>
      </limitInfo>
    </LimitInfoHeader>
  </soapenv:Header>
  <soapenv:Body>
    <queryResponse>
      <result xsi:type="QueryResult">
        <done>true</done>
        <queryLocator xsi:nil="true"/>
        <records xsi:type="sf:sObject">
          <sf:type>Product2</sf:type>
          <sf:Id>01tZ0000004LUXtIAM</sf:Id>
          <sf:Name>Auto_API_BundleProduct1</sf:Name>
        </records>
        <size>1</size>
      </result>
    </queryResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Creating an Option Product

Method: `create()`

Request: Pass the **session Id**, **Product Name**, and **Configuration Type**.

### Example Request

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:enterprise.soap.sforce.com"
  xmlns:urn1="urn:subject.enterprise.soap.sforce.com">
  <soapenv:Header>
    <urn:SessionHeader>
      <urn:sessionId>${#Project#SessionID}</urn:sessionId>
    </urn:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <urn:create>
      <urn:sObjects xsi:type="Product2"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
        <Name>Option Product Name</Name>
        <Apttus_Config2__ConfigurationType__c>Option</
Apttus_Config2__ConfigurationType__c>
      </urn:sObjects>
    </urn:create>
  </soapenv:Body>
</soapenv:Envelope>
```

Response: Returns the **Product Id** with status = 'true' if the Option Product was successfully created.

### Example Response

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="urn:enterprise.soap.sforce.com">
  <soapenv:Header>
    <LimitInfoHeader>
      <limitInfo>
```

```

        <current>5033</current>
        <limit>5000000</limit>
        <type>API REQUESTS</type>
      </limitInfo>
    </LimitInfoHeader>
  </soapenv:Header>
  <soapenv:Body>
    <createResponse>
      <result>
        <id>01tZ0008984yrterIAA</id>
        <success>>true</success>
      </result>
    </createResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Retrieving an Option

Method: `query()`

Request: Pass the **session Id** and **Product Name**.

### Example Request

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:urn="
urn:partner.soap.sforce.com">
  <soapenv:Header>
    <urn:SessionHeader>
      <urn:sessionId>00DZ000000NAEIA!
ASAAQHmIRqgn4R90i1yQjWTIVk4UZmsDe_.eK0Z9z6qLij7Tu.L_Yo8dRA_p80mhKMeRs4uzCZTadIgQ9fEbd
KciXEQYRyaA</urn:sessionId>
    </urn:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <urn:query>
      <urn:queryString>select Id, Name from Product2 where name='Option Product
Name'
    </urn:queryString>
  </urn:query>
</soapenv:Body>
</soapenv:Envelope>

```

Response: Returns the **Option Id**, **Product Name**, and **Size** (indicates the number of Ids).

**Example Response**

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="urn:partner.soap.sforce.com" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:sf="urn:subject.partner.soap.sforce.com">
  <soapenv:Header>
    <LimitInfoHeader>
      <limitInfo>
        <current>648</current>
        <limit>5000000</limit>
        <type>API REQUESTS</type>
      </limitInfo>
    </LimitInfoHeader>
  </soapenv:Header>
  <soapenv:Body>
    <queryResponse>
      <result xsi:type="QueryResult">
        <done>true</done>
        <queryLocator xsi:nil="true"/>
        <records xsi:type="sf:sObject">
          <sf:type>Product2</sf:type>
          <sf:Id>01tZ000000041UYNIA2</sf:Id>
          <sf:Name>Auto_API_OptionProduct2</sf:Name>
        </records>
        <size>1</size>
      </result>
    </queryResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Retrieving Component Id for a Product, Bundle, or Option

Method: `query()`

Request: Pass the **session Id** and **Product Id**.



**Example Request**

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:enterprise.soap.sforce.com">
  <soapenv:Header>
    <urn:SessionHeader>
      <urn:sessionId>${#Project#SessionID}</urn:sessionId>
    </urn:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <urn:query>
      <urn:queryString>select Id from Apttus_Config2__ProductOptionComponent__c
where
  Apttus_Config2__ComponentProductId__c = 'Product Id'</urn:queryString>
    </urn:query>
  </soapenv:Body>
</soapenv:Envelope>

```

Response: Returns the **Component Product Id**.

**Example Response**

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="urn:partner.soap.sforce.com"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:sf="urn:subject.partner.soap.sforce.com">
  <soapenv:Header>
    <LimitInfoHeader>
      <limitInfo>
        <current>649</current>
        <limit>5000000</limit>
        <type>API REQUESTS</type>
      </limitInfo>
    </LimitInfoHeader>
  </soapenv:Header>
  <soapenv:Body>
    <queryResponse>
      <result xsi:type="QueryResult">
        <done>true</done>
        <queryLocator xsi:nil="true"/>
      </result>
    </queryResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

```
<records xsi:type="sf:sObject">
  <sf:type>Apttus_Config2__ProductOptionComponent__c</sf:type>
  <sf:Id>a1YZ00000003pII9MAM</sf:Id>
</records>
<size>1</size>
</result>
</queryResponse>
</soapenv:Body>
</soapenv:Envelope>
```

# API Reference

In this section the following topics are described:

- Proposal Web Service
- Batch Update Service
- CPQ Web Service(Apttus\_CPQApi)
- CPQ Web Service (Apttus\_Config2)
- CPQ Admin Web Service
- Batch Job Service
- Constraint Web Service 2
- Quote Collaboration Service
- Favorite Configuration Global Service
- Quote/Proposal Config Web Service
- Asset Service
- Remote CPQ Admin Controller
- Asset Web Services
- Merge Web Service

**i** The APIs typically create a task after completing execution. These tasks are created for audit purposes and cannot be disabled. You may create custom fields and trigger to differentiate the tasks created by APIs.

## Proposal Web Service

The Proposal web service APIs account for the standard actions to account, opportunity, and quotes/proposals.

You can invoke APIs in Proposal Service using the following command:

```
Apttus_Proposal.ProposalWebService.<Name of the Function>
where the name of the function is API Name and it's parameters.
```

You can use the Proposal web service APIs to complete the following tasks:

- Retrieving Proposal Field Values from Account
- Retrieving Proposal Field Values from Opportunity
- Retrieving Field Values from Proposal
- Creating Proposal Line Items from Opportunity
- Updating a Proposal

- [Cloning Email Template](#)
- [Cloning Document Templates](#)
- [Getting the Proposal Document Output Format](#)
- [Copying Attachments](#)
- [Creating a Task After Proposal is Presented](#)
- [Retrieving Name of the First Template](#)
- [Retrieving Templates in a Proposal](#)
- [Deleting an Email Template](#)

## Retrieving Proposal Field Values from Account

This API fetches the field values of the proposal object that are required to create a proposal from an account. You can use the fetched values to create a new proposal.

API	Signature
createProposalFromAccount	<i>webservice static Apttus_Proposal__Proposal__c createProposalFromAccount(Id accountId, Id recordTypeId)</i>

Request Parameter		
Name	Type	Description
accountId	ID	The ID of the account you need to retrieve values from.
recordTypeId	ID	Record Type of the proposal object

Response - Apttus_Proposal__Proposal__c		
Field	Type	Description
RecordTypeId	ID	Record Type of the proposal object
Apttus_Proposal__Proposal__Name__c	String	The name of the proposal.
Apttus_Proposal__Account__c	ID	The Id of the account associated with the proposal.
Apttus_Proposal__Primary__Contact__c	ID	The primary contact associated with the proposal.
Apttus_Proposal__Description__c	String	The description of the proposal.
Apttus_Proposal__Primary__c	Boolean	Indicates whether the proposal is the primary quote to update the related opportunity.
Apttus_Proposal__ReadyToGenerate__c	Boolean	Indicates whether the proposal is ready for generation.
Apttus_Proposal__ReadyToPresent__c	Boolean	Indicates whether the proposal is ready for presentation.
OwnerId	ID	The ID of the proposal object.
CurrencyIsoCode	String	The currency defined for the proposal.

### Code Sample

The sample code below enables you to retrieve the proposal field values of the account associated with the account ID that you provide. You can use the standard *createRecord* API to create the proposal.

1	/**
---	-----

```

2      * The below method demonstrates fetch the values proposal field values
      from an existing account
3      */
4      public static Apttus_Proposal__Proposal__c createProposal(Id accountID) {
5      Apttus_Proposal__Proposal__c newProposalSO = new
      Apttus_Proposal__Proposal__c();
6      ID proposalRecordTypeID = [select ID FROM recordType where
      sObjectType='Apttus_Proposal__Proposal__c' AND Name = 'Proposal' LIMIT 1].
      Id;
7      newProposalSO =
      Apttus_Proposal.ProposalWebService.createProposalFromAccount(accountID,
      proposalRecordTypeID);
8          return newProposalSO;
9      }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

- [Retrieve Proposal information](#)
- [Create Proposal record](#)

## API Prerequisites

None.

## Response/Request XML

### Example Request

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:prop="http://soap.sforce.com/schemas/class/Apttus_Proposal/
  ProposalWebService">
  <soapenv:Header>
    <prop:SessionHeader>
      <prop:sessionId>00DZ000000NAEIA!
      ASAAQKosATSpsGeD7FUh4RDI18xnzIFPe80Mk89ejrAmDrCBy2lmyJKQjKvuWj3TvT71r6g_epvbo6FeqKUPf
      mSookHTtt0u</prop:sessionId>

```

```

    </prop:SessionHeader>
</soapenv:Header>
<soapenv:Body>
  <prop:createProposalFromAccount>
    <prop:accountId>001Z000001UtoRL</prop:accountId>
  </prop:createProposalFromAccount>
</soapenv:Body>
</soapenv:Envelope>

```

### Example Response

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_Proposal/ProposalWebService"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <createProposalFromAccountResponse>
      <result xsi:type="Apttus_Proposal__Proposal__c">
        <Id xsi:nil="true"/>
        <Apttus_Proposal__Account__c>001Z000001V13ovIAB</
Apttus_Proposal__Account__c>
        <Apttus_Proposal__Primary__c>>false</Apttus_Proposal__Primary__c>
        <Apttus_Proposal__Proposal_Name__c>Auto_API_Account</
Apttus_Proposal__Proposal_Name__c>
        <Apttus_Proposal__ReadyToGenerate__c>>true</
Apttus_Proposal__ReadyToGenerate__c>
        <Apttus_Proposal__ReadyToPresent__c>>true</
Apttus_Proposal__ReadyToPresent__c>
        <CurrencyIsoCode>USD</CurrencyIsoCode>
        <OwnerId>0050U000000r444QAA</OwnerId>
      </result>
    </createProposalFromAccountResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Creating a Proposal

Use data from the Response to create the proposal. The following example uses the `createRecord()` method to create the Proposal object.

**Example Request**

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:enterprise.soap.sforce.com"
  xmlns:urn1="urn:subject.enterprise.soap.sforce.com"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Header>
    <urn:SessionHeader>
      <urn:sessionId>00DZ000000NAEIA!
ASAAQKosATSpsGeD7FUh4RDI18xnzIFPe80Mk89ejrAmDrCBy2lmyJKQjKvuwj3TvT71r6g_epvbo6FeqKUPf
mSookHTttOu</urn:sessionId>
    </urn:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <urn:create>
      <urn:sObjects xsi:type="urn1:Apttus_Proposal__Proposal__c">
        <urn1:Apttus_Proposal__Proposal_Name__c>Auto_API_Account</urn1:Apttus
_Proposal__Proposal_Name__c>
        <urn1:Apttus_Proposal__Account__c>001Z000001V13ovIAB</urn1:Apttus_Pro
posal__Account__c>
        <urn1:Apttus_Proposal__Primary__c>false</urn1:Apttus_Proposal__Primar
y__c>
        <urn1:Apttus_Proposal__ReadyToGenerate__c>true</urn1:Apttus_Proposal_
_ReadyToGenerate__c>
        <urn1:Apttus_Proposal__ReadyToPresent__c>true</urn1:Apttus_Proposal__
ReadyToPresent__c>
        <urn1:CurrencyIsoCode>USD</urn1:CurrencyIsoCode>
        <urn1:Apttus_Proposal__Opportunity__c>006Z000000Gr90D</urn1:Apttus_Pr
oposal__Opportunity__c>
        <urn1:Apttus_QPConfig__PriceListId__c>a1De0000001yPXQ</urn1:Apttus_QP
Config__PriceListId__c>
      </urn:sObjects>
    </urn:create>
  </soapenv:Body>
</soapenv:Envelope>

```

**Example Response**

```

<soapenv:Envelope

```



```

xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns="http://soap.sforce.com/schemas/class/Apttus_Proposal/ProposalWebService"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<soapenv:Body>
  <createProposalFromAccountResponse>
    <result xsi:type="Apttus_Proposal__Proposal__c">
      <Id xsi:nil="true"/>
      <Apttus_Proposal__Account__c>001Z000001V13ovIAB</
Apttus_Proposal__Account__c>
      <Apttus_Proposal__Primary__c>>false</Apttus_Proposal__Primary__c>
      <Apttus_Proposal__Proposal_Name__c>Auto_API_Account</
Apttus_Proposal__Proposal_Name__c>
      <Apttus_Proposal__ReadyToGenerate__c>>true</
Apttus_Proposal__ReadyToGenerate__c>
      <Apttus_Proposal__ReadyToPresent__c>>true</
Apttus_Proposal__ReadyToPresent__c>
      <CurrencyIsoCode>USD</CurrencyIsoCode>
      <OwnerId>0050U000000r444QAA</OwnerId>
    </result>
  </createProposalFromAccountResponse>
</soapenv:Body>
</soapenv:Envelope>

```

## Retrieving Proposal Field Values from Opportunity

This API fetches the field values of the proposal object that are required to create a proposal from an opportunity. You can use the fetched values to create a new proposal.

API	Signature
createProposalFromOpportunity	<i>webService static Apttus_Proposal__Proposal__c createProposalFromOpportunity(Id opptyId, Id recordTypeId)</i>

Request Parameter		
Name	Type	Description
opptyID	ID	The ID of the opportunity you need to retrieve values from.
recordTypeid	ID	Record Type of the proposal object

Response - Apttus_Proposal__Proposal__c		
Field	Type	Description
RecordTypeid	ID	Record Type of the proposal object
Apttus_Proposal__Proposal_Name__c	String	The name of the proposal.
Apttus_Proposal__Account__c	ID	The Id of the account associated with the proposal.
Apttus_Proposal__Opportunity__c	ID	The Id of the opportunity associated with the proposal.
Apttus_Proposal__Primary_Contact__c	ID	The primary contact associated with the proposal.
Apttus_Proposal__Description__c	String	The description of the proposal.
Apttus_Proposal__Primary__c	Boolean	Indicates whether the proposal is the primary quote to up
Apttus_Proposal__ReadyToGenerate__c	Boolean	Indicates whether the proposal is ready for generation.
Apttus_Proposal__ReadyToPresent__c	Boolean	Indicates whether the proposal is ready for presentation.
OwnerId	ID	The ID of the proposal object.
CurrencyIsoCode	String	The currency defined for the proposal.

## Code Sample

The sample code below enables you to retrieve the proposal field values of the opportunity associated with the ID that you provide. You can use the standard *createRecord* API to create the proposal.

```

1  /**
2   * The below method demonstrates how to fetch the values proposal field
   values from an existing opportunity
3   */
4   public static Apttus_Proposal__Proposal__c createProposal(Id
   opportunityID) {
5   Apttus_Proposal__Proposal__c newProposalSO = new
   Apttus_Proposal__Proposal__c();
6   ID proposalRecordTypeID = [select ID FROM recordType where
   sObjectType='Apttus_Proposal__Proposal__c' AND Name = 'Proposal' LIMIT 1].
   Id;
7   newProposalSO =
   Apttus_Proposal.ProposalWebService.createProposalFromOpportunity(opportuni
   tyID, proposalRecordTypeID);
8   return newProposalSO;
9   }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

- [Retrieve Proposal information](#)
- [Create Proposal record](#)

## API Prerequisites

None.

## Response/Request XML

### Example Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:prop="http://soap.sforce.com/schemas/class/Apttus_Proposal/ProposalWebService">
  <soapenv:Header>
    <prop:SessionHeader>
      <prop:sessionId>00DZ000000NAEIA!
ASAAQKosATSpsGeD7FUh4RDI18xnzIFPe80Mk89ejrAmDrCBY2lmyJKQjKvuwj3TvT71r6g_epvbo6FeqKUPf
mSookHTttOu</prop:sessionId>
    </prop:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <prop:createProposalFromOpportunity>
      <prop:opptyId>006Z000000Gr90D</prop:opptyId>
      <prop:recordTypeId>012i00000001ECb</prop:recordTypeId>
    </prop:createProposalFromOpportunity>
  </soapenv:Body>
</soapenv:Envelope>
```

### Example Response

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns="http://soap.sforce.com/schemas/class/Apttus_Proposal/ProposalWebService"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <createProposalFromOpportunityResponse>
      <result xsi:type="Apttus_Proposal__Proposal__c">
        <Id xsi:nil="true"/>
        <Apttus_Proposal__Account__c>001Z0000001V13ovIAB</
Apttus_Proposal__Account__c>
        <Apttus_Proposal__Opportunity__c>006Z000000Gr90DIAZ</
Apttus_Proposal__Opportunity__c>
        <Apttus_Proposal__Primary__c>>false</Apttus_Proposal__Primary__c>
        <Apttus_Proposal__Proposal_Name__c>Auto_API_Opportunity</
Apttus_Proposal__Proposal_Name__c>
```

```

        <Apttus_Proposal__ReadyToGenerate__c>true</
Apttus_Proposal__ReadyToGenerate__c>
        <Apttus_Proposal__ReadyToPresent__c>true</
Apttus_Proposal__ReadyToPresent__c>
        <CurrencyIsoCode>USD</CurrencyIsoCode>
        <OwnerId>0050U000000r444QAA</OwnerId>
        <RecordTypeId>012i0000001ECbAAM</RecordTypeId>
    </result>
</createProposalFromOpportunityResponse>
</soapenv:Body>
</soapenv:Envelope>

```

## Create Proposal

Use data from the Response to create the proposal. The following example uses the `createRecord()` method to create the Proposal object.

### Example Request

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:enterprise.soap.sforce.com"
  xmlns:urn1="urn:subject.enterprise.soap.sforce.com"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Header>
    <urn:SessionHeader>
      <urn:sessionId>00DZ000000NAEIA!
ASAAQKosATSpsGeD7FUh4RDI18xnzIFPe80Mk89ejrAmDrCBY2lmyJKQjKvuwj3TvT71r6g_epvbo6FeqKUPf
mSookHTtt0u</urn:sessionId>
    </urn:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <urn:create>
      <urn:sObjects xsi:type="urn1:Apttus_Proposal__Proposal__c">
        <urn1:Apttus_Proposal__Proposal_Name__c>Auto_API_Opportunity</urn1:Ap
ttus_Proposal__Proposal_Name__c>
        <urn1:Apttus_Proposal__Account__c>001Z000001V13ovIAB</urn1:Apttus_Pro
posal__Account__c>
        <urn1:Apttus_Proposal__Primary__c>false</urn1:Apttus_Proposal__Primar
y__c>
        <urn1:Apttus_Proposal__ReadyToGenerate__c>true</urn1:Apttus_Proposal_
_RReadyToGenerate__c>

```

```

        <urn1:Apttus_Proposal__ReadyToPresent__c>true</urn1:Apttus_Proposal__
ReadyToPresent__c>
        <urn1:CurrencyIsoCode>USD</urn1:CurrencyIsoCode>
        <urn1:Apttus_Proposal__Opportunity__c>006Z000000Gr90D</urn1:Apttus_Pr
oposal__Opportunity__c>
        <urn1:Apttus_QPConfig__PriceListId__c>a1De0000001yPXQ</urn1:Apttus_QP
Config__PriceListId__c>
        </urn:sObjects>
    </urn:create>
</soapenv:Body>
</soapenv:Envelope>

```

### Example Response

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_Proposal/ProposalWebService"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <createProposalFromOpportunityResponse>
      <result xsi:type="Apttus_Proposal__Proposal__c">
        <Id xsi:nil="true"/>
        <Apttus_Proposal__Account__c>001Z000001V13ovIAB</
Apttus_Proposal__Account__c>
        <Apttus_Proposal__Primary__c>>false</Apttus_Proposal__Primary__c>
        <Apttus_Proposal__Proposal_Name__c>Auto_API_Account</
Apttus_Proposal__Proposal_Name__c>
        <Apttus_Proposal__ReadyToGenerate__c>true</
Apttus_Proposal__ReadyToGenerate__c>
        <Apttus_Proposal__ReadyToPresent__c>true</
Apttus_Proposal__ReadyToPresent__c>
        <CurrencyIsoCode>USD</CurrencyIsoCode>
        <OwnerId>0050U000000r444QAA</OwnerId>
      </result>
    </createProposalFromOpportunityResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Retrieving Field Values from Proposal

This API fetches the field values of the proposal object that are required to clone from an existing proposal. You can use the fetched values to clone the proposal.

API	Signature
cloneProposal	<i>webservice static Apttus_Proposal__Proposal__c cloneProposal(Id originalId)</i>

Request Parameter		
Name	Type	Description
proposalId	ID	The ID of the proposal you want to clone.

Response - Apttus_Proposal__Proposal__c		
Field	Type	Description
RecordTypeId	ID	Record Type of the proposal object
Apttus_Proposal__Proposal_Name__c	String	The name of the proposal.
Apttus_Proposal__Account__c	ID	The Id of the account associated with the proposal.
Apttus_Proposal__Primary_Contact__c	ID	The primary contact associated with the proposal.
Apttus_Proposal__Description__c	String	The description of the proposal.
Apttus_Proposal__Primary__c	Boolean	Indicates whether the proposal is the primary quote to up

Response - Apttus_Proposal__Proposal__c		
Field	Type	Description
Apttus_Proposal__ReadyToGenerate__c	Boolean	Indicates whether the proposal is ready for generation.
Apttus_Proposal__ReadyToPresent__c	Boolean	Indicates whether the proposal is ready for presentation.
OwnerId	ID	The ID of the proposal object.
CurrencyIsoCode	String	The currency defined for the proposal.

### Code Sample

The sample code below enables you to retrieve the proposal field values from the proposal associated with the ID that you provide. You can use the standard *createRecord* API to create the new proposal.

```

1  /**
2   * The below code demonstrates how to fetch the fields values from an
3   * existing proposal
4   */
5  Apttus_Proposal__Proposal__c proposalS0 = new
6  Apttus_Proposal__Proposal__c();
7  proposalS0 =
8  Apttus_Proposal.ProposalWebService.cloneProposal(proposalName);
9  System.debug(proposalS0);

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

### API Prerequisites

None.



## Response/Request XML

### Example Request

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:prop="http://soap.sforce.com/schemas/class/Apttus_Proposal/
ProposalWebService">
  <soapenv:Header>
    <prop:SessionHeader>
      <prop:sessionId>00DZ000000NAEIA!
ASAAQKosATSpsGeD7FUh4RDI18xnzIFPe80Mk89ejrAmDrCBY2lmyJKQjKvuwj3TvT71r6g_epvbo6FeqKUPf
mSookHTtt0u</prop:sessionId>
    </prop:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <prop:cloneProposal>
      <prop:originalId>a0eZ0000005pF3U</prop:originalId>
    </prop:cloneProposal>
  </soapenv:Body>
</soapenv:Envelope>

```

### Example Response

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_Proposal/ProposalWebService"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <cloneProposalResponse>
      <result xsi:type="Apttus_Proposal__Proposal__c">
        <Id xsi:nil="true"/>
        <ABOTerminate__c>_HL_ENCODED_/apex/
Apttus_QPConfig__ProposalConfiguration?id=a0eZ0000005pF3U&flow=ABOTerminate_HL__IM1_/
resource/Apttus_QPConfig__Button_Configure_IM2_Configure_Products_IM3__HL__self_HL__</
ABOTerminate__c>
        <Apttus_Proposal__Account__c>001Z000001UeYiiIAF</
Apttus_Proposal__Account__c>
        <Apttus_Proposal__Amount__c>0.00</Apttus_Proposal__Amount__c>
      </result>
    </cloneProposalResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

```

        <Apttus_Proposal__Approval_Stage__c>Draft</
Apttus_Proposal__Approval_Stage__c>
        <Apttus_Proposal__Generate__c>_HL_ENCODED_/apex/
Apttus_Proposal__ProposalGenerate?id=a0eZ0000005pF3U_HL__IM1_/resource/
Apttus_Proposal__Button_Generate_IM2_Generate_IM3__HL__self_HL_</
Apttus_Proposal__Generate__c>
        <Apttus_Proposal__Grand_Total__c>0.00</
Apttus_Proposal__Grand_Total__c>
        <Apttus_Proposal__MakePrimary__c>_HL_ENCODED_/apex/
Apttus_Proposal__ProposalMakePrimary?id=a0eZ0000005pF3U_HL__IM1_/resource/
Apttus_Proposal__Button_MakePrimary_IM2_Make Primary_IM3__HL__self_HL_</
Apttus_Proposal__MakePrimary__c>
        <Apttus_Proposal__Net_Amount__c>0.00</Apttus_Proposal__Net_Amount__c>
        <Apttus_Proposal__Opportunity__c>006Z000000Gr9ODIAZ</
Apttus_Proposal__Opportunity__c>
        <Apttus_Proposal__Payment_Term__c>Net 30 Days</
Apttus_Proposal__Payment_Term__c>
        <Apttus_Proposal__Present__c>_HL_ENCODED_/apex/
Apttus_Proposal__ProposalPresent?id=a0eZ0000005pF3U_HL__IM1_/resource/
Apttus_Proposal__Button_Present_IM2_Present_IM3__HL__self_HL_</
Apttus_Proposal__Present__c>
        <Apttus_Proposal__Preview__c>_HL_ENCODED_/apex/
Apttus_Proposal__ProposalGenerate?id=a0eZ0000005pF3U&action=Preview_HL__IM1_/
resource/Apttus_Proposal__Button_Preview_IM2_Preview_IM3__HL__self_HL_</
Apttus_Proposal__Preview__c>
        <Apttus_Proposal__Primary__c>>false</Apttus_Proposal__Primary__c>
        <Apttus_Proposal__Proposal_Name__c>Test_CR003</
Apttus_Proposal__Proposal_Name__c>
        <Apttus_Proposal__ReadyToGenerate__c>>true</
Apttus_Proposal__ReadyToGenerate__c>
        <Apttus_Proposal__ReadyToPresent__c>>true</
Apttus_Proposal__ReadyToPresent__c>
        <Apttus_Proposal__Request_Approval__c>_HL_ENCODED_/servlet/
servlet.Integration?lid=01N700000009u2c&eid=a0eZ0000005pF3U_HL__IM1_/servlet/
servlet.ImageServer?
oid=00D7000000JXTN&id=01570000000e6nU_IM2_Submit_IM3__HL__blank_HL_</
Apttus_Proposal__Request_Approval__c>
        <Apttus_Proposal__Sales_Tax_Amount__c>0.00</
Apttus_Proposal__Sales_Tax_Amount__c>
        <Apttus_Proposal__SendProposal__c>_HL_ENCODED_/apex/
Apttus_Proposal__DocGen?id=a0eZ0000005pF3U&context=proposal&name=Q-01308003_HL__IM1_/
resource/Apttus_Proposal__Button_SendProposal_IM2_Send Proposal_IM3__HL__self_HL_</
Apttus_Proposal__SendProposal__c>
    
```

```

        <Apttus_QPAsset__ConfigureWithAssets__c>_HL_ENCODED_/apex/
Apttus_QPAsset__ProposalConfiguration?id=a0eZ0000005pF3U_HL__IM1_/resource/
Apttus_QPConfig__Button_Configure_IM2_Configure_Products_IM3__HL__self_HL_</
Apttus_QPAsset__ConfigureWithAssets__c>
        <Apttus_QPConfig__AutoActivateOrder__c>>false</
Apttus_QPConfig__AutoActivateOrder__c>
        <Apttus_QPConfig__AutoCreateBill__c>>false</
Apttus_QPConfig__AutoCreateBill__c>
        <Apttus_QPConfig__AutoCreateRevenue__c>>false</
Apttus_QPConfig__AutoCreateRevenue__c>
        <Apttus_QPConfig__BillingPreferenceId__c>a2ti0000000Fh9SAAS</
Apttus_QPConfig__BillingPreferenceId__c>
        <Apttus_QPConfig__ConfigureNG__c>_HL_ENCODED_/apex/
Apttus_QPConfig__ProposalConfiguration?id=a0eZ0000005pF3U&flow=NGDefault_HL__IM1_/
resource/Apttus_QPConfig__Button_Configure_IM2_Configure_Products_IM3__HL__self_HL_</
Apttus_QPConfig__ConfigureNG__c>
        <Apttus_QPConfig__Configure__c>_HL_ENCODED_/apex/
Apttus_QPConfig__ProposalConfiguration?id=a0eZ0000005pF3U&flow=Default_HL__IM1_/
resource/Apttus_QPConfig__Button_Configure_IM2_Configure_Products_IM3__HL__self_HL_</
Apttus_QPConfig__Configure__c>
        <Apttus_QPConfig__DisableCartVersioning__c>>false</
Apttus_QPConfig__DisableCartVersioning__c>
        <Apttus_QPConfig__IsAutoAccepted__c>>false</
Apttus_QPConfig__IsAutoAccepted__c>
        <Apttus_QPConfig__IsSystemGenerated__c>>false</
Apttus_QPConfig__IsSystemGenerated__c>
        <Apttus_QPConfig__IsTaskPending__c>>false</
Apttus_QPConfig__IsTaskPending__c>
        <Apttus_QPConfig__PriceListId__c>a1De0000001yPXQE2</
Apttus_QPConfig__PriceListId__c>
        <Apttus_QPConfig__SourceChannel__c>Direct</
Apttus_QPConfig__SourceChannel__c>
        <Apttus_QPConfig__SyncAssetChangesToQuote__c>>true</
Apttus_QPConfig__SyncAssetChangesToQuote__c>
        <Apttus_QPConfig__UseType__c>Main</Apttus_QPConfig__UseType__c>
        <AsyncFinalize_CartGrid__c>_HL_ENCODED_/apex/
Apttus_QPConfig__ProposalConfiguration?
id=a0eZ0000005pF3U&flow=AutoCartGrid&asyncFinalize=true_HL__IM1_/resource/
Apttus_QPConfig__Button_Configure_IM2_Configure_Products_IM3__HL__self_HL_</
Apttus_QPConfig__AsyncFinalize_CartGrid__c>
        <AutoCartGrid__c>_HL_ENCODED_/apex/
Apttus_QPConfig__ProposalConfiguration?id=a0eZ0000005pF3U&flow=AutoCartGrid_HL__IM1_/
resource/Apttus_QPConfig__Button_Configure_IM2_Configure_Products_IM3__HL__self_HL_</
AutoCartGrid__c>

```

```

        <AutoCartReadOnly__c>_HL_ENCODED_/apex/
Apttus_QPConfig__ProposalConfiguration?
id=a0eZ0000005pF3U&flow=AutoCartGrid&mode=readOnly_HL__IM1_/resource/
Apttus_QPConfig__Button_Configure_IM2_Configure_Products_IM3__HL__self_HL_</
AutoCartReadOnly__c>
        <CurrencyIsoCode>USD</CurrencyIsoCode>
        <IsDeleted>>false</IsDeleted>
        <Net_Amount__c>0.00</Net_Amount__c>
        <Numeric_Range_Field__c>1</Numeric_Range_Field__c>
        <OwnerId>0050U000000r444QAA</OwnerId>
        <Quote_Checkbox_21145__c>>false</Quote_Checkbox_21145__c>
        <RecordTypeId>012i00000001ECbAAM</RecordTypeId>
        <Region__c>India</Region__c>
        <SystemModstamp>2020-05-20T19:52:43.000Z</SystemModstamp>
    </result>
</cloneProposalResponse>
</soapenv:Body>
</soapenv:Envelope>

```

## Creating a Proposal

Use data from the Response to create the proposal. The following uses the `createRecord()` method to create the Proposal object.

### Example Request

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:urn="urn:partner.soap.sforce.com"
  xmlns:urn1="urn:subject.partner.soap.sforce.com">
  <soapenv:Header>
    <urn:SessionHeader>
      <urn:sessionId>00DZ000000NAEIA!
ASAAQKosATSpsGeD7FUh4RDI18xnzIFPe80Mk89ejrAmDrCBY2lmyJKQjKvuwj3TvT71r6g_epvbo6FeqKUPf
mSookHTtt0u</urn:sessionId>
    </urn:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <urn:create>
      <urn:sObjects>
        <urn1:type>Apttus_Proposal__Proposal__c</urn1:type>

```

```

        <urn1:Apttus_Proposal__Account__c>001Z000001UeYiiIAF</urn1:Apttus_Proposal__Account__c>
        <urn1:Apttus_Proposal__Approval_Stage__c>Draft</urn1:Apttus_Proposal__Approval_Stage__c>
        <urn1:Apttus_Proposal__Opportunity__c>006Z000000Gr90DIAZ</urn1:Apttus_Proposal__Opportunity__c>
        <urn1:Apttus_Proposal__Payment_Term__c>Net 30 Days</urn1:Apttus_Proposal__Payment_Term__c>
        <urn1:Apttus_Proposal__Primary__c>>false</urn1:Apttus_Proposal__Primary__c>
        <urn1:Apttus_Proposal__Proposal_Name__c>Test_CR003-copy2</urn1:Apttus_Proposal__Proposal_Name__c>
        <urn1:Apttus_Proposal__ReadyToGenerate__c>>true</urn1:Apttus_Proposal__ReadyToGenerate__c>
        <urn1:Apttus_Proposal__ReadyToPresent__c>>true</urn1:Apttus_Proposal__ReadyToPresent__c>
        <urn1:Apttus_QPConfig__AutoActivateOrder__c>>false</urn1:Apttus_QPConfig__AutoActivateOrder__c>
        <urn1:Apttus_QPConfig__AutoCreateBill__c>>false</urn1:Apttus_QPConfig__AutoCreateBill__c>
        <urn1:Apttus_QPConfig__AutoCreateRevenue__c>>false</urn1:Apttus_QPConfig__AutoCreateRevenue__c>
        <urn1:Apttus_QPConfig__BillingPreferenceId__c>a2ti0000000Fh9SAAS</urn1:Apttus_QPConfig__BillingPreferenceId__c>
        <urn1:Apttus_QPConfig__DisableCartVersioning__c>>false</urn1:Apttus_QPConfig__DisableCartVersioning__c>
        <urn1:Apttus_QPConfig__IsAutoAccepted__c>>false</urn1:Apttus_QPConfig__IsAutoAccepted__c>
        <urn1:Apttus_QPConfig__IsSystemGenerated__c>>false</urn1:Apttus_QPConfig__IsSystemGenerated__c>
        <urn1:Apttus_QPConfig__IsTaskPending__c>>false</urn1:Apttus_QPConfig__IsTaskPending__c>
        <urn1:Apttus_QPConfig__PriceListId__c>a1De0000001yPXQEA2</urn1:Apttus_QPConfig__PriceListId__c>
        <urn1:Apttus_QPConfig__SourceChannel__c>Direct</urn1:Apttus_QPConfig__SourceChannel__c>
        <urn1:Apttus_QPConfig__SyncAssetChangesToQuote__c>>true</urn1:Apttus_QPConfig__SyncAssetChangesToQuote__c>
        <urn1:Apttus_QPConfig__UseType__c>Main</urn1:Apttus_QPConfig__UseType__c>
        <urn1:CurrencyIsoCode>USD</urn1:CurrencyIsoCode>
        <urn1:Numeric_Range_Field__c>1</urn1:Numeric_Range_Field__c>
        <urn1:OwnerId>0050U000000r444QAA</urn1:OwnerId>
        <urn1:Quote_Checkbox_21145__c>>false</urn1:Quote_Checkbox_21145__c>

```

```

        <urn1:RecordTypeId>012i00000001ECbAAM</urn1:RecordTypeId>
        <urn1:Region__c>India</urn1:Region__c>
    </urn:sObjects>
</urn:create>
</soapenv:Body>
</soapenv:Envelope>
    
```

### Example Response

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="urn:partner.soap.sforce.com">
  <soapenv:Header>
  </soapenv:Header>
  <soapenv:Body>
    <createResponse>
      <result>
        <id>a0eZ00000005pFF1IAM</id>
        <success>>true</success>
      </result>
    </createResponse>
  </soapenv:Body>
</soapenv:Envelope>
    
```

## Creating Proposal Line Items from Opportunity

This API creates proposal line items from opportunity line items, creates a task for the action and invokes custom the callback which extends the interface *CustomClass.IQuoteLifecycleCallback2*. This API is a callback action after the proposal is created from **createProposalFromOpportunity** API.

API	Signature
afterCreateFromOpportunity	<i>webservice static Boolean afterCreateFromOpportunity(Id proposalId)</i>

Request Parameter		
Name	Type	Description
proposalId	ID	The ID of the existing proposal.

Response Parameters		
Field	Type	Description
isSuccess	Boolean	Indicates whether the callback was successfully executed. The values are <i>true</i> and <i>false</i> .

### Code Sample

The sample code below enables you to create proposal line items from the opportunity and create a new task for given proposalID.

```

1  /**
2   * The below code creates proposal line items from opportunity and create
3   * task for the proposal as callback actions.
4   * proposal from createProposalFromOpportunity API.
5   */
6  public Boolean afterCreateFromOpportunity(String proposalName) {
7      Boolean callBackStatus = false;
8      Apttus_Proposal__Proposal__c proposalS0 = [SELECT Id
9          FROM Apttus_Proposal__Proposal__c
10         WHERE Name =: proposalName LIMIT 1];
11     callBackStatus =
12     Apttus_Proposal.ProposalWebService.afterCreateFromOpportunity(proposalS0.I
13     d);
14     return callBackStatus;
15 }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

### API Prerequisites

- [Creating a Proposal from an Opportunity](#)

### Response/Request XML

#### Example Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:prop="http://soap.sforce.com/schemas/class/Apttus_Proposal/ProposalWebService">
  <soapenv:Header>
    <prop:SessionHeader>
      <prop:sessionId>00DZ000000NAEIA!
ASAAQKosATSpsGeD7FUh4RDI18xnzIFPe80Mk89ejrAmDrCBY2lmyJKQjKvuwj3TvT71r6g_epvbo6FeqKUPf
mSookHTttOu</prop:sessionId>
    </prop:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <prop:afterCreateFromOpportunity>
      <prop:proposalId>a0eZ0000005pF3U</prop:proposalId>
    </prop:afterCreateFromOpportunity>
  </soapenv:Body>
</soapenv:Envelope>
```

#### Example Response

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_Proposal/ProposalWebService">
  <soapenv:Body>
    <afterCreateFromOpportunityResponse>
      <result>true</result>
    </afterCreateFromOpportunityResponse>
  </soapenv:Body>
</soapenv:Envelope>
```



```

    </afterCreateFromOpportunityResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Updating a Proposal

This API updates proposal object field values.

API	Signature
updateProposal	<i>webservice static Boolean</i> <i>updateProposal(Apttus_Proposal__Proposal__c proposalSO)</i>

Request Parameter		
Name	Type	Description
proposalSO	Apttus_Proposal__Proposal__c	The proposal SObject

Response - Apttus_Proposal_Proposal_c		
Field	Type	Description
updateStatus	Boolean	Indicated whether the update was successful. The values returned are <i>true</i> and <i>false</i> .

### Code Sample

The sample below enables you to update the proposal object field values.

```

1  /**
2   * The below method demonstrates how to update description for already
   created proposal

```

```

3      */
4      public Boolean updateProposal(String proposalName)
5      {
6          Boolean updateStatus = false;
7          Apttus_Proposal__Proposal__c proposalS0 = [SELECT Id FROM
Apttus_Proposal__Proposal__c WHERE Name = :proposalName LIMIT 1];
8
9          // assign new value to the proposal fields one by one.
10         proposalS0.Apttus_Proposal__Description__c = 'Updated through
updateProposal API';
11         updateStatus =
Apttus_Proposal.ProposalWebService.updateProposal(proposalS0);
12         return updateStatus;
13     }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## Response/Request XML

### Example Request

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:prop="http://soap.sforce.com/schemas/class/Apttus_Proposal/
ProposalWebService">
  <soapenv:Header>
    <prop:SessionHeader>
      <prop:sessionId>00DZ000000NAEIA!
ASAAQBtZlu4.YeuaMdf45TpRUpKEgxgWi30zUbC0161kylPP_Rf0fungPPciWI0obZst4.nxzgnmtZ0s3hLjb
UCWziADgmbo</prop:sessionId>
    </prop:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <prop:updateProposal>
      <prop:proposalS0>
        <prop:Id>a0eZ0000005pG5N</prop:Id>

```

```

        <prop:Apttus_QPConfig__PriceListId__c>a1DZ0000002mg5n</prop:Apttus_QP
Config__PriceListId__c>
        <prop:Apttus_Proposal__Account__c>001Z000001UtoRL</prop:Apttus_Propos
al__Account__c>
        </prop:proposalSO>
    </prop:updateProposal>
</soapenv:Body>
</soapenv:Envelope>

```

### Example Response

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_Proposal/ProposalWebService">
  <soapenv:Body>
    <updateProposalResponse>
      <result>true</result>
    </updateProposalResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Cloning Email Template

Clones an email template.

### Prerequisite:

The email template you are cloning must exist in your org.

API	Signature
cloneEmailTemplate	<i>webService static Id cloneEmailTemplate(Id originalId)</i>

Request Parameter		
Name	Type	Description
originalId	ID	The ID of email template you want to clone.

Response - Apttus_Proposal_Proposal_c		
Field	Type	Description
newTemplateId	ID	The ID of the cloned email template.

### Code Sample

The sample code below enables you to clone an email template and get the ID of the cloned email template.

```

1  /**
2   * The below code demonstrates how to clone from existing email template.
3   */
4  Public Id cloneMailTemplate(Id originalID) {
5      Id newTemplateId =
6      Apttus_Proposal.ProposalWebService.cloneEmailTemplate(originalID);
7      return newTemplateId;
8  }
```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

### API Prerequisites

None.

## Response/Request XML

### Example Request

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:prop="http://soap.sforce.com/schemas/class/Apttus_Proposal/
ProposalWebService">
  <soapenv:Header>
    <prop:SessionHeader>
      <prop:sessionId>00DZ000000NAEIA!
ASAAQKosATSpsGeD7FUh4RDI18xnzIFPe80Mk89ejrAmDrCBY2lmyJKQjKvuwj3TvT71r6g_epvbo6FeqKUPf
mSookHTttOu</prop:sessionId>
    </prop:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <prop:cloneEmailTemplate>
      <prop:originalId>a09i000000TGec8</prop:originalId>
    </prop:cloneEmailTemplate>
  </soapenv:Body>
</soapenv:Envelope>
```

### Example Response

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_Proposal/ProposalWebService">
  <soapenv:Body>
    <cloneEmailTemplateResponse>
      <result>00XZ000000000333MAA</result>
    </cloneEmailTemplateResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

## Cloning Document Templates

This API clones all the attachments and files associated with the source email template and associate them with the newly cloned email template. This API is a callback API of *cloneEmailTemplate* API.

API	Signature
afterEmailTemplateClone	<i>webservice static Boolean afterEmailTemplateClone(Id originalId, Id clonedId)</i>

Request Parameter		
Name	Type	Description
originalId	ID	The ID of email template you want to clone.
clonedId	ID	The ID of the cloned proposal where you want to clone the attachments or files.

Response Parameter		
Field	Type	Description
isSuccess	Boolean	Indicates whether the cloning of the document template was successful. The values are <i>true</i> and <i>false</i> .

### Code Sample

The sample code below allows you to clone an email template and attachments or files from the parent email template and associate to the newly cloned email template.

```

1  /**
2   * The below code clone email template and the callback function of
3   * cloning the document template from the source
4   * and associate with the cloned email template. It returns boolean based
5   * on the status of after clone action.
6   */

```

```

5  public Boolean AfterCloneEmailTemplate(Id SrouceID) {
6      Boolean afterCloneAPIStatus = false;
7      Id clonedId =
Apttus_Proposal.ProposalWebService.cloneEmailTemplate(SrouceID);
8      afterCloneAPIStatus =
Apttus_Proposal.ProposalWebService.afterEmailTemplateClone(SrouceID,
clonedId);
9      return afterCloneAPIStatus;
10 }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## API Prerequisites

- [Cloning Email Template](#)

## Response/Request XML

### Example Request

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:prop="http://soap.sforce.com/schemas/class/Apttus_Proposal/
ProposalWebService">
  <soapenv:Header>
    <prop:SessionHeader>
      <prop:sessionId>00DZ000000NAEIA!
ASAAQKosATspsGeD7FUh4RDI18xnzIFPe80Mk89ejrAmDrCBY2lmyJKQjKvuwj3TvT71r6g_epvbo6FeqKUPf
mSookHTttOu</prop:sessionId>
    </prop:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <prop:afterEmailTemplateClone>
      <prop:originalId>a09i000000TGec8</prop:originalId>
      <prop:cloneId>a09Z000000E2y3l</prop:cloneId>
    </prop:afterEmailTemplateClone>
  </soapenv:Body>
</soapenv:Envelope>

```

```

Example Response

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_Proposal/ProposalWebService">
  <soapenv:Body>
    <afterEmailTemplateCloneResponse>
      <result>>true</result>
    </afterEmailTemplateCloneResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Getting the Proposal Document Output Format

This API retrieves the proposal document output format for a specific user profile.

API	Signature
getDefaultOutputFormat	<i>webservice static Apttus_Proposal__Proposal_Document_Output_Format__c getDefaultOutputFormat(Id profileId, Id recordTypeId)</i>

Request Parameter		
Name	Type	Description
profileId	ID	The ID of the profile.
recordTypeId	ID	Record Type of the proposal object



Response - Apttus_Proposal__Proposal__c		
Field	Type	Description
Id	ID	Record ID of the proposal.
Name	String	The name of the proposal
Apttus_Proposal__Profile_Id__c	ID	The Id of the profile.
Apttus_Proposal__Output_Format__c	String	The document output format (For example PDF, RTF or DOC or DOCX)
Apttus_Proposal__Include_Watermark__c	Boolean	Indicates whether a watermark is added to the document.
Apttus_Proposal__Allow_Override__c	Boolean	Indicates whether you are allowed to override the default output format during the quote/proposal document generation process.
Apttus_Proposal__AllowOverrideWatermark__c	Boolean	Indicates whether you are allowed to override the default watermark setting during the quote/proposal document generation process.
Apttus_Proposal__Proposal_Type_Id__c	String	The type of proposal
CurrencyIsoCode	String	The currency defined for the proposal.

### Code Sample

The sample code below enabled you to retrieve the proposal document output format for the given user profile and record type name.

```

1  /**
2   * The below code demonstrates how to get the document output format for
   * given profile name.
3   */
4  Public Static Id getProfileID (String profileName)

```

```

5  {
6      return [SELECT id FROM profile WHERE Name =: profileName LIMIT 1].Id;
7  }
8  public Id getProposalRecordTypeId(String proposalRecordTypeName)
9  {
10     return [SELECT ID
11             FROM recordType
12             WHERE sObjectType='Apttus_Proposal__Proposal__c'
13             AND Name =: proposalRecordTypeName
14             LIMIT 1].Id;
15 }
16 Public static Apttus_Proposal__Proposal_Document_Output_Format__c
   getDocumentOutputFormat (String profileName, String recordTypeName )
17 {
18     ID profileID = getProfileID(profileName);
19     ID proposalRecordTypeId = getProposalRecordTypeId(recordTypeName);
20     Apttus_Proposal__Proposal_Document_Output_Format__c outputFormatSO =
new Apttus_Proposal__Proposal_Document_Output_Format__c();
21     outputFormatSO =
   Apttus_Proposal.ProposalWebService.getDefaultOutputFormat();
22     return outputFormatSO;
23 }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## API Prerequisites

None.

## Response/Request XML

### Example Request

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"

```

```

    xmlns:prop="http://soap.sforce.com/schemas/class/Apttus_Proposal/
ProposalWebService">
  <soapenv:Header>
    <prop:SessionHeader>
      <prop:sessionId>00DZ000000NAEIA!
ASAAQKosATSpsGeD7FUh4RDI18xnzIFPe80Mk89ejrAmDrCBY2lmyJKQjKvuwj3TvT71r6g_epvbo6FeqKUPf
mSookHTtt0u</prop:sessionId>
    </prop:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <prop:getDefaultOutputFormat>
      <prop:profileId>00eZ0000000Uhn</prop:profileId>
      <prop:recordTypeId>012i00000001ECb</prop:recordTypeId>
    </prop:getDefaultOutputFormat>
  </soapenv:Body>
</soapenv:Envelope>

```

### Example Response

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="ht
tp://soap.sforce.com/schemas/class/Apttus_Proposal/ProposalWebService" xmlns:xsi="htt
p://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <getDefaultOutputFormatResponse>
      <result xsi:type="Apttus_Proposal__Proposal_Document_Output_Format__c">
        <Id>a0bZ00000008cMKnIAM</Id>
        <Apttus_Proposal__AllowOverrideWatermark__c>true</
Apttus_Proposal__AllowOverrideWatermark__c>
        <Apttus_Proposal__Allow_Override__c>true</
Apttus_Proposal__Allow_Override__c>
        <Apttus_Proposal__IncludeWatermark__c>true</
Apttus_Proposal__IncludeWatermark__c>
        <Apttus_Proposal__Output_Format__c>DOCX</
Apttus_Proposal__Output_Format__c>
        <Apttus_Proposal__Profile_Id__c>00eZ0000000UhnIAC</
Apttus_Proposal__Profile_Id__c>
        <Apttus_Proposal__Proposal_Type_Id__c>012i00000001ECbAAM</
Apttus_Proposal__Proposal_Type_Id__c>
        <Name>a0bZ00000008cMKn</Name>
      </result>
    </getDefaultOutputFormatResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

# Copying Attachments

This API copies an attachment to the destination record.

API	Signature
copyAttachment	<i>webService static Boolean copyAttachment(Id destParentId, Id attId)</i>

Request Parameter		
Name	Type	Description
destParentId	ID	The ID of the destination parent record.
attId	ID	The ID of the attachment you want to copy.

Response Parameter		
Field	Type	Description
isSuccess	Boolean	Indicates whether the copy was successful

## Code Sample

The sample code below enables you to clone an attachment from any parent record and assign it to the destination SObject record.

```
1 /**
```

```

2      * The below code demonstrates how to copy from existing attachment and
      associate with given parent ID.
3      */
4      public Boolean createAttachment (String attachmentName, String
      proposalName)
5      {
6          Boolean isSuccess;
7          Id proposalSOID = [SELECT Id FROM Apttus_Proposal__Proposal__c WHERE
      Name = :proposalName LIMIT 1].Id;
8          Id attachmentID = [SELECT Id FROM Attachment WHERE Name
      = :attachmentName LIMIT 1].Id;
9          isSuccess =
      Apttus_Proposal.ProposalWebService.copyAttachment(proposalSOID,
      attachmentID);
10         return isSuccess;
11     }

```

## Creating a Task After Proposal is Presented

This API changes the approval status of a proposal to *Presented* and creates a task.

API	Signature
afterPresent	<i>webService static Boolean afterPresent(Id proposalId)</i>

Request Parameter		
Name	Type	Description
proposalId	ID	The ID of the proposal

Response Parameter		
Field	Type	Description
isSuccess	Boolean	Indicates whether the approval status was changed and task creation was successful

### Code Sample

The sample code below enables you to change the approval status of a proposal to *Presented* and create a task for the given proposal.

```

1  /**
2   * The below code demonstrates the task creation after changing approval
3   * status to 'Presented'
4   */
5  public Boolean afterPresent(String proposalName)
6  {
7      Boolean isSuccess;
8      Id proposalSOID = [SELECT Id FROM Apttus_Proposal__Proposal__c WHERE
9      Name = :proposalName LIMIT 1].Id;
10     isSuccess =
11     Apttus_Proposal.ProposalWebService.afterPresent(proposalSOID);
12     return isSuccess;
13 }

```

## Retrieving Name of the First Template

The API retrieves the name of the first template associated with the given proposal.

API	Signature
<code>selectFirstTemplateNameForProposal</code>	<i>webService static String selectFirstTemplateNameForProposal(Id proposalId)</i>

Request Parameter		
Name	Type	Description
proposalId	ID	The ID of the proposal.

Response Parameter		
Field	Type	Description
templateName	String	The name of the first template.

### Code Sample

The sample code below enables you to get the first template name associated to the given proposal.

```

1  /**
2   * The below code gets first template name.
3   */
4  public String selectFirstTemplateNameForProposal(String proposalName)
5  {
6      String templateName;
7      Id proposalSOID = [SELECT Id FROM Apttus_Proposal__Proposal__c WHERE
8      Name = :proposalName LIMIT 1].Id;
9      templateName =
10     Apttus_Proposal.ProposalWebService.selectFirstTemplateNameForProposal(prop
11     osalSOID);
12     return templateName;
13 }

```

## Retrieving Templates in a Proposal

The API retrieves the templates associated with the given proposal.

API	Signature
getTemplatesForProposal	<i>webservice static List getTemplatesForProposal(Id proposalId, String proposalType)</i>

Request Parameter		
Name	Type	Description
proposalId	ID	The ID of the proposal.
proposalType	String	The type of template to query

Response Parameter		
Field	Type	Description
templateSOs	List<Apttus__APTS_Template__c>	The list of templates matching the configured query criteria.

### Code Sample

The sample code below enables you to retrieve the list of templates for the given proposal.

```

1  /**
2   * The below code demonstrates how to get list of template records for the
   * given proposal
3   */
4  public List<Apttus__APTS_Template__c> getTemplatesForProposal(String
   proposalName)

```



```

5  {
6      Id proposalSOID = [SELECT Id FROM Apttus_Proposal__Proposal__c WHERE
Name =: proposalName LIMIT 1].Id;
7      List<Apttus__APTS_Template__c> templateSOs = new
List<Apttus__APTS_Template__c>();
8      templateSOs =
Apttus_Proposal.ProposalWebService.getTemplatesForProposal(proposalSOID, 'P
roposal');
9      return templateSOs;
10 }

```

## Deleting an Email Template

The API deletes an email template.

API	Signature
deleteEmailTemplate	<i>webService static Boolean deleteEmailTemplate(Id templateId)</i>

Request Parameter		
Name	Type	Description
templateId	ID	The ID of the email template subject you want to delete.

Response Parameter		
Field	Type	Description
isDeleted	Boolean	Indicates whether the email template was deleted.

### Code Sample

The sample code below enables you to delete an email template and return the status of the delete action.

```

1  /**
2   * The below code demonstrates how to delete existing email template.
3   */
4  Public Boolean deleteMailTemplate(Id templateId)
5  {
6      Boolean isDeleted =
7      Apttus_Proposal.ProposalWebService.deleteEmailTemplate(templateId);
8      return isDeleted;
9  }

```

## Batch Update Service

The Batch Update Services enable you to update the single and multiple Category Views.

You can invoke APIs in Batch Update Service using the following command:

```
Apttus_CPQApi.BatchUpdateService.<Name of the Function>
where the name of the function is API Name and it's parameters.
```

In this section:

- [Updating Category Hierarchy for a single Category](#)
- [Updating Category Hierarchy for multiple Categories](#)

## Updating Category Hierarchy for a single Category

This method allows you to run the Category maintenance for a single category. This is applicable to only those categories which have products associated to them. You can invoke this API when a user clicks **Update View** button in the Category Hierarchy.

API	Signature
updateCategoryView	<i>static Id updateCategoryView(Id hierarchyId)</i>

Request Parameter		
Name	Type	Description
hierarchyId	ID	This is the ID of the category hierarchy record.

Response Parameter		
Name	Type	Description
apexBatchJobId	ID	The ID of the Batch Job run.

### Code sample

The code described below is used to run the Category Maintenance after you associate new products or modify the existing products under a category. You can also use this API when associate a Price List Item to a Category, under the related list, Price List Category. This method takes up the ID of the category hierarchy record and returns the Batch job ID after success.

```

1  public void updateCategoryView(String categoryName)
2  {
3      List<Apttus_Config2_ClassificationName_c> categoryList = [SELECT ID
FROM Apttus_Config2_ClassificationName_c WHERE Name =: categoryName];
4      String hierarchyId = categoryList[0].ID;
5      ID apexBatchJobId =
Apttus_CpqApi.BatchUpdateService.updateCategoryView(hierarchyId);
6      ApexPages.addMessage(new ApexPages.Message(ApexPages.severity.info,
'Apex Batch Job Id : ' + apexBatchJobId));
7
8  }
```

## Updating Category Hierarchy for multiple Categories

This method allows you to run the Category maintenance for multiple categories. This is applicable to only those categories which have products associated to them. You can invoke this API when a user clicks **Update View** button in the Category Hierarchy.

API		Signature
updateCategoryViews		<i>static Id updateCategoryViews(Set hierarchyIds)</i>
Request Parameter		
Name	Type	Description
hierarchyIds	Set<ID>	This is the set containing the IDs of the category hierarchy records.
Response Parameter		
Name	Type	Description
apexBatchJobId	ID	The ID of the Batch Job run.

### Code sample

The code described below is used to run the Category Maintenance for multiple Category Hierarchies after you associate new products or modify the existing products under multiple categories. You can also use this API when associate a Price List Item to a Category, under the related list, Price List Category. This method takes up the set of IDs of the category hierarchy records and returns the Batch job ID after success.

```

1  public void updateCategoryViews(List<String> categoryNames)
2  {
3      Set<ID> hierarchyIds = new Set<ID>();
4      for(String categoryName : categoryNames)
5      {
6          List<Apttus_Config2_ClassificationName_c> categoryList = [SELECT
7              ID FROM Apttus_Config2_ClassificationName_c WHERE Name =: categoryName];
8              hierarchyIds.add(categoryList[0].ID);
9      }
10     ID apexBatchJobId =
        Apttus_CpqApi.BatchUpdateService.updateCategoryViews(hierarchyIds);

```

```

11
12     ApexPages.addMessage(new ApexPages.Message(ApexPages.severity.info,
13     'Apex Batch Job Id : ' + apexBatchJobId));
    }

```

## CPQ Web Service(Apttus\_CPQApi)

You can invoke APIs in CPQ Web Service(Apttus\_CPQApi) from the following command:

```
Apttus_CPQApi.CPQWebService.<Name of the Function>
where the name of the function is API Name and it's parameters.
```

You can use the CPQ web service APIs to complete the following tasks:

- [Creating a Cart from a Quote](#)
- [Retrieving Categories for a Price List](#)
- [Retrieving Products and List Prices for a Price List](#)
- [Retrieving Products and List Prices for a Price List and Category](#)
- [Retrieving Products and List Prices For a Price List and Search Text](#)
- [Retrieving Products and List Prices For a Price List Category and Search Text](#)
- [Retrieving Option Groups, Options, and List Prices for a Price List Product](#)
- [Adding Products to a Cart](#)
- [Adding a Bundle to a Cart](#)
- [Adding Custom Bundles](#)
- [Adding Options to a Bundle](#)
- [Cloning Bundle Line Items on the Cart](#)
- [Cloning Line Items on the Cart](#)
- [Cloning Option Line Items on the Cart](#)
- [Comparing Products](#)
- [Associating Constraint Rules to a Cart](#)
- [Applying Constraint Rules to a Cart](#)
- [Applying Constraint Rules to Deleted Products](#)
- [Adding Price Ramps to a Cart \(CPQ Web Service\)](#)
- [Computing the Net Price for a Bundle](#)
- [Updating Price For A Cart](#)
- [Price Breakup for a Cart or Specific Line Item](#)
- [Removing a Bundle from a Cart](#)
- [Removing Multiple Bundles from a Cart](#)
- [Removing Line Items from the Cart](#)

- Removing Options from a Bundle
- Retrieving Constraint Rules Results
- Retrieving Incentives on the Cart
- Retrieving Asset Line Items
- Setting Incentives for Cart
- Retrieving Incentives Applied on the Cart
- Abandoning a Cart
- Finalizing a Cart
- Synchronizing a Cart
- Updating Quote Terms
- Modifying Assets (Deprecated)
- Computing Shipping for Cart Line Items
- Computing Taxes for Cart Line Items
- Adding a Miscellaneous Item to the Cart
- Creating Coupons for Incentives
- Retrieving Products Included by Auto-inclusion Constraint Rule
- Adding Line Items
- Rebuilds the Category Hierarchy

## Creating a Cart from a Quote

This API creates a cart for the quote or proposal referenced by QuotelD. The quote or proposal must be associated with a price list.

API	Signature
<code>createCart</code>	<code>webservice static Apttus_CPQApi.CPQ.CreateCartResponseDO createCart(Apttus_CPQApi.CPQ.CreateCartRequestDO request)</code>

Parameters			
Name	Type	Required?	Description
request	Apttus_CPQApi.CPQ.CreateCartRequestDO	Yes	The request data object.

Request Data Object - Apttus_CPQApi.CPQ.CreateCartRequestDO		
Field	Type	Description
Properties	List<Apttus_CPQApi.CPQ.PropertyDO>	The list of properties applicable to the cart
QuoteID	ID	The id of the quote/proposal to be associated with the cart.

Data Object - Apttus_CPQApi.CPQ.PropertyDO		
Field	Type	Description
Name	String	Specify the features applicable to the cart. Applicable values are: <ul style="list-style-type: none"> <li>• useAdvancedApproval: Enables Advanced Approval for a cart.</li> <li>• useDealOptimizer: Enables Deal Optimizer for a cart</li> </ul>
Value	String	The applicable values are true or false. Specifying the value as true enables the feature for a cart.

Response Data Object - Apttus_CPQApi.CPQ.CreateCartResponseDO		
Field	Type	Description
CartId	ID	The ID of the newly created cart object

### Code Sample

The sample below enables you to create a cart for a valid quote with a Quote ID. Using the sample below, you can search for a valid quote using a quote number. If a quote exists with the quote number entered, you can create a cart using the createCart API or you will be prompted with a message to enter a valid quote number. You can invoke this API in use cases when you want to show a cart page based on the quote. For example for a realized opportunity, you can create a quote. Based on a valid quote ID, you can create a cart using this API.

```

1  /**
2   * The below method demonstrates how to create a cart for a quote
3   */
4  public static void createCart(String quoteNumber)
5  {
6
7      List<Apttus_Proposal__Proposal__c> quote = [SELECT Id FROM
Apttus_Proposal__Proposal__c WHERE Name = :quoteNumber LIMIT 1];
8
9      if(!quote.isEmpty())
10     {
11
12         // Create the request object
13         Apttus_CPQApi.CPQ.CreateCartRequestDO request = new
Apttus_CPQApi.CPQ.CreateCartRequestDO();
14         List<Apttus_Config2.Property> Properties = new
List<Apttus_Config2.Property>();
15         Properties.add(new Apttus_Config2.Property('useAdvancedApproval', 'f
alse'));
16         Properties.add(new
Apttus_Config2.Property('isCartVersioningDisabled', 'true'));
17
18         request.QuoteId = quote.get(0).Id;

```



```

19     request.Properties = Properties;
20     // Excute the createCart routine
21     Apttus_CPQApi.CPQ.CreateCartResponseD0 response =
Apttus_CPQApi.CPQWebService.createCart(request);
22     System.debug('Cart has been successfully created. CartId = ' +
response.CartId);
23     }
24 }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## API Prerequisites

None.

## Response/Request XML

### Example Request

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:cpq="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:cpq1="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ">
  <soapenv:Header>
    <cpq:SessionHeader>
      <cpq:sessionId>00D4P000000z7dE!
AQQAQLOWqt9rvkoE67lP.NXIamuJl0mjo0R6a_l0vkPoATAH.gKFFRWManncKXN0KGY92akqlxsM0ze0lRfEr
g4V2LtI03XQ</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <cpq:createCart>
      <cpq:request>
        <cpq1:QuoteId>a0c4P00000GKQzL</cpq1:QuoteId>
      </cpq:request>
    </cpq:createCart>
  </soapenv:Body>
</soapenv:Envelope>

```

```

    </soapenv:Body>
</soapenv:Envelope>

```

### Example Response

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:CreateCartResponseDO="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ">
  <soapenv:Body>
    <createCartResponse>
      <result>
        <CreateCartResponseDO:CartId>a10Z00000002YK9rMAG</CreateCartResponseDO:
CartId>
      </result>
    </createCartResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Retrieving Categories for a Price List

This API is used to get the unique list of categories for all the products within the specified price list.

API	Signature
getCategoriesForPriceList	<pre> webService static Apttus_CPQApi.CPQ.CategorySearchResultDO getCategoriesForPriceList(Id priceListId) </pre>

Parameters			
Name	Type	Required?	Description
priceListId	ID	Yes	The id of the price list

Response Data Object - Apttus_CPQApi.CPQ.CategorySearchResultDO		
Field	Type	Description
Categories	List<Apttus_CPQApi.CPQ.CategoryDO>	The list of category data objects.
HasCategories	Boolean	Indicates if there are categories for the price list.

Data Object - Apttus_CPQApi.CPQ.CategoryDO		
Field	Type	Description
CategoryId	ID	The Id of the category.
ChildCategories	List<Apttus_CPQApi.CPQ.CategoryDO>	The list of child category data objects associated with the category.
HasChildCategories	Boolean	Indicates if the category has child categories.
Name	String	The category name.
ParentCategoryId	ID	The Id of the parent category.
ProductCount	Integer	The number of products within a category.

## Code Sample

The code sample below enables you to search for categories based on the pricelist ID. All the categories and subcategories associated with the price list are displayed. You can search for categories associated with a price list till the nth level. You can provide a search field on the cart page that enables the user to search for products by categories. For example, your cart page has multiple categories, such as Laptop, camera, Desktop, Accessories and so on. The user should be able to search for products by category by entering category name such as laptop. Use this API to invoke the categories and its associated sub-categories of products and then display it to the user.

```

1  public void executeSearch()
2  {
3  //Query for fetching pricelist id by querying price list name
4      List<Apttus_Config2__PriceList__c> priceListItemList = [select id
5          from Apttus_Config2__PriceList__c
6          where name = :priceListName limit 1];
7
8  //If an id is returned in the list execute the getCategoryForPricelist
9  API.
10     if(priceListItemList.size() > 0)
11     {
12         pricelistId = priceListItemList[0].ID;
13         Apttus_CPQApi.CPQ.CategorySearchResultDO result =
14         Apttus_CPQApi.CPQWebService.getCategoriesForPricelist(pricelistId);
15         lstwrap = New List<CategoryWrapperClass>();
16         For( Apttus_CPQApi.CPQ.CategoryDO catresult :
17         result.Categories)
18         {
19             CategoryWrapperClass wrap = New CategoryWrapperClass();
20             wrap.CategoryId = catresult.CategoryId;
21             wrap.categoryName = catresult.name;
22             lstwrap.add(wrap);
23             //If a category has sub categories fetch the sub-category
24             name and id
25             if(catresult.HasChildCategories)
26             {
27                 For( Apttus_CPQApi.CPQ.CategoryDO subcatresult :
28                 catresult.ChildCategories)
29                 {

```

```

25         CategoryWrapperClass subwrap = New
CategoryWrapperClass();
26         subwrap.CategoryId = subcatresult.CategoryId;
27         subwrap.categoryName = subcatresult.name;
28         lstwrap.add(subwrap);
29         // If the sub category has child categories fetch
the sub category name and ID.
30         if(subcatresult.HasChildCategories)
31         {
32             For( Apttus_CPQApi.CPQ.CategoryDO
subsubcatresult : subcatresult.ChildCategories)
33             {
34                 CategoryWrapperClass subsubwrap = New
CategoryWrapperClass();
35                 subsubwrap.CategoryId =
subsubcatresult.CategoryId;
36                 subsubwrap.categoryName =
subsubcatresult.name;
37                 lstwrap.add(subsubwrap);
38             }
39         }
40     }
41 }
42 }
43 }
44 //If no Price List exists with the searched string name execute the
else condition
45 else
46 {
47     lstwrap = New List<CategoryWrapperClass>();
48     ApexPages.addMessage(new
ApexPages.Message(ApexPages.severity.info, 'Category Name not found.
Please enter valid Category Name.));
49 }
50 }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## Response/Request XML

### Example Request

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:cpq="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService">
  <soapenv:Header>
    <cpq:SessionHeader>
      <cpq:sessionId>00DZ000000NAEIA!
ASAAQN7k3xhhvMe.j.8gpR.ijcGq77HJeF4MEDfbxbBAqZ8r4WWNTv30Vb6o1bjtHJLbq5mvHcuAH_ie6sTC3
DzrWgnLdUuD</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <cpq:getCategoriesForPriceList>
      <cpq:priceListId>a1DZ0000002mg5n</cpq:priceListId>
    </cpq:getCategoriesForPriceList>
  </soapenv:Body>
</soapenv:Envelope>
```

### Example Response

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:CategorySearchResultD0="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <getCategoriesForPriceListResponse>
      <result>
        <CategorySearchResultD0:Categories>
          <CategorySearchResultD0:CategoryId>a0nZ0000003x6pLIAQ</
CategorySearchResultD0:CategoryId>
          <CategorySearchResultD0:HasChildCategories>>false</
CategorySearchResultD0:HasChildCategories>
          <CategorySearchResultD0:Name>Auto_API_Category</
CategorySearchResultD0:Name>
```


```

        <CategorySearchResultDO:ParentCategoryId xsi:nil="true"/>
        <CategorySearchResultDO:ProductCount>15</CategorySearchResultDO:P
productCount>
        </CategorySearchResultDO:Categories>
        <CategorySearchResultDO:HasCategories>true</CategorySearchResultDO:Ha
sCategories>
    </result>
</getCategoriesForPriceListResponse>
</soapenv:Body>
</soapenv:Envelope>

```

## Retrieving Products and List Prices for a Price List

This API is used to get the list of all the products within the specified price list. This API automatically takes into consideration product visibility rules and will enforce them if applicable. For more information, see [Configuring Product Visibility](#).

 This API works only when you have associated a price list and price list items to a category. This API also considers the based on price list scenario.

API	Signature
getProductsForPriceList	<pre> webService static Apttus_CPQApi.CPQ.ProductSearchResultDO getProductsForPriceList(Id priceListId) </pre>

Parameters		
Name	Type	Description
priceListId	ID	The id of the price list.

Response Data Object - Apttus_CPQApi.CPQ.ProductSearchResultDO		
Field	Type	Description
HasProducts	Boolean	This returns true if the list of product data objects is not empty.
Products	List<Apttus_CPQApi.CPQ.ProductDO>	The list of product data objects.

Data Object - Apttus_CPQApi.CPQ.ProductDO		
Field	Type	Description
ContentUrl	String	The product content location.
Description	String	The product description.
HasPrices	Boolean	Indicates if there are list prices for the product.
HasProperties	Boolean	Indicates if there are properties set for the product.
ImageUrl	String	The location of the image, if there is one, associated with the product.
Name	String	The product name.
Prices	List<Apttus_CPQApi.CPQ.PriceDO>	The list of price data objects.
ProductCode	String	The product code.
ProductId	ID	The Id of the product.



Data Object - Apttus_CPQApi.CPQ.ProductDO		
Field	Type	Description
Property	List<Apttus_CPQApi.CPQ.PropertyDO>	The list of property data objects.

Data Object - Apttus_CPQApi.CPQ.PriceDO		
Field	Type	Description
ChargeType	String	The charge type.
Priceltem	Apttus_Config2__PriceListItem__c object	The price list items for a price list.
Value	Decimal	The list price.

Data Object - Apttus_CPQApi.CPQ.PropertyDO		
Field	Type	Description
Name	String	The name of the property.
Value	Decimal	The value of a property.

### Code Sample

Using the sample you can show the user the list of products for a particular price list. For example if you have a price list created for a particular company, the user can search the price list by name and view all the products associated with that price list. For example, when a user searches for badger price list and clicks search, invoke the `getProductsForPriceList` API in which you pass the `pricelistID` as a parameter. Fetch and then display all the product components to the user such as Machinery, standard price, quantity and name and description of the product.

```

1  public void getProductsForPriceList()
2  {
3      //If the priceList name by which the user searches the price list
exists fetch the ID.
4      if(priceListId == null || priceListId== '0')
5      {
6          List<Apttus_Config2__PriceList__c> priceListItemList = [select id
from Apttus_Config2__PriceList__c where Name = :priceListName limit 1];
7          //If the priceListItem exists, the list size >0 assign the ID at
the 0th position of the list to priceListId
8          if(priceListItemList.size() > 0)
9          {
10             priceListId = priceListItemList[0].ID;
11         }
12         else
13         {
14             lstProductwrapAll = New List<ProductWrapperClass>();
15             lstProductwrap = New List<ProductWrapperClass>();
16             return;
17         }
18     }
19
20     //Fetch id
21     Apttus_CPQApi.CPQ.ProductSearchResultDO productResult =
Apttus_CPQApi.CPQWebService.getProductsForPriceList(priceListId);
22     productCount = 'Product Count: ' + productResult.Products.size();
23     lstProductwrapAll = New List<ProductWrapperClass>();
24     lstProductwrap = New List<ProductWrapperClass>();
25
26     //For the fetched pricelistID fetch and display the following
27     For(Apttus_CPQApi.CPQ.ProductDO catresult : productResult.Products)
28     {
29         ProductWrapperClass wrap = New ProductWrapperClass ();
30         wrap.ProductId = catresult.ProductId;
31         wrap.ProductCode= catresult.ProductCode;
32         wrap.ProductName=catresult.Name;
33         wrap.Description=catresult.Description;
34         wrap.ImageUrl=catresult.ImageUrl;
35         wrap.ContentUrl=catresult.ContentUrl;
36         wrap.HasPrices=catresult.HasPrices;
37         wrap.Prices=catresult.Prices;
38         wrap.Quantity=1;
39         lstProductwrapAll.add(wrap);

```

```

40     }
41 }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## Response/Request XML

### Example Request

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cpq="
http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService">
  <soapenv:Header>
    <cpq:SessionHeader>
      <cpq:sessionId>00DZ000000NAEIA!
ASAAQN7k3xhhvMe.j.8gpR.ijcGq77HJeF4MEDfbxbBAqZ8r4WWNTv30Vb6o1bjtHJLbq5mvHcuAH_ie6sTC3
DzrWgnLdUuD</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <cpq:getProductsForPriceList>
      <cpq:priceListId>a1De0000001yPXQ</cpq:priceListId>
    </cpq:getProductsForPriceList>
  </soapenv:Body>
</soapenv:Envelope>

```

### Example Response

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:ProductSearchResultD0="http://soap.sforce.com/schemas/class/Apttus_CPQApi/
CPQ"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <getProductsForPriceListResponse>
      <result>

```

```

        <ProductSearchResultD0:HasProducts>true</ProductSearchResultD0:HasPro
ducts>
        <ProductSearchResultD0:Products>
            <ProductSearchResultD0:ContentUrl xsi:nil="true"/>
            <ProductSearchResultD0:Description xsi:nil="true"/>
            <ProductSearchResultD0:HasPrices>true</ProductSearchResultD0:HasP
rices>
            <ProductSearchResultD0:HasProperties>false</ProductSearchResultD0:
HasProperties>
            <ProductSearchResultD0:ImageUrl xsi:nil="true"/>
            <ProductSearchResultD0:Name>AutoQuoteBundle1</
ProductSearchResultD0:Name>
            <ProductSearchResultD0:Prices>
                <ProductSearchResultD0:ChargeType>Standard Price</
ProductSearchResultD0:ChargeType>
                <ProductSearchResultD0:PriceItem xsi:type="Apttus_Config2__Pr
iceListItem__c">
                    <Id>a1Ce0000003fQ1QEAU</Id>
                    <Apttus_Config2__Active__c>true</
Apttus_Config2__Active__c>
                    <Apttus_Config2__AllocateGroupAdjustment__c>true</
Apttus_Config2__AllocateGroupAdjustment__c>
                    <Apttus_Config2__AllowManualAdjustment__c>true</
Apttus_Config2__AllowManualAdjustment__c>
                    <Apttus_Config2__AllowProration__c>false</
Apttus_Config2__AllowProration__c>
                    <Apttus_Config2__ChargeType__c>Standard Price</
Apttus_Config2__ChargeType__c>
                    <Apttus_Config2__ListPrice__c>100.00000</
Apttus_Config2__ListPrice__c>
                    <Apttus_Config2__PriceIncludedInBundle__c>false</
Apttus_Config2__PriceIncludedInBundle__c>
                    <Apttus_Config2__PriceListId__c>a1De0000001yPXQE2</
Apttus_Config2__PriceListId__c>
                    <Apttus_Config2__PriceMethod__c>Per Unit</
Apttus_Config2__PriceMethod__c>
                    <Apttus_Config2__PriceType__c>One Time</
Apttus_Config2__PriceType__c>
                    <Apttus_Config2__ProductId__c>01te0000005NGpDAAW</
Apttus_Config2__ProductId__c>
                    <Apttus_Config2__ProductId__r xsi:type="Product2">
                        <Id>01te0000005NGpDAAW</Id>
                        <Name>AutoQuoteBundle1</Name>

```

```

        </Apttus_Config2__ProductId__r>
        <Name>PI-0000538767</Name>
    </ProductSearchResultD0:PriceItem>
    <ProductSearchResultD0:Value>100.00000</ProductSearchResultD0:
Value>

    </ProductSearchResultD0:Prices>
    <ProductSearchResultD0:ProductCode xsi:nil="true"/>
    <ProductSearchResultD0:ProductId>01te0000005NGpDAAW</
ProductSearchResultD0:ProductId>
    </ProductSearchResultD0:Products>
    <ProductSearchResultD0:Products>
        <ProductSearchResultD0:ContentUrl xsi:nil="true"/>
        <ProductSearchResultD0:Description xsi:nil="true"/>
        <ProductSearchResultD0:HasPrices>true</ProductSearchResultD0:HasP
rices>
        <ProductSearchResultD0:HasProperties>>false</ProductSearchResultD0:
HasProperties>
        <ProductSearchResultD0:ImageUrl xsi:nil="true"/>
        <ProductSearchResultD0:Name>AutoQuoteBundle2</
ProductSearchResultD0:Name>
        <ProductSearchResultD0:Prices>
            <ProductSearchResultD0:ChargeType>Standard Price</
ProductSearchResultD0:ChargeType>
            <ProductSearchResultD0:PriceItem xsi:type="Apttus_Config2__Pr
iceListItem__c">
                <Id>a1Ce0000003fQ2EEAU</Id>
                <Apttus_Config2__Active__c>true</
Apttus_Config2__Active__c>
                <Apttus_Config2__AllocateGroupAdjustment__c>true</
Apttus_Config2__AllocateGroupAdjustment__c>
                <Apttus_Config2__AllowManualAdjustment__c>true</
Apttus_Config2__AllowManualAdjustment__c>
                <Apttus_Config2__AllowProration__c>>false</
Apttus_Config2__AllowProration__c>
                <Apttus_Config2__ChargeType__c>Standard Price</
Apttus_Config2__ChargeType__c>
                <Apttus_Config2__ListPrice__c>100.00000</
Apttus_Config2__ListPrice__c>
                <Apttus_Config2__PriceIncludedInBundle__c>>false</
Apttus_Config2__PriceIncludedInBundle__c>
                <Apttus_Config2__PriceListId__c>a1De00000001yPXQE2</
Apttus_Config2__PriceListId__c>
                <Apttus_Config2__PriceMethod__c>Per Unit</
Apttus_Config2__PriceMethod__c>

```

```

        <Apttus_Config2__PriceType__c>One Time</
Apttus_Config2__PriceType__c>
        <Apttus_Config2__ProductId__c>01te0000005NGpNAAW</
Apttus_Config2__ProductId__c>
        <Apttus_Config2__ProductId__r xsi:type="Product2">
            <Id>01te0000005NGpNAAW</Id>
            <Name>AutoQuoteBundle2</Name>
        </Apttus_Config2__ProductId__r>
        <Name>PI-0000538772</Name>
    </ProductSearchResultDO:PriceItem>
    <ProductSearchResultDO:Value>100.00000</ProductSearchResultDO:
Value>
        </ProductSearchResultDO:Prices>
        <ProductSearchResultDO:ProductCode xsi:nil="true"/>
        <ProductSearchResultDO:ProductId>01te0000005NGpNAAW</
ProductSearchResultDO:ProductId>
        </ProductSearchResultDO:Products>
    </result>
</getProductsForPriceListResponse>
</soapenv:Body>
</soapenv:Envelope>

```

## Retrieving Products and List Prices for a Price List and Category

This API is used to get the list of products and related list prices for a specified category in the price list. This API automatically takes into consideration product visibility rules and will enforce them if applicable. For more information, see [Configuring Product Visibility](#).

API	Signature
getProductsForPriceListCategory	<pre> webService static Apttus_CPQApi.CPQ.ProductSearchResultDO getProductsForPriceListCategory(Id priceListId, Id categoryId) </pre>

Parameters		
Name	Type	Description
priceListId	ID	The id of the price list.
categoryID	ID	The id of the category.

Response Data Object - Apttus_CPQApi.CPQ.ProductSearchResultDO		
Field	Type	Description
HasProducts	Boolean	This returns true if the list of product data objects is not empty.
Products	List<Apttus_CPQApi.CPQ.ProductDO>	The list of product data objects.

Data Object - Apttus_CPQApi.CPQ.ProductDO		
Field	Type	Description
ContentUrl	String	The product content location.
Description	String	The product description.
HasPrices	Boolean	Indicates if there are list prices for the product.
HasProperties	Boolean	Indicates if there are properties set for the product.
ImageUrl	String	The location of the image, if there is one, associated with the product.

Data Object - Apttus_CPQApi.CPQ.ProductDO		
Field	Type	Description
Name	String	The product name.
Prices	List<Apttus_CPQApi.CPQ.PriceDO>	The list of price data objects.
ProductCode	String	The product code.
ProductId	ID	The Id of the product.
Property	List<Apttus_CPQApi.CPQ.PropertyDO>	The list of property data objects.

Data Object - Apttus_CPQApi.CPQ.PriceDO		
Field	Type	Description
ChargeType	String	The charge type.
Priceltem	Apttus_Config2__PriceListItem__c object	The price list items for a price list.
Value	Decimal	The list price.

Data Object - Apttus_CPQApi.CPQ.PropertyDO		
Field	Type	Description
Name	String	The name of the property.



Data Object - Apttus_CPQApi.CPQ.PropertyDO		
Field	Type	Description
Value	Decimal	The value of a property.

## Code Sample

The sample below enables you to search for categories using price list name. Once the user selects the category and proceeds to search, pass the pricelistID and categoryID as parameters to the API. The user can select and view the products associated with that category and its components. For example, user enters the price list name and selects the category-Hardware. All the products associated with the Hardware category, such as Laptop are displayed. You can display all the fields associated to that product.

```

1  public void getProductList()
2  {
3      categoryId = '';
4      for(CategoryWrapperClass wrap: lstwrap )
5      {
6          if(wrap.selected == true)
7          {
8              categoryId = categoryId + wrap.CategoryId + ',';
9          }
10     }
11     //If no category is selected prompt the user with an appropriate error
12     message.
13     if(categoryId.Trim()=='')
14     {
15         ApexPages.addMessage(new
16         ApexPages.Message(ApexPages.severity.info, 'Please select at least one
17         Category.));
18         lstProductwrap = New List<ProductWrapperClass>();
19         lstProductwrapAll = New List<ProductWrapperClass>();
20     }
21     //Pass the pricelistID and categoryID as parameters to the API
22     else
23     {
24         Apttus_CPQApi.CPQ.ProductSearchResultDO productResult =
25         Apttus_CPQApi.CPQWebService.getProductsForPriceListCategory(priceListId,ca
26         tegoryId.Substring(0,categoryId.length()-1));

```

```

23         productCount = 'Product Count: ' + productResult.Products.size();
24
25         lstProductwrap = New List<ProductWrapperClass>();
26         lstProductwrapAll = New List<ProductWrapperClass>();
27         For(Apttus_CPQApi.CPQ.ProductDO catresult :
productResult.Products)
28             //Fetch and display the following product fields to the user
29             {
30                 ProductWrapperClass wrap = New ProductWrapperClass ();
31                 wrap.ProductId = catresult.ProductId;
32                 wrap.ProductCode= catresult.ProductCode;
33                 wrap.ProductName=catresult.Name;
34                 wrap.Description=catresult.Description;
35                 wrap.ImageUrl=catresult.ImageUrl;
36                 wrap.ContentUrl=catresult.ContentUrl;
37                 wrap.HasPrices=catresult.HasPrices;
38                 wrap.Prices=catresult.Prices;
39                 wrap.Quantity=1;
40                 lstProductwrapAll.add(wrap);
41             }
42
43     }
44 }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## Response/Request XML

### Example Request

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:cpq="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService">
  <soapenv:Header>
    <cpq:SessionHeader>

```

```

    <cpq:sessionId>00DZ000000NAEIA!
ASAAQN7k3xhhvMe.j.8gpR.ijcGq77HJeF4MEDfbxbBAqZ8r4WWNTv30Vb6o1bjtHJLbq5mvHcuAH_ie6sTC3
DzrWgnLdUuD</cpq:sessionId>
    </cpq:SessionHeader>
</soapenv:Header>
<soapenv:Body>
    <cpq:getProductsForPriceListCategory>
        <cpq:priceListId>a1DZ0000002mg5n</cpq:priceListId>
        <cpq:categoryId>a0nZ0000003x6pL</cpq:categoryId>
    </cpq:getProductsForPriceListCategory>
</soapenv:Body>
</soapenv:Envelope>

```

### Example Response

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:ProductSearchResultDO="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <getProductsForPriceListCategoryResponse>
      <result>
        <ProductSearchResultDO:HasProducts>true</ProductSearchResultDO:HasProducts>
        <ProductSearchResultDO:Products>
          <ProductSearchResultDO:ContentUrl xsi:nil="true"/>
          <ProductSearchResultDO:Description xsi:nil="true"/>
          <ProductSearchResultDO:HasPrices>true</ProductSearchResultDO:HasPrices>
          <ProductSearchResultDO:HasProperties>>false</ProductSearchResultDO:HasProperties>
          <ProductSearchResultDO:ImageUrl xsi:nil="true"/>
          <ProductSearchResultDO:Name>Auto_API_BundleProduct1</ProductSearchResultDO:Name>
          <ProductSearchResultDO:Prices>
            <ProductSearchResultDO:ChargeType>Standard Price</ProductSearchResultDO:ChargeType>
            <ProductSearchResultDO:PriceItem xsi:type="Apttus_Config2__PriceListItem__c">
              <Id>a1CZ000000K0raoMAD</Id>

```

```

        <Apttus_Config2__Active__c>true</
Apttus_Config2__Active__c>
        <Apttus_Config2__AllocateGroupAdjustment__c>true</
Apttus_Config2__AllocateGroupAdjustment__c>
        <Apttus_Config2__AllowManualAdjustment__c>true</
Apttus_Config2__AllowManualAdjustment__c>
        <Apttus_Config2__AllowProration__c>>false</
Apttus_Config2__AllowProration__c>
        <Apttus_Config2__ChargeType__c>Standard Price</
Apttus_Config2__ChargeType__c>
        <Apttus_Config2__ListPrice__c>10.00000</
Apttus_Config2__ListPrice__c>
        <Apttus_Config2__PriceIncludedInBundle__c>>false</
Apttus_Config2__PriceIncludedInBundle__c>
        <Apttus_Config2__PriceListId__c>a1DZ0000002mg5nMAA</
Apttus_Config2__PriceListId__c>
        <Apttus_Config2__PriceMethod__c>Per Unit</
Apttus_Config2__PriceMethod__c>
        <Apttus_Config2__PriceType__c>One Time</
Apttus_Config2__PriceType__c>
        <Apttus_Config2__ProductId__c>01tZ0000004LUXtIAM</
Apttus_Config2__ProductId__c>
        <Apttus_Config2__ProductId__r xsi:type="Product2">
            <Id>01tZ0000004LUXtIAM</Id>
            <Name>Auto_API_BundleProduct1</Name>
        </Apttus_Config2__ProductId__r>
        <Name>PI-0000539885</Name>
    </ProductSearchResultD0:PriceItem>
    <ProductSearchResultD0:Value>10.00000</ProductSearchResultD0:
Value>
        </ProductSearchResultD0:Prices>
    <ProductSearchResultD0:ProductCode xsi:nil="true"/>
    <ProductSearchResultD0:ProductId>01tZ0000004LUXtIAM</
ProductSearchResultD0:ProductId>
    </ProductSearchResultD0:Products>
    <ProductSearchResultD0:Products>
        <ProductSearchResultD0:ContentUrl xsi:nil="true"/>
        <ProductSearchResultD0:Description xsi:nil="true"/>
        <ProductSearchResultD0:HasPrices>true</ProductSearchResultD0:HasP
rices>
        <ProductSearchResultD0:HasProperties>>false</ProductSearchResultD0:
HasProperties>
        <ProductSearchResultD0:ImageUrl xsi:nil="true"/>

```

```

        <ProductSearchResultD0:Name>Auto_API_BundleProduct2</
ProductSearchResultD0:Name>
        <ProductSearchResultD0:Prices>
            <ProductSearchResultD0:ChargeType>Standard Price</
ProductSearchResultD0:ChargeType>
            <ProductSearchResultD0:PriceItem xsi:type="Apttus_Config2__Pr
iceListItem__c">
                <Id>a1CZ000000K0rajMAD</Id>
                <Apttus_Config2__Active__c>true</
Apttus_Config2__Active__c>
                <Apttus_Config2__AllocateGroupAdjustment__c>true</
Apttus_Config2__AllocateGroupAdjustment__c>
                <Apttus_Config2__AllowManualAdjustment__c>true</
Apttus_Config2__AllowManualAdjustment__c>
                <Apttus_Config2__AllowProration__c>false</
Apttus_Config2__AllowProration__c>
                <Apttus_Config2__ChargeType__c>Standard Price</
Apttus_Config2__ChargeType__c>
                <Apttus_Config2__ListPrice__c>10.00000</
Apttus_Config2__ListPrice__c>
                <Apttus_Config2__PriceIncludedInBundle__c>false</
Apttus_Config2__PriceIncludedInBundle__c>
                <Apttus_Config2__PriceListId__c>a1DZ0000002mg5nMAA</
Apttus_Config2__PriceListId__c>
                <Apttus_Config2__PriceMethod__c>Per Unit</
Apttus_Config2__PriceMethod__c>
                <Apttus_Config2__PriceType__c>One Time</
Apttus_Config2__PriceType__c>
                <Apttus_Config2__ProductId__c>01tZ00000004LUXyIAM</
Apttus_Config2__ProductId__c>
                <Apttus_Config2__ProductId__r xsi:type="Product2">
                    <Id>01tZ00000004LUXyIAM</Id>
                    <Name>Auto_API_BundleProduct2</Name>
                </Apttus_Config2__ProductId__r>
                <Name>PI-0000539884</Name>
            </ProductSearchResultD0:PriceItem>
            <ProductSearchResultD0:Value>10.00000</ProductSearchResultD0:
Value>
        </ProductSearchResultD0:Prices>
        <ProductSearchResultD0:ProductCode xsi:nil="true"/>
        <ProductSearchResultD0:ProductId>01tZ00000004LUXyIAM</
ProductSearchResultD0:ProductId>
    </ProductSearchResultD0:Products>
</result>

```

```

        </getProductsForPriceListCategoryResponse>
    </soapenv:Body>
</soapenv:Envelope>
    
```

## Retrieving Products and List Prices For a Price List and Search Text

This API is used to get the list of products that match search text criteria you used for the products in the price list. This API automatically takes into consideration product visibility rules and will enforce them if applicable. For more information, see [Configuring Product Visibility](#).

API	Signature
<code>getProductsForSearchText</code>	<pre> webService static Apttus_CPQApi.CPQ.ProductSearchResultDO getProductsForSearchText(Id priceListId, String searchText)                     </pre>

Parameters		
Name	Type	Description
priceListId	ID	The id of the price list.
searchText	String	The search terms that will be used to retrieve the products for the price list.

Response Data Object - Apttus_CPQApi.CPQ.ProductSearchResultDO		
Field	Type	Description
HasProducts	Boolean	This returns true if the list of product data objects is not empty.
Products	List<Apttus_CPQApi.CPQ.ProductDO>	The list of product data objects.

Data Object - Apttus_CPQApi.CPQ.ProductDO		
Field	Type	Description
ContentUrl	String	The product content location.
Description	String	The product description.
HasPrices	Boolean	Indicates if there are list prices for the product.
HasProperties	Boolean	Indicates if there are properties set for the product.
ImageUrl	String	The location of the image, if there is one, associated with the product.
Name	String	The product name.
Prices	List<Apttus_CPQApi.CPQ.PriceDO>	The list of price data objects.
ProductCode	String	The product code.
ProductId	ID	The Id of the product.
Property	List<Apttus_CPQApi.CPQ.PropertyDO>	The list of property data objects.

Data Object - Apttus_CPQApi.CPQ.PriceDO		
Field	Type	Description
ChargeType	String	The charge type.
Priceltem	Apttus_Config2__PriceListItem__c	The price list items for a price list.
Value	Decimal	The list price.

Data Object - Apttus_CPQApi.CPQ.PropertyDO		
Field	Type	Description
Name	String	The name of the property.
Value	String	The value of a property.

### Code Sample

Using the sample below you can search for products using the price list id and product name search string. For example, you can provide to search fields for the user one for price list and the other for Search Text. Once the user enters the price list name, fetch the price list ID using the SOQL query. The user then types in the Product Name and clicks Search. Invoke the API where you pass the priceListID and the corresponding search string as parameters. The response object returns the product object fields. For example, if the user enters the name of the Hardware price list and the name of the product, such as Laptop and clicks Search, invoke the API and pass the priceListId and the product name as parameters of the API and fetch and display the product information.

```

1  class ProductWrapperClass
2  {
3      Id ProductId;
4      String ProductCode;
5      String ProductName;

```



```

6      String Description;
7      String imageUrl;
8      String ContentUrl;
9      Boolean HasPrices;
10     List<Apttus_CPQApi.CPQ.PriceDO> Prices;
11 }
12
13 public void getProductsForSearchText(Id priceListId, String productName) {
14
15
16     //Pass the pricelist ID and the Product Name entered in the search
17     text as parameters to the API
18
19     Apttus_CPQApi.CPQ.ProductSearchResultDO productResult =
20     Apttus_CPQApi.CPQWebService.getProductsForSearchText(priceListId,
21     productName);
22     String productCount = 'Product Count: ' +
23     productResult.Products.size();
24     List<ProductWrapperClass> lstProductwrapAll = New
25     List<ProductWrapperClass>();
26     For(Apttus_CPQApi.CPQ.ProductDO catresult : productResult.Products)
27
28     {
29         ProductWrapperClass wrap = New ProductWrapperClass();
30         wrap.ProductId = catresult.ProductId;
31         wrap.ProductCode= catresult.ProductCode;
32         wrap.ProductName=catresult.Name;
33         wrap.Description=catresult.Description;
34         wrap.ImageUrl=catresult.ImageUrl;
35         wrap.ContentUrl=catresult.ContentUrl;
36         wrap.HasPrices=catresult.HasPrices;
37         wrap.Prices=catresult.Prices;
38         lstProductwrapAll.add(wrap);
39     }
40     ApexPages.addMessage(new ApexPages.Message(ApexPages.severity.info,
41     'Product Count : ' + productCount));
42
43     ApexPages.addMessage(new ApexPages.Message(ApexPages.severity.info,
44     'Product Data Object : ' + lstProductwrapAll));
45 }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## Response/Request XML

### Example Request

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:cpq="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService">
  <soapenv:Header>
    <cpq:SessionHeader>
      <cpq:sessionId>00D4P000000z7dE!
AQQAQL6XYy.QsjqQQ6RTBnh.1ApTbiqkGAdVz9BS70lxobcyXgHHplmGXAE7p_cf6ziWJ8tpQt_4Q4Bi2VtY
eMyzjhaPbf0</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <cpq:getProductsForSearchText>
      <cpq:priceListId>a1a4P00000D3AVG</cpq:priceListId>
      <cpq:searchText>server</cpq:searchText>
    </cpq:getProductsForSearchText>
  </soapenv:Body>
</soapenv:Envelope>
```

### Example Response

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:ProductSearchResultDO="http://soap.sforce.com/schemas/class/Apttus_CPQApi/
CPQ"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <getProductsForSearchTextResponse>
      <result>
```

```

ducts>
    <ProductSearchResultD0:HasProducts>true</ProductSearchResultD0:HasPro
    <ProductSearchResultD0:Products>
      <ProductSearchResultD0:ContentUrl xsi:nil="true"/>
      <ProductSearchResultD0:Description xsi:nil="true"/>
      <ProductSearchResultD0:HasPrices>true</ProductSearchResultD0:HasP
rices>
      <ProductSearchResultD0:HasProperties>>false</ProductSearchResultD0:
HasProperties>
      <ProductSearchResultD0:ImageUrl xsi:nil="true"/>
      <ProductSearchResultD0:Name>BL460c Gen8 Server Blade</
ProductSearchResultD0:Name>
      <ProductSearchResultD0:Prices>
        <ProductSearchResultD0:ChargeType>Standard Price</
ProductSearchResultD0:ChargeType>
        <ProductSearchResultD0:PriceItem xsi:type="Apttus_Config2__Pr
iceListItem__c">
          <Id>a1Z4P00000EDIFjUAP</Id>
          <Apttus_Config2__Active__c>true</
Apttus_Config2__Active__c>
          <Apttus_Config2__AllocateGroupAdjustment__c>true</
Apttus_Config2__AllocateGroupAdjustment__c>
          <Apttus_Config2__AllowManualAdjustment__c>true</
Apttus_Config2__AllowManualAdjustment__c>
          <Apttus_Config2__AllowProration__c>>false</
Apttus_Config2__AllowProration__c>
          <Apttus_Config2__ChargeType__c>Standard Price</
Apttus_Config2__ChargeType__c>
          <Apttus_Config2__ListPrice__c>2939.00000</
Apttus_Config2__ListPrice__c>
          <Apttus_Config2__PriceIncludedInBundle__c>>false</
Apttus_Config2__PriceIncludedInBundle__c>
          <Apttus_Config2__PriceListId__c>a1a4P00000D3AVGQA3</
Apttus_Config2__PriceListId__c>
          <Apttus_Config2__PriceMethod__c>Per Unit</
Apttus_Config2__PriceMethod__c>
          <Apttus_Config2__PriceType__c>One Time</
Apttus_Config2__PriceType__c>
          <Apttus_Config2__PriceUom__c>Each</
Apttus_Config2__PriceUom__c>
          <Apttus_Config2__ProductId__c>01t4P0000080fJ0QAK</
Apttus_Config2__ProductId__c>
          <Apttus_Config2__ProductId__r xsi:type="Product2">
            <Id>01t4P0000080fJ0QAK</Id>

```

```

        <Family>Hardware</Family>
        <Name>BL460c Gen8 Server Blade</Name>
        <ProductCode>HW-BL002</ProductCode>
    </Apttus_Config2__ProductId__r>
    <Name>PI-0000000351</Name>
</ProductSearchResultD0:PriceItem>
<ProductSearchResultD0:Value>2939.00000</
ProductSearchResultD0:Value>
    </ProductSearchResultD0:Prices>
    <ProductSearchResultD0:ProductCode>HW-BL002</
ProductSearchResultD0:ProductCode>
    <ProductSearchResultD0:ProductId>01t4P0000080fJ0QAK</
ProductSearchResultD0:ProductId>
    </ProductSearchResultD0:Products>
    <ProductSearchResultD0:Products>
        <ProductSearchResultD0:ContentUrl xsi:nil="true"/>
        <ProductSearchResultD0:Description xsi:nil="true"/>
        <ProductSearchResultD0:HasPrices>true</ProductSearchResultD0:HasP
rices>
        <ProductSearchResultD0:HasProperties>>false</ProductSearchResultD0:
HasProperties>
        <ProductSearchResultD0:ImageUrl xsi:nil="true"/>
        <ProductSearchResultD0:Name>BL660c Gen8 Server Blade</
ProductSearchResultD0:Name>
        <ProductSearchResultD0:Prices>
            <ProductSearchResultD0:ChargeType>Standard Price</
ProductSearchResultD0:ChargeType>
            <ProductSearchResultD0:PriceItem xsi:type="Apttus_Config2__Pr
iceListItem__c">
                <Id>a1Z4P00000EDIGIUA5</Id>
                <Apttus_Config2__Active__c>true</
Apttus_Config2__Active__c>
                <Apttus_Config2__AllocateGroupAdjustment__c>true</
Apttus_Config2__AllocateGroupAdjustment__c>
                <Apttus_Config2__AllowManualAdjustment__c>true</
Apttus_Config2__AllowManualAdjustment__c>
                <Apttus_Config2__AllowProration__c>>false</
Apttus_Config2__AllowProration__c>
                <Apttus_Config2__ChargeType__c>Standard Price</
Apttus_Config2__ChargeType__c>
                <Apttus_Config2__ListPrice__c>7786.00000</
Apttus_Config2__ListPrice__c>

```

```

        <Apttus_Config2__PriceIncludedInBundle__c>>false</
Apttus_Config2__PriceIncludedInBundle__c>
        <Apttus_Config2__PriceListId__c>a1a4P00000D3AVGQA3</
Apttus_Config2__PriceListId__c>
        <Apttus_Config2__PriceMethod__c>Per Unit</
Apttus_Config2__PriceMethod__c>
        <Apttus_Config2__PriceType__c>One Time</
Apttus_Config2__PriceType__c>
        <Apttus_Config2__PriceUom__c>Each</
Apttus_Config2__PriceUom__c>
        <Apttus_Config2__ProductId__c>01t4P0000080fKsQAK</
Apttus_Config2__ProductId__c>
        <Apttus_Config2__ProductId__r xsi:type="Product2">
            <Id>01t4P0000080fKsQAK</Id>
            <Family>Hardware</Family>
            <Name>BL660c Gen8 Server Blade</Name>
            <ProductCode>HW-BL004</ProductCode>
        </Apttus_Config2__ProductId__r>
        <Name>PI-0000000386</Name>
    </ProductSearchResultDO:PriceItem>
    <ProductSearchResultDO:Value>7786.00000</
ProductSearchResultDO:Value>
    </ProductSearchResultDO:Prices>
    <ProductSearchResultDO:ProductCode>HW-BL004</
ProductSearchResultDO:ProductCode>
    <ProductSearchResultDO:ProductId>01t4P0000080fKsQAK</
ProductSearchResultDO:ProductId>
</ProductSearchResultDO:Products>
<ProductSearchResultDO:Products>
    <ProductSearchResultDO:ContentUrl xsi:nil="true"/>
    <ProductSearchResultDO:Description xsi:nil="true"/>
    <ProductSearchResultDO:HasPrices>true</ProductSearchResultDO:HasP
rices>
    <ProductSearchResultDO:HasProperties>>false</ProductSearchResultDO:
HasProperties>
    <ProductSearchResultDO:ImageUrl xsi:nil="true"/>
    <ProductSearchResultDO:Name>Cloud Server Solution</
ProductSearchResultDO:Name>
    <ProductSearchResultDO:Prices>
        <ProductSearchResultDO:ChargeType>Standard Price</
ProductSearchResultDO:ChargeType>
        <ProductSearchResultDO:PriceItem xsi:type="Apttus_Config2__Pr
iceListItem__c">
            <Id>a1Z4P00000EDIFUUA5</Id>

```

```

        <Apttus_Config2__Active__c>true</
Apttus_Config2__Active__c>
        <Apttus_Config2__AllocateGroupAdjustment__c>true</
Apttus_Config2__AllocateGroupAdjustment__c>
        <Apttus_Config2__AllowManualAdjustment__c>true</
Apttus_Config2__AllowManualAdjustment__c>
        <Apttus_Config2__AllowProration__c>>false</
Apttus_Config2__AllowProration__c>
        <Apttus_Config2__ChargeType__c>Standard Price</
Apttus_Config2__ChargeType__c>
        <Apttus_Config2__ListPrice__c>45999.00000</
Apttus_Config2__ListPrice__c>
        <Apttus_Config2__PriceIncludedInBundle__c>>false</
Apttus_Config2__PriceIncludedInBundle__c>
        <Apttus_Config2__PriceListId__c>a1a4P00000D3AVGQA3</
Apttus_Config2__PriceListId__c>
        <Apttus_Config2__PriceMethod__c>Per Unit</
Apttus_Config2__PriceMethod__c>
        <Apttus_Config2__PriceType__c>One Time</
Apttus_Config2__PriceType__c>
        <Apttus_Config2__PriceUom__c>Each</
Apttus_Config2__PriceUom__c>
        <Apttus_Config2__ProductId__c>01t4P0000080fJDQA0</
Apttus_Config2__ProductId__c>
        <Apttus_Config2__ProductId__r xsi:type="Product2">
            <Id>01t4P0000080fJDQA0</Id>
            <Name>Cloud Server Solution</Name>
            <ProductCode>CSS</ProductCode>
        </Apttus_Config2__ProductId__r>
        <Name>PI-0000000324</Name>
    </ProductSearchResultD0:PriceItem>
    <ProductSearchResultD0:Value>45999.00000</
ProductSearchResultD0:Value>
    </ProductSearchResultD0:Prices>
    <ProductSearchResultD0:ProductCode>CSS</ProductSearchResultD0:Pro
ductCode>
    <ProductSearchResultD0:ProductId>01t4P0000080fJDQA0</
ProductSearchResultD0:ProductId>
    </ProductSearchResultD0:Products>
    <ProductSearchResultD0:Products>
        <ProductSearchResultD0:ContentUrl xsi:nil="true"/>
        <ProductSearchResultD0:Description xsi:nil="true"/>

```

```

        <ProductSearchResultD0:HasPrices>true</ProductSearchResultD0:HasP
rices>
        <ProductSearchResultD0:HasProperties>>false</ProductSearchResultD0:
HasProperties>
        <ProductSearchResultD0:ImageUrl xsi:nil="true"/>
        <ProductSearchResultD0:Name>DL320e Gen8 Hot Plug 4LFF CTO
Server</ProductSearchResultD0:Name>
        <ProductSearchResultD0:Prices>
            <ProductSearchResultD0:ChargeType>Standard Price</
ProductSearchResultD0:ChargeType>
            <ProductSearchResultD0:PriceItem xsi:type="Apttus_Config2__Pr
iceListItem__c">
                <Id>a1Z4P00000EDIGYUA5</Id>
                <Apttus_Config2__Active__c>true</
Apttus_Config2__Active__c>
                <Apttus_Config2__AllocateGroupAdjustment__c>true</
Apttus_Config2__AllocateGroupAdjustment__c>
                <Apttus_Config2__AllowManualAdjustment__c>true</
Apttus_Config2__AllowManualAdjustment__c>
                <Apttus_Config2__AllowProration__c>>false</
Apttus_Config2__AllowProration__c>
                <Apttus_Config2__ChargeType__c>Standard Price</
Apttus_Config2__ChargeType__c>
                <Apttus_Config2__ListPrice__c>1056.00000</
Apttus_Config2__ListPrice__c>
                <Apttus_Config2__PriceIncludedInBundle__c>>false</
Apttus_Config2__PriceIncludedInBundle__c>
                <Apttus_Config2__PriceListId__c>a1a4P00000D3AVGQA3</
Apttus_Config2__PriceListId__c>
                <Apttus_Config2__PriceMethod__c>Per Unit</
Apttus_Config2__PriceMethod__c>
                <Apttus_Config2__PriceType__c>One Time</
Apttus_Config2__PriceType__c>
                <Apttus_Config2__PriceUom__c>Each</
Apttus_Config2__PriceUom__c>
                <Apttus_Config2__ProductId__c>01t4P0000080fL7QAK</
Apttus_Config2__ProductId__c>
                <Apttus_Config2__ProductId__r xsi:type="Product2">
                    <Id>01t4P0000080fL7QAK</Id>
                    <Family>Hardware</Family>
                    <Name>DL320e Gen8 Hot Plug 4LFF CTO Server</Name>
                    <ProductCode>HW-DL006</ProductCode>
                </Apttus_Config2__ProductId__r>
                <Name>PI-0000000402</Name>

```

```

        </ProductSearchResultD0:PriceItem>
        <ProductSearchResultD0:Value>1056.00000</
ProductSearchResultD0:Value>
        </ProductSearchResultD0:Prices>
        <ProductSearchResultD0:ProductCode>HW-DL006</
ProductSearchResultD0:ProductCode>
        <ProductSearchResultD0:ProductId>01t4P0000080fL7QAK</
ProductSearchResultD0:ProductId>
        </ProductSearchResultD0:Products>
        <ProductSearchResultD0:Products>
        <ProductSearchResultD0:ContentUrl xsi:nil="true"/>
        <ProductSearchResultD0:Description xsi:nil="true"/>
        <ProductSearchResultD0:HasPrices>true</ProductSearchResultD0:HasP
rices>
        <ProductSearchResultD0:HasProperties>>false</ProductSearchResultD0:
HasProperties>
        <ProductSearchResultD0:ImageUrl xsi:nil="true"/>
        <ProductSearchResultD0:Name>DL360p Gen8 10-SFF CTO Server</
ProductSearchResultD0:Name>
        <ProductSearchResultD0:Prices>
        <ProductSearchResultD0:ChargeType>Standard Price</
ProductSearchResultD0:ChargeType>
        <ProductSearchResultD0:PriceItem xsi:type="Apttus_Config2__Pr
iceListItem__c">
        <Id>a1Z4P00000EDIGFUA5</Id>
        <Apttus_Config2__Active__c>true</
Apttus_Config2__Active__c>
        <Apttus_Config2__AllocateGroupAdjustment__c>true</
Apttus_Config2__AllocateGroupAdjustment__c>
        <Apttus_Config2__AllowManualAdjustment__c>true</
Apttus_Config2__AllowManualAdjustment__c>
        <Apttus_Config2__AllowProration__c>false</
Apttus_Config2__AllowProration__c>
        <Apttus_Config2__ChargeType__c>Standard Price</
Apttus_Config2__ChargeType__c>
        <Apttus_Config2__ListPrice__c>4467.00000</
Apttus_Config2__ListPrice__c>
        <Apttus_Config2__PriceIncludedInBundle__c>false</
Apttus_Config2__PriceIncludedInBundle__c>
        <Apttus_Config2__PriceListId__c>a1a4P00000D3AVGQA3</
Apttus_Config2__PriceListId__c>
        <Apttus_Config2__PriceMethod__c>Per Unit</
Apttus_Config2__PriceMethod__c>

```



```

        <Apttus_Config2__PriceType__c>One Time</
Apttus_Config2__PriceType__c>
        <Apttus_Config2__PriceUom__c>Each</
Apttus_Config2__PriceUom__c>
        <Apttus_Config2__ProductId__c>01t4P0000080fKrQAK</
Apttus_Config2__ProductId__c>
        <Apttus_Config2__ProductId__r xsi:type="Product2">
            <Id>01t4P0000080fKrQAK</Id>
            <Family>Hardware</Family>
            <Name>DL360p Gen8 10-SFF CTO Server</Name>
            <ProductCode>HW-DL008</ProductCode>
        </Apttus_Config2__ProductId__r>
        <Name>PI-0000000383</Name>
    </ProductSearchResultDO:PriceItem>
    <ProductSearchResultDO:Value>4467.00000</
ProductSearchResultDO:Value>
        </ProductSearchResultDO:Prices>
    <ProductSearchResultDO:ProductCode>HW-DL008</
ProductSearchResultDO:ProductCode>
    <ProductSearchResultDO:ProductId>01t4P0000080fKrQAK</
ProductSearchResultDO:ProductId>
    </ProductSearchResultDO:Products>
    <ProductSearchResultDO:Products>
        <ProductSearchResultDO:ContentUrl xsi:nil="true"/>
        <ProductSearchResultDO:Description xsi:nil="true"/>
        <ProductSearchResultDO:HasPrices>true</ProductSearchResultDO:HasP
rices>
        <ProductSearchResultDO:HasProperties>false</ProductSearchResultDO:
HasProperties>
        <ProductSearchResultDO:ImageUrl xsi:nil="true"/>
        <ProductSearchResultDO:Name>DL380p Gen8 E-Mail or Messaging
Server Up to 1000 User Records</ProductSearchResultDO:Name>
        <ProductSearchResultDO:Prices>
            <ProductSearchResultDO:ChargeType>Standard Price</
ProductSearchResultDO:ChargeType>
            <ProductSearchResultDO:PriceItem xsi:type="Apttus_Config2__Pr
iceListItem__c">
                <Id>a1Z4P00000EDIG8UAP</Id>
                <Apttus_Config2__Active__c>true</
Apttus_Config2__Active__c>
                <Apttus_Config2__AllocateGroupAdjustment__c>true</
Apttus_Config2__AllocateGroupAdjustment__c>
                <Apttus_Config2__AllowManualAdjustment__c>true</
Apttus_Config2__AllowManualAdjustment__c>

```

```

        <Apttus_Config2__AllowProration__c>>false</
Apttus_Config2__AllowProration__c>
        <Apttus_Config2__ChargeType__c>Standard Price</
Apttus_Config2__ChargeType__c>
        <Apttus_Config2__ListPrice__c>8090.00000</
Apttus_Config2__ListPrice__c>
        <Apttus_Config2__PriceIncludedInBundle__c>>false</
Apttus_Config2__PriceIncludedInBundle__c>
        <Apttus_Config2__PriceListId__c>a1a4P00000D3AVGQA3</
Apttus_Config2__PriceListId__c>
        <Apttus_Config2__PriceMethod__c>Per Unit</
Apttus_Config2__PriceMethod__c>
        <Apttus_Config2__PriceType__c>One Time</
Apttus_Config2__PriceType__c>
        <Apttus_Config2__PriceUom__c>Each</
Apttus_Config2__PriceUom__c>
        <Apttus_Config2__ProductId__c>01t4P0000080fKLQAK</
Apttus_Config2__ProductId__c>
        <Apttus_Config2__ProductId__r xsi:type="Product2">
            <Id>01t4P0000080fKLQAK</Id>
            <Family>Hardware</Family>
            <Name>DL380p Gen8 E-Mail or Messaging Server Up to
1000 User Records</Name>
            <ProductCode>HW-DL011</ProductCode>
        </Apttus_Config2__ProductId__r>
        <Name>PI-0000000376</Name>
    </ProductSearchResultD0:PriceItem>
    <ProductSearchResultD0:Value>8090.00000</
ProductSearchResultD0:Value>
    </ProductSearchResultD0:Prices>
    <ProductSearchResultD0:ProductCode>HW-DL011</
ProductSearchResultD0:ProductCode>
    <ProductSearchResultD0:ProductId>01t4P0000080fKLQAK</
ProductSearchResultD0:ProductId>
    </ProductSearchResultD0:Products>
    <ProductSearchResultD0:Products>
        <ProductSearchResultD0:ContentUrl xsi:nil="true"/>
        <ProductSearchResultD0:Description xsi:nil="true"/>
        <ProductSearchResultD0:HasPrices>true</ProductSearchResultD0:HasP
rices>
        <ProductSearchResultD0:HasProperties>>false</ProductSearchResultD0:
HasProperties>
        <ProductSearchResultD0:ImageUrl xsi:nil="true"/>

```

```

        <ProductSearchResultD0:Name>DL385 Gen8 File and Print Server
Recommended</ProductSearchResultD0:Name>
        <ProductSearchResultD0:Prices>
            <ProductSearchResultD0:ChargeType>Standard Price</
ProductSearchResultD0:ChargeType>
            <ProductSearchResultD0:PriceItem xsi:type="Apttus_Config2__Pr
iceListItem__c">
                <Id>a1Z4P00000EDIFmUAP</Id>
                <Apttus_Config2__Active__c>true</
Apttus_Config2__Active__c>
                <Apttus_Config2__AllocateGroupAdjustment__c>true</
Apttus_Config2__AllocateGroupAdjustment__c>
                <Apttus_Config2__AllowManualAdjustment__c>true</
Apttus_Config2__AllowManualAdjustment__c>
                <Apttus_Config2__AllowProration__c>false</
Apttus_Config2__AllowProration__c>
                <Apttus_Config2__ChargeType__c>Standard Price</
Apttus_Config2__ChargeType__c>
                <Apttus_Config2__ListPrice__c>4631.00000</
Apttus_Config2__ListPrice__c>
                <Apttus_Config2__PriceIncludedInBundle__c>false</
Apttus_Config2__PriceIncludedInBundle__c>
                <Apttus_Config2__PriceListId__c>a1a4P00000D3AVGQA3</
Apttus_Config2__PriceListId__c>
                <Apttus_Config2__PriceMethod__c>Per Unit</
Apttus_Config2__PriceMethod__c>
                <Apttus_Config2__PriceType__c>One Time</
Apttus_Config2__PriceType__c>
                <Apttus_Config2__PriceUom__c>Each</
Apttus_Config2__PriceUom__c>
                <Apttus_Config2__ProductId__c>01t4P0000080fKPQA0</
Apttus_Config2__ProductId__c>
                <Apttus_Config2__ProductId__r xsi:type="Product2">
                    <Id>01t4P0000080fKPQA0</Id>
                    <Family>Hardware</Family>
                    <Name>DL385 Gen8 File and Print Server Recommended</
Name>
                    <ProductCode>HW-DL012</ProductCode>
                </Apttus_Config2__ProductId__r>
                <Name>PI-0000000354</Name>
            </ProductSearchResultD0:PriceItem>
            <ProductSearchResultD0:Value>4631.00000</
ProductSearchResultD0:Value>
        </ProductSearchResultD0:Prices>

```

```

        <ProductSearchResultD0:ProductCode>HW-DL012</
ProductSearchResultD0:ProductCode>
        <ProductSearchResultD0:ProductId>01t4P0000080fKPQA0</
ProductSearchResultD0:ProductId>
    </ProductSearchResultD0:Products>
    <ProductSearchResultD0:Products>
        <ProductSearchResultD0:ContentUrl xsi:nil="true"/>
        <ProductSearchResultD0:Description xsi:nil="true"/>
        <ProductSearchResultD0:HasPrices>true</ProductSearchResultD0:HasP
rices>
        <ProductSearchResultD0:HasProperties>>false</ProductSearchResultD0:
HasProperties>
        <ProductSearchResultD0:ImageUrl xsi:nil="true"/>
        <ProductSearchResultD0:Name>ML350e Gen8 Hot Plug 6LFF CTO
Server</ProductSearchResultD0:Name>
        <ProductSearchResultD0:Prices>
            <ProductSearchResultD0:ChargeType>Standard Price</
ProductSearchResultD0:ChargeType>
            <ProductSearchResultD0:PriceItem xsi:type="Apttus_Config2__Pr
iceListItem__c">
                <Id>a1Z4P00000EDIGHUA5</Id>
                <Apttus_Config2__Active__c>true</
Apttus_Config2__Active__c>
                <Apttus_Config2__AllocateGroupAdjustment__c>true</
Apttus_Config2__AllocateGroupAdjustment__c>
                <Apttus_Config2__AllowManualAdjustment__c>true</
Apttus_Config2__AllowManualAdjustment__c>
                <Apttus_Config2__AllowProration__c>false</
Apttus_Config2__AllowProration__c>
                <Apttus_Config2__ChargeType__c>Standard Price</
Apttus_Config2__ChargeType__c>
                <Apttus_Config2__ListPrice__c>2666.00000</
Apttus_Config2__ListPrice__c>
                <Apttus_Config2__PriceIncludedInBundle__c>false</
Apttus_Config2__PriceIncludedInBundle__c>
                <Apttus_Config2__PriceListId__c>a1a4P00000D3AVGQA3</
Apttus_Config2__PriceListId__c>
                <Apttus_Config2__PriceMethod__c>Per Unit</
Apttus_Config2__PriceMethod__c>
                <Apttus_Config2__PriceType__c>One Time</
Apttus_Config2__PriceType__c>
                <Apttus_Config2__PriceUom__c>Each</
Apttus_Config2__PriceUom__c>

```

```

        <Apttus_Config2__ProductId__c>01t4P0000080fJ9QAK</
Apttus_Config2__ProductId__c>
        <Apttus_Config2__ProductId__r xsi:type="Product2">
            <Id>01t4P0000080fJ9QAK</Id>
            <Family>Hardware</Family>
            <Name>ML350e Gen8 Hot Plug 6LFF CTO Server</Name>
            <ProductCode>HW-ML001</ProductCode>
        </Apttus_Config2__ProductId__r>
        <Name>PI-0000000385</Name>
    </ProductSearchResultDO:PriceItem>
    <ProductSearchResultDO:Value>2666.00000</
ProductSearchResultDO:Value>
    </ProductSearchResultDO:Prices>
    <ProductSearchResultDO:ProductCode>HW-ML001</
ProductSearchResultDO:ProductCode>
    <ProductSearchResultDO:ProductId>01t4P0000080fJ9QAK</
ProductSearchResultDO:ProductId>
    </ProductSearchResultDO:Products>
    <ProductSearchResultDO:Products>
        <ProductSearchResultDO:ContentUrl xsi:nil="true"/>
        <ProductSearchResultDO:Description xsi:nil="true"/>
        <ProductSearchResultDO:HasPrices>true</ProductSearchResultDO:HasP
rices>
        <ProductSearchResultDO:HasProperties>>false</ProductSearchResultDO:
HasProperties>
        <ProductSearchResultDO:ImageUrl xsi:nil="true"/>
        <ProductSearchResultDO:Name>WS460c Gen8 Graphics Server Blade</
ProductSearchResultDO:Name>
        <ProductSearchResultDO:Prices>
            <ProductSearchResultDO:ChargeType>Standard Price</
ProductSearchResultDO:ChargeType>
            <ProductSearchResultDO:PriceItem xsi:type="Apttus_Config2__Pr
iceListItem__c">
                <Id>a1Z4P00000EDIGBUA5</Id>
                <Apttus_Config2__Active__c>true</
Apttus_Config2__Active__c>
                <Apttus_Config2__AllocateGroupAdjustment__c>true</
Apttus_Config2__AllocateGroupAdjustment__c>
                <Apttus_Config2__AllowManualAdjustment__c>true</
Apttus_Config2__AllowManualAdjustment__c>
                <Apttus_Config2__AllowProration__c>>false</
Apttus_Config2__AllowProration__c>
                <Apttus_Config2__ChargeType__c>Standard Price</
Apttus_Config2__ChargeType__c>

```

```

        <Apttus_Config2__ListPrice__c>4954.00000</
Apttus_Config2__ListPrice__c>
        <Apttus_Config2__PriceIncludedInBundle__c>>false</
Apttus_Config2__PriceIncludedInBundle__c>
        <Apttus_Config2__PriceListId__c>a1a4P00000D3AVGQA3</
Apttus_Config2__PriceListId__c>
        <Apttus_Config2__PriceMethod__c>Per Unit</
Apttus_Config2__PriceMethod__c>
        <Apttus_Config2__PriceType__c>One Time</
Apttus_Config2__PriceType__c>
        <Apttus_Config2__PriceUom__c>Each</
Apttus_Config2__PriceUom__c>
        <Apttus_Config2__ProductId__c>01t4P0000080fJ8QAK</
Apttus_Config2__ProductId__c>
        <Apttus_Config2__ProductId__r xsi:type="Product2">
            <Id>01t4P0000080fJ8QAK</Id>
            <Family>Hardware</Family>
            <Name>WS460c Gen8 Graphics Server Blade</Name>
            <ProductCode>HW-BL003</ProductCode>
        </Apttus_Config2__ProductId__r>
        <Name>PI-0000000379</Name>
    </ProductSearchResultD0:PriceItem>
    <ProductSearchResultD0:Value>4954.00000</
ProductSearchResultD0:Value>
    </ProductSearchResultD0:Prices>
    <ProductSearchResultD0:ProductCode>HW-BL003</
ProductSearchResultD0:ProductCode>
    <ProductSearchResultD0:ProductId>01t4P0000080fJ8QAK</
ProductSearchResultD0:ProductId>
    </ProductSearchResultD0:Products>
</result>
</getProductsForSearchTextResponse>
</soapenv:Body>
</soapenv:Envelope>

```

## Retrieving Products and List Prices For a Price List Category and Search Text

This API is used to get the list of products that match search text criteria you used and belong to the specified category id in the price list. This API automatically takes into

consideration product visibility rules and will enforce them if applicable. For more information, see [Configuring Product Visibility](#).

API	Signature
<b>getProductsForCategorySearchText</b>	<i>WebService static Apttus_CPQApi.CPQ.ProductSearchResultDO getProductsForCategorySearchText(Id priceListId, Id categoryId, String searchText)</i>

Parameters		
Name	Type	Description
priceListId	ID	The id of the price list.
categoryId	ID	The id of the category.
searchText	String	The search terms that will be used to retrieve the products for the price list.

Response Data Object - Apttus_CPQApi.CPQ.ProductSearchResultDO		
Field	Type	Description
HasProducts	Boolean	This returns true if the list of product data objects is not empty.
Products	List<Apttus_CPQApi.CPQ.ProductDO>	The list of product data objects.

Data Object - Apttus_CPQApi.CPQ.ProductDO		
Field	Type	Description
ContentUrl	String	The product content location.
Description	String	The product description.
HasPrices	Boolean	Indicates if there are list prices for the product.
HasProperties	Boolean	Indicates if there are properties set for the product.
ImageUrl	String	The location of the image, if there is one, associated with the product.
Name	String	The product name.
Prices	List<Apttus_CPQApi.CPQ.PriceDO>	The list of price data objects.
ProductCode	String	The product code.
ProductId	ID	The Id of the product.
Property	List<Apttus_CPQApi.CPQ.PropertyDO>	The list of property data objects.
Data Object - Apttus_CPQApi.CPQ.PriceDO		
Field	Type	Description
ChargeType	String	The charge type.
PriceItem	Apttus_Config2__PriceListItem__c object	The price list items for a price list.
Value	Decimal	The list price.



Data Object - Apttus_CPQApi.CPQ.PropertyDO		
Field	Type	Description
Name	String	The name of the property.
Value	Decimal	The value of a property.

### Code Sample

The user can enter the name of price list to view all the categories belonging to that price list. The sample below defines the behavior when the user selects a category and searches a product from the categories displayed. Fetch and pass the pricelistID, categoryID, and productID as parameters to the API and display the fields of CPQ.ProductDo object. For example, the user enters a valid price list name, all the subsequent categories for the pricelist are displayed. The user selects the category Hardware and searches for a product Laptop. All the products matching the search string laptop are displayed along with the fields such as price, name , id and so on. You can execute the count loop to specify the number of products to be displayed per page and to provide next and previous pages.

```

1  public void getProductsForCategorySearchText()
2  {
3      //Search for single category multiple category using comma separator
4      //and fetch category id
5      categoryId = '';
6
7      for(CategoryWrapperClass wrap: lstwrap )
8      {
9          if(wrap.selected == true)
10         {
11             categoryId = categoryId + wrap.CategoryId + ',';
12         }
13     }
14     //If no category selected is selected by the user, show the following
15     //message
16     if(categoryId.Trim()=='')
17     {

```

```

16         ApexPages.addMessage(new
ApexPages.Message(ApexPages.severity.info, 'Please select at least one
Category.));
17         lstProductwrap = New List<ProductWrapperClass>();
18         lstProductwrapAll = New List<ProductWrapperClass>();
19     }
20     //If searched price list is valid, categories for that pricelist are
displayed and if the user selects a category and enters search text by
'productName' parameter, pass the PricelistID, categoryID, and productID
as parameters.
21     else
22     {
23         Apttus_CPQApi.CPQ.ProductSearchResultDO productResult =
Apttus_CPQApi.CPQWebService.getProductsForCategorySearchText(priceListId,c
ategoryId.Substring(0,categoryId.length()-1),productName);
24         productCount = 'Product Count: ' + productResult.Products.size();
25         lstProductwrap = New List<ProductWrapperClass>();
26         lstProductwrapAll = New List<ProductWrapperClass>();
27         For(Apttus_CPQApi.CPQ.ProductDO catresult :
productResult.Products)
28         {
29             // Fetch and display the following components for the products
30             ProductWrapperClass wrap = New ProductWrapperClass ();
31             wrap.ProductId = catresult.ProductId;
32             wrap.ProductCode= catresult.ProductCode;
33             wrap.ProductName=catresult.Name;
34             wrap.Description=catresult.Description;
35             wrap.ImageUrl=catresult.ImageUrl;
36             wrap.ContentUrl=catresult.ContentUrl;
37             wrap.HasPrices=catresult.HasPrices;
38             wrap.Quantity=1;
39             wrap.Prices=catresult.Prices;
40             lstProductwrapAll.add(wrap);
41         }
42     }
43 }
44 }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## Response/Request XML

### Example Request

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:cpq="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService">
  <soapenv:Header>
    <cpq:SessionHeader>
      <cpq:sessionId>00D4P000000z7dE!
AQQAQGL6XYy.QsjqQQ6RTBnh.1ApTbiqkGAdVz9BS70lxobcyXgHHpLmGXAE7p_cf6ziWJ8tpQt_4Q4Bi2VtY
eMyzjhaPbf0</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <cpq:getProductsForCategorySearchText>
      <cpq:priceListId>a1a4P00000D3AVG</cpq:priceListId>
      <cpq:categoryId>a0nZ0000003x6pL</cpq:categoryId>
      <cpq:searchText>server</cpq:searchText>
    </cpq:getProductsForCategorySearchText>
  </soapenv:Body>
</soapenv:Envelope>
```

### Example Response

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:ProductSearchResultD0="http://soap.sforce.com/schemas/class/Apttus_CPQApi/
CPQ"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <getProductsForCategorySearchTextResponse>
      <result>
        <ProductSearchResultD0:HasProducts>true</ProductSearchResultD0:HasPro
ducts>
        <ProductSearchResultD0:Products>
          <ProductSearchResultD0:ContentUrl xsi:nil="true"/>
          <ProductSearchResultD0:Description xsi:nil="true"/>
        </ProductSearchResultD0:Products>
      </result>
    </getProductsForCategorySearchTextResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

```

        <ProductSearchResultD0:HasPrices>true</ProductSearchResultD0:HasP
rices>
        <ProductSearchResultD0:HasProperties>>false</ProductSearchResultD0:
HasProperties>
        <ProductSearchResultD0:ImageUrl xsi:nil="true"/>
        <ProductSearchResultD0:Name>AutoQuoteStandalone1</
ProductSearchResultD0:Name>
        <ProductSearchResultD0:Prices>
            <ProductSearchResultD0:ChargeType>Standard Price</
ProductSearchResultD0:ChargeType>
            <ProductSearchResultD0:PriceItem xsi:type="Apttus_Config2__Pr
iceListItem__c">
                <Id>a1Ce0000003fQ0XEAU</Id>
                <Apttus_Config2__Active__c>true</
Apttus_Config2__Active__c>
                <Apttus_Config2__AllocateGroupAdjustment__c>true</
Apttus_Config2__AllocateGroupAdjustment__c>
                <Apttus_Config2__AllowManualAdjustment__c>true</
Apttus_Config2__AllowManualAdjustment__c>
                <Apttus_Config2__AllowProration__c>false</
Apttus_Config2__AllowProration__c>
                <Apttus_Config2__ChargeType__c>Standard Price</
Apttus_Config2__ChargeType__c>
                <Apttus_Config2__ListPrice__c>100.00000</
Apttus_Config2__ListPrice__c>
                <Apttus_Config2__PriceIncludedInBundle__c>false</
Apttus_Config2__PriceIncludedInBundle__c>
                <Apttus_Config2__PriceListId__c>a1De0000001yPXQE2</
Apttus_Config2__PriceListId__c>
                <Apttus_Config2__PriceMethod__c>Per Unit</
Apttus_Config2__PriceMethod__c>
                <Apttus_Config2__PriceType__c>One Time</
Apttus_Config2__PriceType__c>
                <Apttus_Config2__ProductId__c>01te0000005NGoUAAW</
Apttus_Config2__ProductId__c>
                <Apttus_Config2__ProductId__r xsi:type="Product2">
                    <Id>01te0000005NGoUAAW</Id>
                    <Name>AutoQuoteStandalone1</Name>
                </Apttus_Config2__ProductId__r>
                <Name>PI-0000538756</Name>
            </ProductSearchResultD0:PriceItem>
        <ProductSearchResultD0:Value>100.00000</ProductSearchResultD0:
Value>

```

```

        </ProductSearchResultD0:Prices>
        <ProductSearchResultD0:ProductCode xsi:nil="true"/>
        <ProductSearchResultD0:ProductId>01te0000005NGoUAAW</
ProductSearchResultD0:ProductId>
        </ProductSearchResultD0:Products>
    </result>
</getProductsForCategorySearchTextResponse>
</soapenv:Body>
</soapenv:Envelope>

```

## Retrieving Option Groups, Options, and List Prices for a Price List Product

This retrieves option groups, options, and list prices for a price list product.

API	Signature
getOptionGroupsForPriceListProduct	<pre> webService static Apttus_CPQApi.CPQ.ProductOptionGroupSearchResultD O getOptionGroupsForPriceListProduct(Id priceListId, Id productId) </pre>

Parameters		
Name	Type	Description
pricelistId	ID	The Id of the price list.
productId	ID	The Id of the product bundle.

Response Data Object - Apttus_CPQApi.CPQ.ProductOptionGroupSearchResultDO		
Name	Type	Description
HasOptionGroups	Boolean	Indicates if the bundle has option groups.
OptionGroups	List<Apttus_CPQApi.CPQ.ProductOptionGroupDO>	The list of product option group data objects.

Data Object - Apttus_CPQApi.CPQ.ProductOptionGroupDO		
Field	Type	Description
ChildOptionGroups	List<Apttus_CPQApi.CPQ.ProductOptionGroupDO>	List of child option group data objects.
HasChildOptionGroups	Boolean	Indicates whether there are child option groups.
HasOptionComponents	Boolean	Indicates if the option group has product components.
Label	String	The option group label.
MaxOptions	Integer	Maximum number of options that can be selected from the group.
MinOptions	Integer	Minimum number of options that must be selected from the group.
Name	String	The option group name.
OptionComponents	List<Apttus_CPQApi.CPQ.ProductOptionComponentDO>	List of product component data objects.
ParentOptiongroupId	ID	Id of the parent option group.

Data Object - Apttus_CPQApi.CPQ.ProductOptionGroupDO		
Field	Type	Description
ProductOptionGroupId	ID	Id of the product option group.

Data Object - Apttus_CPQApi.CPQ.ProductOptionComponentDO		
Field	Type	Description
ComponentId	ID	Id of the component.
ComponentProductId	ID	Id of the component product.
Description	String	The product description.
HasPrices	Boolean	Indicates if there are list prices for the product.
ImageUrl	String	The location of the image, if there is one, associated with the product.
IsDefault	Boolean	Indicates if the option is included by default for the bundle.
IsRequired	Boolean	Indicates if the option is mandatory for the bundle.
Name	String	The product name.
Prices	List<Apttus_CPQApi.CPQ.Price DO>	List of price data objects.
ProductCode	String	The product code.

Data Object - Apttus_CPQApi.CPQ.PriceDO		
Field	Type	Description
ChargeType	String	The charge type.
Priceltem	Apttus_Config2__PriceListItem__c object	The price list items for a price list.
Value	Decimal	The list price.

### Code Sample

Using the sample below you enable the user to search and select products by a price list. You can then pass the IDs of the price list and product as arguments to the API. If in the result object HasOptionGroups returns true, fetch and display the option group components. If the option group components HasComponents=true display the option components such as quantity. For example if the user searches a product and clicks Search, invoke the API to fetch the option groups for that bundle.

```

1  public void getOptionGroupsForPriceListProduct()
2  {
3      string productID='';
4      //Fetch product ID of selected products
5      for(ProductWrapperClass selProdWrap: lstProductwrap)
6      {
7          if(selProdWrap.selected)
8          {
9              productID=selProdWrap.ProductId;
10             break;
11         }
12     }
13
14     //If no product is selected, show an appropriate error message
15     if(productID=='')
16     {
17         ApexPages.addMessage(new
18         ApexPages.Message(ApexPages.severity.info, 'Please select at least one
19         Product. '));

```



```

18
19     lstOptionGroup = new
List<Apttus_CPQApi.CPQ.ProductOptionGroupDO>();
20     lstProductOptionWrapper = new List<ProductOptionWrapperClass>();
21 }
22 //If product is selected, pass pricelistID and productID as parameters
to the API
23 else
24 {
25     Apttus_CPQApi.CPQ.ProductOptionGroupSearchResultDO result =
Apttus_CPQApi.CPQWebService.getOptionGroupsForPriceListProduct(priceListId
, productID);
26     lstOptionGroup = new
List<Apttus_CPQApi.CPQ.ProductOptionGroupDO>();
27     lstProductOptionWrapper = new List<ProductOptionWrapperClass>();
28
29     //If the product has option groups, i,e the flag HasOptionGroups
returns true run the loop below fetching the components.
30     if(result.HasOptionGroups)
31     {
32         //For an option group fetch and display the following
components
33         For(Apttus_CPQApi.CPQ.ProductOptionGroupDO optionResult :
result.OptionGroups)
34         {
35             ProductOptionWrapperClass objProductOptionWrapperClass=new
ProductOptionWrapperClass();
36
objProductOptionWrapperClass.ProductOptionGroupId=optionResult.ProductOpti
onGroupId;
37             objProductOptionWrapperClass.Name=optionResult.Name;
38             objProductOptionWrapperClass.Label=optionResult.Label;
39
objProductOptionWrapperClass.ParentOptiongroupId=optionResult.ParentOpti
ongroupId;
40
objProductOptionWrapperClass.MinOptions=optionResult.MinOptions;
41
objProductOptionWrapperClass.MaxOptions=optionResult.MaxOptions;
42
objProductOptionWrapperClass.HasChildOptionGroups=optionResult.HasChildOpt
ionGroups;
43
objProductOptionWrapperClass.ChildOptionGroups=optionResult.ChildOptionGro
ups;

```

```

44         List<ProductQuantityOptionWrapperClass>
listOptionComponent= new List<ProductQuantityOptionWrapperClass>();
45
46         //For an option group with components such as min, max and
quantity fetch and display the editable components to the user
47         for(Apttus_CPQApi.CPQ.ProductOptionComponentDO
optionComponent : optionResult.OptionComponents)
48             {
49                 ProductQuantityOptionWrapperClass
objProductQuantityOptionWrapperClass = new
ProductQuantityOptionWrapperClass ();
50                 objProductQuantityOptionWrapperClass.OptionComponent =
optionComponent;
51                 objProductQuantityOptionWrapperClass.Quantity = 1;
52
53                 //Fetch and display the list price of the option
components
54                 List<Apttus_CPQApi.CPQ.PriceDO> priceList = new
List<Apttus_CPQApi.CPQ.PriceDO>();
55                 for(Apttus_CPQApi.CPQ.PriceDO price :
optionComponent.Prices)
56                     {
57                         priceList.Add(price);
58                     }
59                 objProductQuantityOptionWrapperClass.prices =
priceList ;
60
listOptionComponent.Add(objProductQuantityOptionWrapperClass);
61             }
62             objProductOptionWrapperClass.OptionComponents =
listOptionComponent;
63             lstOptionGroup.Add(optionResult);
64             lstProductOptionWrapper.Add(objProductOptionWrapperClass);
65         }
66
67         ApexPages.addMessage(new
ApexPages.Message(ApexPages.severity.info, 'Option Count: ' +
lstProductOptionWrapper.size()));
68     }
69 }
70 }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## Response/Request XML

### Example Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cpq="
http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService">
  <soapenv:Header>
    <cpq:SessionHeader>
      <cpq:sessionId>00D4P000000z7dE!
AQQAQL0Wqt9rvkoE67lP.NXIamuJl0mjo0R6a_l0vkPoATAH.gKFFRWMannckXN0KGY92akqlxsM0ze0lRfEr
g4V2LtI03XQ</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <cpq:getOptionGroupsForPriceListProduct>
      <cpq:priceListId>a1a4P00000D3AVG</cpq:priceListId>
      <cpq:productId>01t4P0000080fJ8</cpq:productId>
    </cpq:getOptionGroupsForPriceListProduct>
  </soapenv:Body>
</soapenv:Envelope>
```

### Example Response

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:ProductOptionGroupSearchResultD0="http://soap.sforce.com/schemas/class/
Apttus_CPQApi/CPQ"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <getOptionGroupsForPriceListProductResponse>
      <result>
        <ProductOptionGroupSearchResultD0:HasOptionGroups>true</
ProductOptionGroupSearchResultD0:HasOptionGroups>
```

```

        <ProductOptionGroupSearchResultD0:OptionGroups>
            <ProductOptionGroupSearchResultD0:HasChildOptionGroups>>false</
ProductOptionGroupSearchResultD0:HasChildOptionGroups>
            <ProductOptionGroupSearchResultD0:HasOptionComponents>>true</
ProductOptionGroupSearchResultD0:HasOptionComponents>
            <ProductOptionGroupSearchResultD0:Label>Enclosure & Rack</
ProductOptionGroupSearchResultD0:Label>
            <ProductOptionGroupSearchResultD0:MaxOptions>1</
ProductOptionGroupSearchResultD0:MaxOptions>
            <ProductOptionGroupSearchResultD0:MinOptions>1</
ProductOptionGroupSearchResultD0:MinOptions>
            <ProductOptionGroupSearchResultD0:Name>Enclosure & Rack</
ProductOptionGroupSearchResultD0:Name>
            <ProductOptionGroupSearchResultD0:OptionComponents>
                <ProductOptionGroupSearchResultD0:ComponentId>a1z4P000009c5Ra
QAI</ProductOptionGroupSearchResultD0:ComponentId>
                <ProductOptionGroupSearchResultD0:ComponentProductId>01t4P000
0080fLWQA0</ProductOptionGroupSearchResultD0:ComponentProductId>
                <ProductOptionGroupSearchResultD0:Description xsi:nil="true"/
>
                <ProductOptionGroupSearchResultD0:HasPrices>>true</
ProductOptionGroupSearchResultD0:HasPrices>
                <ProductOptionGroupSearchResultD0:ImageUrl xsi:nil="true"/>
                <ProductOptionGroupSearchResultD0:IsDefault>>true</
ProductOptionGroupSearchResultD0:IsDefault>
                <ProductOptionGroupSearchResultD0:IsRequired>>false</
ProductOptionGroupSearchResultD0:IsRequired>
                <ProductOptionGroupSearchResultD0:Name>c3000 Server
Enclosure</ProductOptionGroupSearchResultD0:Name>
                <ProductOptionGroupSearchResultD0:Prices>
                    <ProductOptionGroupSearchResultD0:ChargeType>Standard
Price</ProductOptionGroupSearchResultD0:ChargeType>
                    <ProductOptionGroupSearchResultD0:PriceItem xsi:type="Apt
tus_Config2__PriceListItem__c">
                        <Id>a1Z4P000000EDIH4UAP</Id>
                        <Apttus_Config2__Active__c>true</
Apttus_Config2__Active__c>
                        <Apttus_Config2__AllocateGroupAdjustment__c>true</
Apttus_Config2__AllocateGroupAdjustment__c>
                        <Apttus_Config2__AllowManualAdjustment__c>true</
Apttus_Config2__AllowManualAdjustment__c>
                        <Apttus_Config2__AllowProration__c>>false</
Apttus_Config2__AllowProration__c>

```

```

        <Apttus_Config2__ChargeType__c>Standard Price</
Apttus_Config2__ChargeType__c>
        <Apttus_Config2__ListPrice__c>199.00000</
Apttus_Config2__ListPrice__c>
        <Apttus_Config2__PriceIncludedInBundle__c>>false</
Apttus_Config2__PriceIncludedInBundle__c>
        <Apttus_Config2__PriceListId__c>a1a4P00000D3AVGQA3</
Apttus_Config2__PriceListId__c>
        <Apttus_Config2__PriceMethod__c>Per Unit</
Apttus_Config2__PriceMethod__c>
        <Apttus_Config2__PriceType__c>One Time</
Apttus_Config2__PriceType__c>
        <Apttus_Config2__PriceUom__c>Each</
Apttus_Config2__PriceUom__c>
        <Apttus_Config2__ProductId__c>01t4P0000080fLWQA0</
Apttus_Config2__ProductId__c>
        <Apttus_Config2__ProductId__r xsi:type="Product2">
            <Id>01t4P0000080fLWQA0</Id>
            <Family>Hardware</Family>
            <Name>c3000 Server Enclosure</Name>
            <ProductCode>HW-ER001</ProductCode>
        </Apttus_Config2__ProductId__r>
        <Name>PI-0000000435</Name>
        </ProductOptionGroupSearchResultD0:PriceItem>
        <ProductOptionGroupSearchResultD0:Value>199.00000</
ProductOptionGroupSearchResultD0:Value>
        </ProductOptionGroupSearchResultD0:Prices>
        <ProductOptionGroupSearchResultD0:ProductCode>HW-ER001</
ProductOptionGroupSearchResultD0:ProductCode>
        </ProductOptionGroupSearchResultD0:OptionComponents>
        <ProductOptionGroupSearchResultD0:OptionComponents>
            <ProductOptionGroupSearchResultD0:ComponentId>a1z4P000009c5Rb
QAI</ProductOptionGroupSearchResultD0:ComponentId>
            <ProductOptionGroupSearchResultD0:ComponentProductId>01t4P000
0080fKTQA0</ProductOptionGroupSearchResultD0:ComponentProductId>
            <ProductOptionGroupSearchResultD0:Description xsi:nil="true"/>
        >
            <ProductOptionGroupSearchResultD0:HasPrices>>true</
ProductOptionGroupSearchResultD0:HasPrices>
            <ProductOptionGroupSearchResultD0:ImageUrl xsi:nil="true"/>
            <ProductOptionGroupSearchResultD0:IsDefault>>false</
ProductOptionGroupSearchResultD0:IsDefault>
            <ProductOptionGroupSearchResultD0:IsRequired>>false</
ProductOptionGroupSearchResultD0:IsRequired>

```

```

        <ProductOptionGroupSearchResultD0:Name>c7000 Server
Enclosure</ProductOptionGroupSearchResultD0:Name>
        <ProductOptionGroupSearchResultD0:Prices>
            <ProductOptionGroupSearchResultD0:ChargeType>Standard
Price</ProductOptionGroupSearchResultD0:ChargeType>
            <ProductOptionGroupSearchResultD0:PriceItem xsi:type="Apt
tus_Config2__PriceListItem__c">
                <Id>a1Z4P00000EDIFqUAP</Id>
                <Apttus_Config2__Active__c>true</
Apttus_Config2__Active__c>
                <Apttus_Config2__AllocateGroupAdjustment__c>true</
Apttus_Config2__AllocateGroupAdjustment__c>
                <Apttus_Config2__AllowManualAdjustment__c>true</
Apttus_Config2__AllowManualAdjustment__c>
                <Apttus_Config2__AllowProration__c>false</
Apttus_Config2__AllowProration__c>
                <Apttus_Config2__ChargeType__c>Standard Price</
Apttus_Config2__ChargeType__c>
                <Apttus_Config2__ListPrice__c>299.00000</
Apttus_Config2__ListPrice__c>
                <Apttus_Config2__PriceIncludedInBundle__c>false</
Apttus_Config2__PriceIncludedInBundle__c>
                <Apttus_Config2__PriceListId__c>a1a4P00000D3AVGQA3</
Apttus_Config2__PriceListId__c>
                <Apttus_Config2__PriceMethod__c>Per Unit</
Apttus_Config2__PriceMethod__c>
                <Apttus_Config2__PriceType__c>One Time</
Apttus_Config2__PriceType__c>
                <Apttus_Config2__PriceUom__c>Each</
Apttus_Config2__PriceUom__c>
                <Apttus_Config2__ProductId__c>01t4P0000080fKTQA0</
Apttus_Config2__ProductId__c>
                <Apttus_Config2__ProductId__r xsi:type="Product2">
                    <Id>01t4P0000080fKTQA0</Id>
                    <Family>Hardware</Family>
                    <Name>c7000 Server Enclosure</Name>
                    <ProductCode>HW-ER002</ProductCode>
                </Apttus_Config2__ProductId__r>
                <Name>PI-0000000358</Name>
            </ProductOptionGroupSearchResultD0:PriceItem>
            <ProductOptionGroupSearchResultD0:Value>299.00000</
ProductOptionGroupSearchResultD0:Value>
        </ProductOptionGroupSearchResultD0:Prices>

```

```

        <ProductOptionGroupSearchResultDO:ProductCode>HW-ER002</
ProductOptionGroupSearchResultDO:ProductCode>
        </ProductOptionGroupSearchResultDO:OptionComponents>
        <ProductOptionGroupSearchResultDO:ParentOptionGroupId xsi:nil="tr
ue"/>
        <ProductOptionGroupSearchResultDO:ProductOptionGroupId>a154P00000
5liDzQAI</ProductOptionGroupSearchResultDO:ProductOptionGroupId>
        </ProductOptionGroupSearchResultDO:OptionGroups>
    </result>
</getOptionGroupsForPriceListProductResponse>
</soapenv:Body>
</soapenv:Envelope>

```

## Adding Products to a Cart

This API adds one or more products (with default options) to the cart along with quantity, term, start date, and end date.

API	Signature
<b>addMultiProducts</b>	<i>WebService static Apttus_CPQApi.CPQ.AddMultiProductResponseDO addMultiProducts(Apttus_CPQApi.CPQ.AddMultiProductRequestDO request)</i>

Parameters		
Name	Type	Description
request	Apttus_CPQApi.CPQ.AddMultiProductRequestDO	The request data object.

Request Data Object - Apttus_CPQApi.CPQ.AddMultiProductRequestDO		
Field	Type	Description
CartId	ID	The Id of the cart.
SelectedProducts	List <Apttus_CPQApi.CPQ.SelectedProductDO>	The list of selected product data objects.
Data Object - Apttus_CPQApi.CPQ.SelectedProductDO		
Field	Type	Description
AttributeValues	List	List of attributes values.
Comments	String	Comments associated with the record.
CopyBundleConfigurationFromSource	Boolean	You can use this to copy the configuration of the bundle from the source
CustomData	Apttus_Config2__LineItem__c	This can be used to include the list of custom fields you have added to the product.
CustomFields	List<String> CustomFields	List of custom fields created for your product.
EndDate	Date	The end date.
ProductId	ID	Id of the product bundle.
Quantity	Decimal	The bundle quantity.
RelatedLineItems	List	List of related line items for bundle
SellingTerm	Decimal	The bundle selling term.



Data Object - Apttus_CPQApi.CPQ.SelectedProductDO		
Field	Type	Description
SourceFields	List	List of the fields in the source bundle that you want to copy.
SourceId	ID	ID of the source bundle.
StartDate	Date	The start date. You should ensure you use the correct date format.

Response Data Object - Apttus_CPQApi.CPQ.AddMultiProductResponseDO		
Field	Type	Description
LineNumbers	List<Decimal>	The list of line numbers added to the cart.

### Code Sample

The sample below enables you to add multiple selected products with a specific product ID and its associated quantity, validity and selling term to a specific cart with a specific cartID. For example, if the user has selected software and hardware products, the products are added to the cart, on click of Add to Cart. The user is navigated to the cart page where they can view and change the quantity, selling term and other editable aspects of the product.

1	<code>/**</code>
2	<code> * The below method demonstrates how to add multiple products to an existing cart (every quote has a cart)</code>
3	<code> * Lets assume the Quote's Cart is blank and the standalone/bundle products are Laptop, Monitor, Wifi Router</code>
4	<code> * The input of this method is Quote Number and the Ids of the Laptop bundle product, Monitor and Wifi Router standalone products</code>
5	<code>*/</code>

```

6  public static void addMultipleProducts(String quoteNumber, List<ID>
productIds)
7  {
8
9      List<Apttus_Config2__ProductConfiguration__c> cart = [SELECT Id FROM
Apttus_Config2__ProductConfiguration__c WHERE
Apttus_QPConfig__ProposalId__r.Name = :quoteNumber LIMIT 1];
10
11     if(!cart.isEmpty() && productIds != null && !productIds.isEmpty())
12     {
13
14         // Assume the quantity and selling term for the product is 1
15         Integer quantity = 1;
16         Integer sellingTerm = 1;
17
18         // Create the request object
19         Apttus_CPQApi.CPQ.AddMultiProductRequestDO request = new
Apttus_CPQApi.CPQ.AddMultiProductRequestDO();
20         request.CartId = cart.get(0).Id;
21
22         // Add the products to the request
23         for(ID productId : productIds)
24         {
25             Apttus_CPQApi.CPQ.SelectedProductDO selectedProduct = new
Apttus_CPQApi.CPQ.SelectedProductDO();
26             selectedProduct.ProductId = productId;
27             selectedProduct.Quantity = quantity;
28             selectedProduct.SellingTerm = sellingTerm;
29             request.SelectedProducts.add(selectedProduct);
30         }
31
32         // Execute the addMultipleProducts routine
33         Apttus_CPQApi.CPQ.AddMultiProductResponseDO response =
Apttus_CPQApi.CPQWebService.addMultipleProducts(request);
34
35         System.debug('Line Numbers of added products = ' +
response.LineNumbers);
36     }
37 }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## API Prerequisites

- [Creating a Cart from a Quote](#)

## Request/Response XML

### Example Request

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:cpq="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:cpq1="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ">
  <soapenv:Header>
    <cpq:SessionHeader>
      <cpq:sessionId>00DZ000000NAEIA!
ASAAQHmIRqgn4R90i1yQjWTIVk4UZmsDe_.eK0Z9z6qLij7Tu.L_Yo8dRA_p80mhKMeRs4uzCZTadIgQ9fEbd
KciXEQYRyaA</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <cpq:addMultiProducts>
      <cpq:request>
        <cpq1:CartId>a10Z0000002YtUs</cpq1:CartId>
        <cpq1:SelectedProducts>
          <cpq1:Comments>Some comment</cpq1:Comments>
          <cpq1:ProductId>01tZ0000004LUXU</cpq1:ProductId>
          <cpq1:EndDate>2020-06-30</cpq1:EndDate>
          <cpq1:Quantity>5</cpq1:Quantity>
          <cpq1:SellingTerm>1</cpq1:SellingTerm>
          <cpq1:StartDate>2020-05-28</cpq1:StartDate>
        </cpq1:SelectedProducts>
        <cpq1:SelectedProducts>
          <cpq1:Comments>some comment</cpq1:Comments>
          <cpq1:ProductId>01tZ0000004LUXt</cpq1:ProductId>
          <cpq1:EndDate>2020-06-30</cpq1:EndDate>
```

```

        <cpq1:Quantity>1</cpq1:Quantity>
        <cpq1:SellingTerm>1</cpq1:SellingTerm>
        <cpq1:StartDate>2020-05-28</cpq1:StartDate>
    </cpq1:SelectedProducts>
    <cpq1:SelectedProducts>
        <cpq1:Comments>Some comment</cpq1:Comments>
        <cpq1:ProductId>01tZ0000004lUYI</cpq1:ProductId>
        <cpq1:EndDate>2020-06-30</cpq1:EndDate>
        <cpq1:Quantity>2</cpq1:Quantity>
        <cpq1:SellingTerm>5</cpq1:SellingTerm>
        <cpq1:StartDate>2020-05-28</cpq1:StartDate>
    </cpq1:SelectedProducts>
    </cpq:request>
</cpq:addMultiProducts>
</soapenv:Body>
</soapenv:Envelope>

```

### Example Response

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:AddMultiProductResponseD0="http://soap.sforce.com/schemas/class/
Apttus_CPQApi/CPQ">
  <soapenv:Body>
    <addMultiProductsResponse>
      <result>
        <AddMultiProductResponseD0:LineNumbers>1</AddMultiProductResponseD0:L
ineNumbers>
        <AddMultiProductResponseD0:LineNumbers>2</AddMultiProductResponseD0:L
ineNumbers>
        <AddMultiProductResponseD0:LineNumbers>3</AddMultiProductResponseD0:L
ineNumbers>
      </result>
    </addMultiProductsResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Adding a Bundle to a Cart

This adds a bundle along with selected products, options, and updated quantity to the cart.

API	Signature
<b>addBundle</b>	<i>webservice static Apttus_CPQApi.CPQ.AddBundleResponseDO addBundle(Apttus_CPQApi.CPQ.AddBundleRequestDO request)</i>

Parameters		
Name	Type	Description
request	Apttus_CPQApi.CPQ.AddBundleRequestDO	The id of the agreement

Request Data Object - Apttus_CPQApi.CPQ.AddBundleRequestDO		
Field	Type	Description
CartId	ID	The Id of the cart.
SelectedBundle	Apttus_CPQApi.CPQ.SelectedBundleDO	The bundle to be added to the cart.

Data Object - Apttus_CPQApi.CPQ.SelectedBundleDO		
Field	Type	Description
SelectedOptions	List<Apttus_CPQApi.CPQ.SelectedOptionDO>	The list of selected option data objects.
SelectedProduct	Apttus_CPQApi.CPQ.SelectedProductDO	The selected product data object.

Data Object - Apttus_CPQApi.CPQ.SelectedProductDO		
Field	Type	Description
AttributeValues	List	List of attributes values.
Comments	String	Comments associated with the record.
CopyBundleConfigurationFromSource	Boolean	You can use this to copy the configuration of the bundle from the source
CustomData	Apttus_Config2__LineItem_c	This can be used to include the list of custom fields you have added to the product.
CustomFields	List<String> CustomFields	List of custom fields created for your product.
EndDate	Date	The end date.
ProductId	ID	Id of the product bundle.
Quantity	Decimal	The bundle quantity.
RelatedLineItems	List	List of related line items for bundle

Data Object - Apttus_CPQApi.CPQ.SelectedProductDO		
Field	Type	Description
SellingTerm	Decimal	The bundle selling term.
SourceFields	List	List of the fields in the source bundle that you want to copy.
Sourceld	ID	ID of the source bundle.
StartDate	Date	The start date. You should ensure you use the correct date format.

Data Object - Apttus_CPQApi.CPQ.SelectedOptionDO		
Field	Type	Description
AttributeValues	List	List of attributes values.
Comments	String	Comments associated with the record.
ComponentId	ID	Id of the component.
ComponentProductId	ID	Id of the component product.
CustomData	Apttus_Config2__LineItem__c	The values to be set for the custom fields in the CustomFields List
CustomFields	List<String>	List of Custom Field's API Name
EndDate	Date	The end date.
Quantity	Decimal	The option quantity.
SellingTerm	Decimal	The option selling term.

Data Object - Apttus_CPQApi.CPQ.SelectedOptionDO		
Field	Type	Description
SourceId	ID	ID of the source option.
StartDate	Date	The start date. You should ensure you use the correct date format.

Response Data Object - Apttus_CPQApi.CPQ.AddBundleResponseDO		
Field	Type	Description
LineNumber	Decimal	The bundle line number.

### Code Sample

The sample below enables you to add a bundled product with a specific product ID and its associated quantity to a specific cart with a specific cartID. For example, a user selects a bundled product-Laptop+Mouse and clicks Add to Cart, the request to add bundle to the cart is invoked and the bundle with the specific product ID is added to a cart with the cart ID.

```

1  /**
2   * The below method demonstrates how to add a bundle product to an
3   * existing cart (every quote has a cart)
4   * Lets assume the Quote's Cart is blank and Laptop is a bundle product
5   * and its two options are Keyboard and Mouse
6   * The input of this method is Quote Number and the Id of the Laptop
7   * bundle product
8   * Inside the method we will add the Laptop bundle product and also its
9   * two options Keyboard and Mouse to the cart
10  */
11  public static void addBundle(String quoteNumber, ID bundleProductId)
12  {

```



```

9
10     List<Apttus_Config2__ProductConfiguration__c> cart = [SELECT
Apttus_Config2__PriceListId__c,Id FROM
Apttus_Config2__ProductConfiguration__c WHERE
Apttus_QPConfig__ProposalId__r.Name = :quoteNumber LIMIT 1];
11
12     if(!cart.isEmpty() && bundleProductId != null)
13     {
14
15         // Assume the quantity and selling term for the bundle product and
its options is 1
16         Integer quantity = 1;
17         Integer sellingTerm = 1;
18
19         // Create the request object
20         Apttus_CPQApi.CPQ.AddBundleRequestDO request = new
Apttus_CPQApi.CPQ.AddBundleRequestDO();
21         request.CartId = cart.get(0).Id;
22
23         // Add the bundle product to the request
24         request.SelectedBundle = new Apttus_CPQApi.CPQ.SelectedBundleDO();
25         request.SelectedBundle.SelectedProduct = new
Apttus_CPQApi.CPQ.SelectedProductDO();
26         request.SelectedBundle.SelectedProduct.ProductId =
bundleProductId;
27         request.SelectedBundle.SelectedProduct.Quantity = quantity;
28         request.SelectedBundle.SelectedProduct.SellingTerm = sellingTerm;
29
30         // Get all the options of the bundle product
31         Apttus_CPQApi.CPQ.ProductOptionGroupSearchResultDO
productOptionGroupResult =
Apttus_CPQApi.CPQWebService.getOptionGroupsForPriceListProduct(cart.get(0)
.Apttus_Config2__PriceListId__c, bundleProductId);
32         if(productOptionGroupResult.HasOptionGroups)
33         {
34             // Add the option products to the request
35             request.SelectedBundle.SelectedOptions = new
List<Apttus_CPQApi.CPQ.SelectedOptionDO>();
36             for(Apttus_CPQApi.CPQ.ProductOptionGroupDO
productOptionGroup : productOptionGroupResult.OptionGroups)
37             {
38                 if(productOptionGroup.HasOptionComponents)
39                 {
40                     for(Apttus_CPQApi.CPQ.ProductOptionComponentDO
productOptionComponent : productOptionGroup.OptionComponents)

```

```

41         {
42             Apttus_CPQApi.CPQ.SelectedOptionDO
selectedOptionDO = new Apttus_CPQApi.CPQ.SelectedOptionDO();
43             selectedOptionDO.ComponentId =
productOptionComponent.ComponentId;
44             selectedOptionDO.ComponentProductId =
productOptionComponent.ComponentProductId;
45             selectedOptionDO.Quantity = quantity;
46             selectedOptionDO.SellingTerm = sellingTerm;
47
request.SelectedBundle.SelectedOptions.add(selectedOptionDO);
48         }
49     }
50 }
51 }
52
53     // Execute the addBundle routine
54     Apttus_CPQApi.CPQ.AddBundleResponseDO response =
Apttus_CPQApi.CPQWebService.addBundle(request);
55
56     System.debug('Line Number of added bundle = ' +
response.LineNumber);
57 }
58 }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## API Prerequisites

- [Creating a Cart from a Quote](#)

## Response/Request XML

### Example Request

```
<soapenv:Envelope
```

```

xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:cpq="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
xmlns:cpq1="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ">
<soapenv:Header>
  <cpq:SessionHeader>
    <cpq:sessionId>00DZ000000NAEIA!
ASAAQFRdZcUFQ8LrHGF_qYNN9deP0ee.07JRqgeS3IF8IIlzILBdAOPySGSkif1VBtALP6pryVNOXfhn0faHO
pGsc9GFVLL0</cpq:sessionId>
  </cpq:SessionHeader>
</soapenv:Header>
<soapenv:Body>
  <cpq:addBundle>
    <cpq:request>
      <cpq1:CartId>a10Z0000002YdIH</cpq1:CartId>
      <cpq1:SelectedBundle>
        <cpq1:SelectedProduct>
          <cpq1:ProductId>01tZ0000004LUXt</cpq1:ProductId>
          <cpq1:Quantity>1</cpq1:Quantity>
          <cpq1:SellingTerm>1</cpq1:SellingTerm>
          <cpq1:StartDate>2020-05-14</cpq1:StartDate>
          <cpq1:EndDate>2020-05-15</cpq1:EndDate>
          <cpq1:Comments>None</cpq1:Comments>
        </cpq1:SelectedProduct>
        <cpq1:SelectedOptions>
          <cpq1:ComponentId>a1YZ0000002jPuS</cpq1:ComponentId>
          <cpq1:ComponentProductId>01tZ0000004u2io</cpq1:ComponentProdu
ctId>
          <cpq1:Quantity>1</cpq1:Quantity>
          <cpq1:SellingTerm>1</cpq1:SellingTerm>
          <cpq1:StartDate>2020-05-14</cpq1:StartDate>
          <cpq1:EndDate>2020-05-15</cpq1:EndDate>
          <cpq1:Comments>none</cpq1:Comments>
        </cpq1:SelectedOptions>
      </cpq1:SelectedBundle>
    </cpq:request>
  </cpq:addBundle>
</soapenv:Body>
</soapenv:Envelope>

```

### Example Response

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"

```

```

xmlns="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
xmlns:AddBundleResponseDO="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ">
  <soapenv:Body>
    <addBundleResponse>
      <result>
        <AddBundleResponseDO:LineNumber>8</AddBundleResponseDO:LineNumber>
      </result>
    </addBundleResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Adding Custom Bundles

This API adds custom bundles to the cart. You can define fields like **Quantity**, **Selling Term**, **Start Date**, and **End Date** for the bundle. You can add more than one bundle. The sequencing of the products added to the cart is based on the order in which you add the line items into the request DataObject list. If you have more than one option associated with the bundle, the option sequence would happen based on the sequence we add them to the DataObject list.

API	Signature
addCustomBundle	<i>WebService static Apttus_CPQApi.CPQ.AddCustomBundleResponseDO addCustomBundle(Apttus_CPQApi.CPQ.AddCustomBundleRequestDO request)</i>

Request Parameter		
Name	Type	Description
request	Apttus_CPQApi.CPQ.AddCustomBundleRequestDO	The request data object

Request Data Object - Apttus_CPQApi.CPQ.AddCustomBundleRequestDO		
Field	Type	Description
CartId	ID	The Id of the cart.
LineItemColl	Apttus_CPQApi.CPQ.LineItemCollDO O	The Line Item data object

Data Object - Apttus_CPQApi.CPQ.LineItemCollDO		
Field	Type	Description
LineItems	List<Apttus_Config2__LineItem__c>	The list of the Line Items.

Response Data Object - Apttus_CPQApi.CPQ.AddCustomBundleResponseDO		
Field	Type	Description
LineNumber	Decimal	The bundle line number.

### Code Sample

The sample code below enables you to add custom bundles with a list of bundle and standalone line items to an existing cart by running the code in Execute Anonymous window from developer console

.

1	<code>/**</code>
2	<code> * The below method demonstrates how to add custom bundle products to an existing cart (every quote has a cart)</code>
3	<code> * Lets assume the Quote's Cart is already configured and the standalone/ bundle products are Laptop, Monitor, Wifi Router</code>

```
4      * The input of this method is Quote Number and the line items, Monitor
5      and Wifi Router standalone products
6      */
7      List<Apttus_Config2__LineItem__c> lineItems = new
8      List<Apttus_Config2__LineItem__c>();
9      Apttus_CPQApi.CPQ.LineItemCollDO itemCollDO = new
10     Apttus_CPQApi.CPQ.LineItemCollDO();
11     Product2 bundleSO = [SELECT Id FROM Product2 WHERE Name = Laptop'
12     LIMIT 1];
13     Product2 optionSO = [SELECT Id FROM Product2 WHERE Name = Monitor'
14     LIMIT 1];
15     Product2 optionSO1 = [SELECT Id FROM Product2 WHERE Name = 'Wifi
16     Router' LIMIT 1];
17     Product2 standaloneSO = [SELECT Id FROM Product2 WHERE Name =
18     'Quoting Standalone 4' LIMIT 1];
19
20     // bundle products
21     itemCollDO.LineItems.add(new
22     Apttus_Config2__LineItem__c(Apttus_Config2__Quantity__c = 1,
23     Apttus_Config2__SellingTerm__c = 1,
24     Apttus_Config2__Comments__c = 'Test bundle',
25     Apttus_Config2__ProductId__c = bundleSO.Id,
26     Apttus_Config2__HasOptions__c = true,
27     Apttus_Config2__LineType__c = 'Product/Service',
28     Apttus_Config2__IsPrimaryLine__c = true,
29     Apttus_Config2__ChargeType__c = 'License Fee',
30     Apttus_Config2__PriceType__c = 'One Time',
31     Apttus_Config2__PriceMethod__c = 'Per Unit',
32     Apttus_Config2__ListPrice__c = 5000,
33     Apttus_Config2__IsCustomPricing__c = true,
```

```
27 Apttus_Config2__AllowManualAdjustment__c = true));
28
29
30     // option products
31     itemCollDO.LineItems.add(new
32     Apttus_Config2__LineItem__c(Apttus_Config2__Quantity__c = 1,
33
34     Apttus_Config2__SellingTerm__c = 1,
35
36     Apttus_Config2__Comments__c = 'Test option 1',
37
38     Apttus_Config2__ProductId__c = bundleS0.Id,
39
40     Apttus_Config2__OptionId__c = optionS0.Id,
41
42     Apttus_Config2__LineType__c = 'Option',
43
44     Apttus_Config2__IsQuantityModifiable__c = true,
45
46     Apttus_Config2__IsPrimaryLine__c = true,
47
48     Apttus_Config2__ChargeType__c = 'License Fee',
49
50     Apttus_Config2__PriceType__c = 'One Time',
51
52     Apttus_Config2__PriceMethod__c = 'Per Unit',
53
54     Apttus_Config2__ListPrice__c = 750,
55
56     Apttus_Config2__IsCustomPricing__c = true,
57
58     Apttus_Config2__AllowManualAdjustment__c = true));
59
60     itemCollDO.LineItems.add(new
61     Apttus_Config2__LineItem__c(Apttus_Config2__Quantity__c = 1,
62
63     Apttus_Config2__SellingTerm__c = 1,
64
65     Apttus_Config2__Comments__c = 'Test option 2',
66
67     Apttus_Config2__ProductId__c = bundleS0.Id,
68
69     Apttus_Config2__OptionId__c = optionS01.Id,
```

```
51 Apttus_Config2__LineType__c = 'Option',
52 Apttus_Config2__IsQuantityModifiable__c = true,
53 Apttus_Config2__IsPrimaryLine__c = false,
54 Apttus_Config2__ChargeType__c = 'Installation Fee',
55 Apttus_Config2__PriceType__c = 'One Time',
56 Apttus_Config2__PriceMethod__c = 'Per Unit',
57 Apttus_Config2__ListPrice__c = 10,
58 Apttus_Config2__IsCustomPricing__c = true,
59 Apttus_Config2__AllowManualAdjustment__c = true));
60
61 itemCollDO.LineItems.add(new
62 Apttus_Config2__LineItem__c(Apttus_Config2__Quantity__c = 1,
63 Apttus_Config2__SellingTerm__c = 1,
64 Apttus_Config2__Comments__c = 'Test bundle',
65 Apttus_Config2__ProductId__c = standaloneS0.Id,
66 Apttus_Config2__HasOptions__c = false,
67 Apttus_Config2__LineType__c = 'Product/Service',
68 Apttus_Config2__IsPrimaryLine__c = true,
69 Apttus_Config2__ChargeType__c = 'License Fee',
70 Apttus_Config2__PriceType__c = 'One Time',
71 Apttus_Config2__PriceMethod__c = 'Per Unit',
72 Apttus_Config2__ListPrice__c = 5000,
Apttus_Config2__IsCustomPricing__c = true,
```



```

73 Apttus_Config2__AllowManualAdjustment__c = true));
74     Apttus_CPQApi.CPQ.AddCustomBundleRequestDO request = new
Apttus_CPQApi.CPQ.AddCustomBundleRequestDO();
75     request.CartId = 'a1I6C00000116mn'; //cart Id
76     request.LineItemColl = itemCollDO;
77     Apttus_CPQApi.CPQWebService.addCustomBundle(request);

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## Response/Request XML

### Example Request

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:cpq="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:cpq1="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ">
  <soapenv:Header>
    <cpq:SessionHeader>
      <cpq:sessionId>00DZ000000NAEIA!
ASAAQJnmdThijKiZBe7nFMnVQReN90.MMbcgJD70Lgwd3vGv0DaYV9beTHNKgBG0W1ecM4askcVGqf9Qzk2nW
zcvo0m58gGD</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <cpq:addCustomBundle>
      <cpq:request>
        <cpq1:CartId>a10Z0000002Yqya</cpq1:CartId>
        <cpq1:LineItemColl>
          <cpq1:LineItems>
            <cpq:Apttus_Config2__Quantity__c>1</cpq:Apttus_Config2__Quant
ity__c>
            <cpq:Apttus_Config2__SellingTerm__c>1</cpq:Apttus_Config2__Se
llingTerm__c>
            <cpq:Apttus_Config2__Comments__c>None</cpq:Apttus_Config2__Co
mments__c>
          </cpq1:LineItems>
        </cpq1:LineItemColl>
      </cpq:request>
    </cpq:addCustomBundle>
  </soapenv:Body>
</soapenv:Envelope>

```

```

        <cpq:Apttus_Config2__ProductId__c>01te0000005NGpD</cpq:Apttus
_Apttus_Config2__ProductId__c>
        <cpq:Apttus_Config2__ClassificationId__c>a0ne0000007G1mR</cpq:
_Apttus_Config2__ClassificationId__c>
        <cpq:Apttus_Config2__HasOptions__c>>false</cpq:Apttus_Config2_
_Apttus_Config2__HasOptions__c>
        <cpq:Apttus_Config2__LineType__c>Product/Service</cpq:Apttus_
_Apttus_Config2__LineType__c>
        <cpq:Apttus_Config2__IsQuantityModifiable__c>>true</cpq:Apttus
_Apttus_Config2__IsQuantityModifiable__c>
        <cpq:Apttus_Config2__IsPrimaryLine__c >true</cpq:Apttus_Confi
g2__IsPrimaryLine__c >
        <cpq:Apttus_Config2__ChargeType__c>Standard Price</cpq:Apttus
_Apttus_Config2__ChargeType__c>
        <cpq:Apttus_Config2__PriceType__c >One Time</cpq:Apttus_Confi
g2__PriceType__c >
        <cpq:Apttus_Config2__PriceMethod__c>Per Unit</cpq:Apttus_Conf
ig2__PriceMethod__c>
        <cpq:Apttus_Config2__ListPrice__c>100</cpq:Apttus_Config2__Li
stPrice__c>
        <cpq:Apttus_Config2__IsCustomPricing__c>true</cpq:Apttus_Conf
ig2__IsCustomPricing__c>
        <cpq:Apttus_Config2__AllowManualAdjustment__c>true</cpq:Apttu
s_Config2__AllowManualAdjustment__c>
    </cpq1:LineItems>
    <cpq1:LineItems>
        <cpq:Apttus_Config2__Quantity__c>1</cpq:Apttus_Config2__Quant
ity__c>
        <cpq:Apttus_Config2__SellingTerm__c>1</cpq:Apttus_Config2__Se
llingTerm__c>
        <cpq:Apttus_Config2__Comments__c>Some comment</cpq:Apttus_Con
fig2__Comments__c>
        <cpq:Apttus_Config2__ProductId__c>01te0000005NGpX</cpq:Apttus
_Apttus_Config2__ProductId__c>
        <cpq:Apttus_Config2__HasOptions__c>>false</cpq:Apttus_Config2_
_Apttus_Config2__HasOptions__c>
        <cpq:Apttus_Config2__LineType__c>Product/Service</cpq:Apttus_
_Apttus_Config2__LineType__c>
        <cpq:Apttus_Config2__IsQuantityModifiable__c>true</cpq:Apttus
_Apttus_Config2__IsQuantityModifiable__c>
        <cpq:Apttus_Config2__IsPrimaryLine__c >true</cpq:Apttus_Confi
g2__IsPrimaryLine__c >

```

```

        <cpq:Apttus_Config2__ChargeType__c>Standard Price</cpq:Apttus
_Apttus_Config2__ChargeType__c>
        <cpq:Apttus_Config2__PriceType__c >One Time</cpq:Apttus_Confi
g2__PriceType__c >
        <cpq:Apttus_Config2__PriceMethod__c>Per Unit</cpq:Apttus_Conf
ig2__PriceMethod__c>
        <cpq:Apttus_Config2__ListPrice__c>100</cpq:Apttus_Config2__Li
stPrice__c>
        <cpq:Apttus_Config2__IsCustomPricing__c>true</cpq:Apttus_Conf
ig2__IsCustomPricing__c>
        <cpq:Apttus_Config2__AllowManualAdjustment__c>false</cpq:Aptt
us_Config2__AllowManualAdjustment__c>
        </cpq1:LineItems>
        <cpq1:LineItems>
        <cpq:Apttus_Config2__Quantity__c>2</cpq:Apttus_Config2__Quant
ity__c>
        <cpq:Apttus_Config2__SellingTerm__c>1</cpq:Apttus_Config2__Se
llingTerm__c>
        <cpq:Apttus_Config2__OptionId__c>01te0000005NGpm</cpq:Apttus_
Config2__OptionId__c>
        <cpq:Apttus_Config2__LineType__c>Option</cpq:Apttus_Config2__
LineType__c>
        <cpq:Apttus_Config2__IsQuantityModifiable__c>true</cpq:Apttus
_Config2__IsQuantityModifiable__c>
        <cpq:Apttus_Config2__IsPrimaryLine__c >false</cpq:Apttus_Conf
ig2__IsPrimaryLine__c >
        <cpq:Apttus_Config2__ChargeType__c>Standard Price</cpq:Apttus
_Config2__ChargeType__c>
        <cpq:Apttus_Config2__PriceType__c >One Time</cpq:Apttus_Confi
g2__PriceType__c >
        <cpq:Apttus_Config2__PriceMethod__c>Per Unit</cpq:Apttus_Conf
ig2__PriceMethod__c>
        <cpq:Apttus_Config2__ListPrice__c>200</cpq:Apttus_Config2__Li
stPrice__c>
        <cpq:Apttus_Config2__IsCustomPricing__c>false</cpq:Apttus_Con
fig2__IsCustomPricing__c>
        <cpq:Apttus_Config2__AllowManualAdjustment__c>false</cpq:Aptt
us_Config2__AllowManualAdjustment__c>
        </cpq1:LineItems>
        </cpq1:LineItemColl>
    </cpq:request>
</cpq:addCustomBundle>
</soapenv:Body>
</soapenv:Envelope>

```

```

Example Response

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:AddCustomBundleResponseDO="http://soap.sforce.com/schemas/class/
Apttus_CPQApi/CPQ">
  <soapenv:Body>
    <addCustomBundleResponse>
      <result>
        <AddCustomBundleResponseDO:LineNumber>6</AddCustomBundleResponseDO:Li
neNumber>
      </result>
    </addCustomBundleResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Adding Options to a Bundle

This adds one or more Options Products to a Bundle product.

API	Signature	
<b>addOptions</b>	<i>webservice static Apttus_CPQApi.CPQ.AddOptionsResponseDO addOptions(Id cartId, Integer lineNumber, List selectedOptions)</i>	
Request Parameters		
Name	Type	Description
CartID	ID	The cart ID
lineNumber	Integer	Line Number of the Bundle Product
selectedOptions	List<Apttus_CPQApi.CPQ.Selected OptionDO>	List of Options

Data Object - Apttus_CPQApi.CPQ.SelectedOptionDO		
Field	Type	Description
AttributeValues	List	List of attributes values.
Comments	String	Comments associated with the record.
ComponentId	ID	Id of the component.
ComponentProductId	ID	Id of the component product.
CustomData	Apttus_Config2__LineItem__c	The values to be set for the custom fields in the CustomFields List
CustomFields	List<String>	List of Custom Field's API Name
EndDate	Date	The end date.
Quantity	Decimal	The option quantity.
SellingTerm	Decimal	The option selling term.
SourceId	ID	ID of the source option.
StartDate	Date	The start date. You should ensure you use the correct date format.

### Code Sample

Using the sample below you can dynamically add options to a bundled product that the user selects in the cart line item. For example, if the user selects a bundled product, Laptop, in the cart line item you can fetch the configured options for that selected product using the *addoptions* API. In the sample below, for a selected product you initially fetch the productID. Using the retrieve option groups API *getOptionGroupsForPriceListProduct*, you can check whether the selected product has option groups associated. If an option group is associated, fetch the option components and create a list comprising the CPQ.SelectedOptionDO. Invoke the *addoptions* API and pass the cartID, productID, and the list CPQ.SelectedOptionDO.

```

1  public void addOptions()
2  {
3      string productName='';
4      Integer lineNumber = 0;
5      for(LineItemWrapperClass selProdWrap: lstWrapItems)
6      {
7          //For a selected bundled product from a cart line item execute the
loop below
8          if(selProdWrap.selected)
9          {
10             productName = selProdWrap.ProductName;
11             lineNumber = Integer.valueOf(selProdWrap.LineNumber);
12             break;
13         }
14     }
15     //If no product is selected, show an appropriate error message
16     if(String.isBlank(productName))
17     {
18         ApexPages.addMessage(new
ApexPages.Message(ApexPages.severity.info, 'Please select at least one
Product. '));
19     }
20     else
21     {
22         List<Apttus_CPQApi.CPQ.SelectedOptionDO> selectedOptDOList = new
List<Apttus_CPQApi.CPQ.SelectedOptionDO>();
23
24         //For the selected product fetch the product ID
25         List<Product2> productId = [SELECT Id FROM Product2 WHERE Name
= :productName LIMIT 1];
26
27         //Execute the getoptionGroups API to fetch the options with option
groups
28         Apttus_CPQApi.CPQ.ProductOptionGroupSearchResultDO result =
Apttus_CPQApi.CPQWebService.getOptionGroupsForPriceListProduct(priceListId
, productId[0].Id);
29
30         //If the selected product has option groups execute the loop below
31         if(result.HasOptionGroups)
32         {

```

```

33         List<Apttus_CPQApi.CPQ.ProductOptionGroupDO> prodOptGrpDOList
= result.OptionGroups;
34         for(Apttus_CPQApi.CPQ.ProductOptionGroupDO prodOptGrpDO :
prodOptGrpDOList)
35             {
36                 //For an option group if a product has option components
such as quantity execute the loop below
37                 if(prodOptGrpDO.HasOptionComponents)
38                     {
39                         List<Apttus_CPQApi.CPQ.ProductOptionComponentDO>
prodOptCompDOList = new
List<Apttus_CPQApi.CPQ.ProductOptionComponentDO>();
40                         prodOptCompDOList = prodOptGrpDO.OptionComponents;
41
42                         //Fetch all the option components for a particular
option group.
43                         for(Apttus_CPQApi.CPQ.ProductOptionComponentDO
prodOptCompDO :prodOptCompDOList )
44                             {
45                                 Apttus_CPQApi.CPQ.SelectedOptionDO selectedOptDO =
new Apttus_CPQApi.CPQ.SelectedOptionDO();
46                                 selectedOptDO.ComponentId =
prodOptCompDO.ComponentId;
47                                 selectedOptDO.ComponentProductId =
prodOptCompDO.ComponentProductId;
48                                 selectedOptDO.Quantity = 1;
49                                 selectedOptDOList.add(selectedOptDO);
50                             }
51                         }
52                     }
53             }
54             //Invoke the addOptions API and pass the cartID, selected product
line number, and the Option List retrieved from the getoptiongroups API
55             Apttus_CPQApi.CPQ.AddOptionsResponseDO addOptRespDO =
Apttus_CPQApi.CPQWebService.addOptions(cartId, lineNumber,
selectedOptDOList);
56
57         }
58     }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## Response/Request XML

### Example Request

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:cpq="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:cpq1="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ">
  <soapenv:Header>
    <cpq:SessionHeader>
      <cpq:sessionId>00DZ000000NAEIA!
ASAAQHmIRqgn4R90i1yQjWTIVk4UZmsDe_.eK0Z9z6qLij7Tu.L_Yo8dRA_p80mhKMeRs4uzCZTadIgQ9fEbD
KciXEQYRyaA</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <cpq:addOptions>
      <cpq:cartId>a10Z0000002ZQVZ</cpq:cartId>
      <cpq:lineNumber>1</cpq:lineNumber>
      <cpq:selectedOptions>
        <cpq1:Comments>Some comment</cpq1:Comments>
        <cpq1:ComponentId>a1YZ0000003pIGM</cpq1:ComponentId>
        <cpq1:ComponentProductId>01tZ0000004LUYI</cpq1:ComponentProductId>
        <cpq1:EndDate>2020-06-30</cpq1:EndDate>
        <cpq1:Quantity>5</cpq1:Quantity>
        <cpq1:SellingTerm>1</cpq1:SellingTerm>
        <cpq1:StartDate>2020-05-28</cpq1:StartDate>
      </cpq:selectedOptions>
    </cpq:addOptions>
  </soapenv:Body>
</soapenv:Envelope>
```



**Example Response**

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:AddOptionsResponseD0="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <addOptionsResponse>
      <result>
        <AddOptionsResponseD0:addedOptionLineItems xsi:type="Apttus_Config2__LineItem__c">
          <Id>a19Z00000006sPeBIAU</Id>
          <Apttus_Config2__AddedBy__c>User</Apttus_Config2__AddedBy__c>
          <Apttus_Config2__AllowManualAdjustment__c>true</Apttus_Config2__AllowManualAdjustment__c>
          <Apttus_Config2__Comments__c>Some comment</Apttus_Config2__Comments__c>
          <Apttus_Config2__ConfigStatus__c>NA</Apttus_Config2__ConfigStatus__c>
          <Apttus_Config2__ConfigurationId__c>a10Z00000002ZQVZMA4</Apttus_Config2__ConfigurationId__c>
          <Apttus_Config2__ConstraintCheckStatus__c>NA</Apttus_Config2__ConstraintCheckStatus__c>
          <Apttus_Config2__Customizable__c>>false</Apttus_Config2__Customizable__c>
          <Apttus_Config2__Description__c>Auto_API_OptionProduct1</Apttus_Config2__Description__c>
          <Apttus_Config2__EndDate__c>2020-06-30</Apttus_Config2__EndDate__c>
          <Apttus_Config2__HasAttributes__c>>false</Apttus_Config2__HasAttributes__c>
          <Apttus_Config2__HasOptions__c>>false</Apttus_Config2__HasOptions__c>
          <Apttus_Config2__IsPrimaryLine__c>>true</Apttus_Config2__IsPrimaryLine__c>
          <Apttus_Config2__IsQuantityModifiable__c>>false</Apttus_Config2__IsQuantityModifiable__c>
          <Apttus_Config2__ItemSequence__c>2</Apttus_Config2__ItemSequence__c>
          <Apttus_Config2__LineNumber__c>1</Apttus_Config2__LineNumber__c>
          <Apttus_Config2__LineType__c>Option</Apttus_Config2__LineType__c>

```

```


        <Apttus_Config2__OptionId__c>01tZ0000004lUYIIA2</
Apttus_Config2__OptionId__c>
        <Apttus_Config2__ParentBundleNumber__c>1</
Apttus_Config2__ParentBundleNumber__c>
        <Apttus_Config2__PriceListId__c>a1DZ0000002mg5nMAA</
Apttus_Config2__PriceListId__c>
        <Apttus_Config2__PricingStatus__c>Pending</
Apttus_Config2__PricingStatus__c>
        <Apttus_Config2__PrimaryLineNumber__c>4</
Apttus_Config2__PrimaryLineNumber__c>
        <Apttus_Config2__ProductId__c>01tZ0000004lUXUIA2</
Apttus_Config2__ProductId__c>
        <Apttus_Config2__ProductOptionId__c>a1YZ0000003pIGMMA2</
Apttus_Config2__ProductOptionId__c>
        <Apttus_Config2__ProductVersion__c>1.00</
Apttus_Config2__ProductVersion__c>
        <Apttus_Config2__Quantity__c>5</Apttus_Config2__Quantity__c>
        <Apttus_Config2__SellingTerm__c>1</Apttus_Config2__SellingTerm__c>
        <Apttus_Config2__StartDate__c>2020-05-28</
Apttus_Config2__StartDate__c>
        <Apttus_Config2__Uom__c>Each</Apttus_Config2__Uom__c>
    </AddOptionsResponseDO:addedOptionLineItems>
</result>
</addOptionsResponse>
</soapenv:Body>
</soapenv:Envelope>

```

## Cloning Bundle Line Items on the Cart

This method allows you to clone the primary bundle line items along with option line items and child line items (if applicable) on the Cart. This API is invoked when you click **Clone** action icon next to a bundle product on the Cart.

API	Signature
cloneBundleLineItems	<i>webservice static Apttus_CPQApi.CPQ.CloneLineItemsResponseDO cloneBundleLineItems(Apttus_CPQApi.CPQ.CloneLineItemsRequestDO request)</i>

Request Parameter		
Name	Type	Description
Request	Apttus_CPQApi.CPQ.CloneLineItemsRequestDO	This is the request data object.
Request Data Object - Apttus_CPQApi.CPQ.CloneLineItemsRequestDO		
Name	Type	Description
Cart ID	ID	The Id of the cart.
Primary LineNumber	Integer	The line numbers of the primary line items which have to be cloned using this API.
<div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> You must pass the value <i>Apttus_Config2__LineNumber_c</i> field instead of <i>Apttus_Config2__PrimaryLineNumber_c</i> to clone bundles.</p> </div>		
Response Data Object - Apttus_CPQApi.CPQ.CloneLineItemsResponseDO		
Name	Type	Description
OriginalToCloneMap	List<Apttus_CPQApi.CPQ.IntegerMapDO>	The list of primary line item numbers.
Response Data Object - Apttus_CPQApi.CPQ.IntegerMapDO		
Name	Type	Description
mapItem.Key	Integer	The line item numbers of the source primary line item from which new line items are cloned using this API.
mapItem.Value	Integer	The line item numbers of the cloned line items.

### Code Sample

Using the below sample, you can clone the primary bundle line items after you configure your products, add the attributes, configure your options and arrive on the cart. This API accepts the Cart ID and the primary bundle and option line item numbers. In the form of

response, this API provides the original line items in the `mapItem.Key` parameter and the newly cloned line items in the `mapItem.Value` parameter.

For example, once you configure your cart by adding products using `addBundle` or `addMultiProducts` API, you can invoke this API in order to clone the bundle and option line items on the cart at any of the following:

- After the product is added on the Cart.
- Before making changes to pricing of the products.
- After updating the cart pricing using `updatePriceForCart` API.

```
/**
 * The below method demonstrates how to clone bundle product in an existing cart
 (every quote has a cart)
 * Lets assume the Quote's Cart has a 3 products,
 * Laptop is a bundle product (line number 1) and Monitor and Wifi Router are
 standalone products (line number 2 and 3 respectively)
 * The input of this method is Quote Number and the line number of the Laptop bundle
 product
 */
public static void cloneBundleLineItems(String quoteNumber, List<Integer>
primaryLineNumbers)
{
    List<Apttus_Config2__ProductConfiguration__c> cart = [SELECT Id FROM
Apttus_Config2__ProductConfiguration__c WHERE Apttus_QPConfig__ProposalId__r.Name
= :quoteNumber LIMIT 1];

    if(!cart.isEmpty() && primaryLineNumbers != null && !
primaryLineNumbers.isEmpty())
    {

        // Create the request object
        Apttus_CPQApi.CPQ.CloneLineItemsRequestDO request = new
Apttus_CPQApi.CPQ.CloneLineItemsRequestDO();
        request.CartId = cart.get(0).Id;
        request.PrimaryLineNumbers = primaryLineNumbers;

        // Execute the cloneBundleLineItems routine
        Apttus_CPQApi.CPQ.CloneLineItemsResponseDO response =
Apttus_CPQApi.CPQWebService.cloneBundleLineItems(request);
    }
}
```

```

    for(Apttus_CPQApi.CPQ.IntegerMapDO intMapDO : response.OriginalToCloneMap)
    {
        System.debug('Source Bundle Line Number = ' + intMapDO.Key + ', Cloned
Bundle Line Number = ' + intMapDO.Value);
    }
}
}

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## API Prerequisites

- [Creating a Cart from a Quote](#)
- [Adding Products to a Cart OR Adding a Bundle to a Cart](#)
- [Updating Price For A Cart](#)

## Response/Request XML

### Example Request

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:cpq="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:cpq1="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ">
  <soapenv:Header>
    <cpq:SessionHeader>
      <cpq:sessionId>00DZ000000NAEIA!
ASAAQGrz6Z.wmD7Foe_ZhEj44N.Sh34LJRyn0uiVnJa1hQAZgWx3lQ2kM._9AWarLziTn5kpbVtSuMZEgdXN_
Mlze.kGwLVg</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <cpq:cloneBundleLineItems>
      <cpq:request>
        <cpq1:CartId>a10Z0000002YouHMAS</cpq1:CartId>
        <cpq1:PrimaryLineNumbers>3</cpq1:PrimaryLineNumbers>
      </cpq:request>
    </cpq:cloneBundleLineItems>
  </soapenv:Body>
</soapenv:Envelope>

```

```

        <cpq1:PrimaryLineNumbers>4</cpq1:PrimaryLineNumbers>
    </cpq:request>
</cpq:cloneBundleLineItems>
</soapenv:Body>
</soapenv:Envelope>

```

### Example Response

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:CloneLineItemsResponseD0="http://soap.sforce.com/schemas/class/
Apttus_CPQApi/CPQ">
  <soapenv:Body>
    <cloneBundleLineItemsResponse>
      <result>
        <CloneLineItemsResponseD0:OriginalToCloneMap>
          <CloneLineItemsResponseD0:Key>3</CloneLineItemsResponseD0:Key>
          <CloneLineItemsResponseD0:Value>5</CloneLineItemsResponseD0:Value
        >
        </CloneLineItemsResponseD0:OriginalToCloneMap>
        <CloneLineItemsResponseD0:OriginalToCloneMap>
          <CloneLineItemsResponseD0:Key>4</CloneLineItemsResponseD0:Key>
          <CloneLineItemsResponseD0:Value>6</CloneLineItemsResponseD0:Value
        >
        </CloneLineItemsResponseD0:OriginalToCloneMap>
      </result>
    </cloneBundleLineItemsResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Cloning Line Items on the Cart

This method enables you to clone the primary line items on the Cart. This API is invoked when you click **Clone** action icon next to a standalone product on the Cart.

API	Signature	
cloneLineItems	<i>webService static Apttus_CPQApi.CPQ.CloneLineItemsResponseDO cloneLineItems(Apttus_CPQApi.CPQ.CloneLineItemsRequestDO request)</i>	
Request Parameter		
Name	Type	Description
Request	Apttus_CPQApi.CPQ.CloneLineItemsRequestDO	This is the request data object.
Request Data Object - Apttus_CPQApi.CPQ.CloneLineItemsRequestDO		
Name	Type	Description
Cart ID	ID	The Id of the cart.
PrimaryLineNumber	Integer	The line item numbers of the primary line items which have to be cloned using this API.
Response Data Object - Apttus_CPQApi.CPQ.CloneLineItemsResponseDO		
Name	Type	Description
OriginalToCloneMap	List<Apttus_CPQApi.CPQ.IntegerMapDO>	The list of primary line item numbers.
Response Data Object - Apttus_CPQApi.CPQ.IntegerMapDO		
Name	Type	Description
mapItem.Key	Integer	The line item numbers of the source primary line item from which new line items are cloned using this API.
mapItem.Value	Integer	The line item numbers of the cloned line items.

## Code Sample

Using the following sample, you can clone the primary line items after you configure your products, add the attributes and arrive on the cart. This API accepts the Cart ID and the primary line item numbers, and clones these line items. In the form of the response, this API provides the original line items in the `mapItem.Key` parameter and the newly cloned line items in the `mapItem.Value` parameter.

For example, once you configure your cart using *addMultiProducts* API, you can invoke this API in order to clone the line items on the cart at any of the following:

- After the product is added on the Cart.
- Before making changes to pricing of the products.
- After updating the cart pricing using *updatePriceForCart* API.

```
/**
 * The below method demonstrates how to clone standalone product in an existing cart
 (every quote has a cart)
 * Lets assume the Quote's Cart has a 3 products,
 * Laptop is a bundle product (line number 1) and Monitor and Wifi Router are
 standalone products (line number 2 and 3 respectively)
 * The input of this method is Quote Number and the line numbers of the Monitor and
 Wifi Router are standalone products
 */
public static void cloneLineItems(String quoteNumber, List<Integer>
primaryLineNumbers)
{

    List<Apttus_Config2__ProductConfiguration__c> cart = [SELECT Id FROM
Apttus_Config2__ProductConfiguration__c WHERE Apttus_QPConfig__ProposalId__r.Name
= :quoteNumber LIMIT 1];

    if(!cart.isEmpty() && primaryLineNumbers != null && !
primaryLineNumbers.isEmpty())
    {

        // Create the request object
        Apttus_CPQApi.CPQ.CloneLineItemsRequestDO request = new
Apttus_CPQApi.CPQ.CloneLineItemsRequestDO();
        request.CartId = cart.get(0).Id;
        request.PrimaryLineNumbers = primaryLineNumbers;

        // Execute the cloneLineItems routine
```



```

    Apttus_CPQApi.CPQ.CloneLineItemsResponseDO response =
    Apttus_CPQApi.CPQWebService.cloneLineItems(request);

    for(Apttus_CPQApi.CPQ.IntegerMapDO intMapDO : response.OriginalToCloneMap)
    {
        System.debug('Source Bundle Line Number = ' + intMapDO.Key + ', Cloned
        Bundle Line Number = ' + intMapDO.Value);
    }
}
}

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

### API Prerequisites

- [Creating a Cart from a Quote](#)
- [Adding Products to a Cart](#)
- [Updating Price For A Cart](#)

## Response/Request XML

### Example Request

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:cpq="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:cpq1="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ">
  <soapenv:Header>
    <cpq:SessionHeader>
      <cpq:sessionId>00DZ000000NAEIA!
      ASAAQGrz6Z.wmD7Foe_ZhEj44N.Sh34lJRYn0uiVnJa1hQAZgWx3lQ2kM._9AWarLZiTn5kpbVtSuMZEgdXN_
      Mlze.kGwLVg</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <cpq:cloneLineItems>
      <cpq:request>

```

```

        <cpq1:CartId>a10Z00000002YotOMAS</cpq1:CartId>
        <cpq1:PrimaryLineNumbers>1</cpq1:PrimaryLineNumbers>
        <cpq1:PrimaryLineNumbers>2</cpq1:PrimaryLineNumbers>
        <cpq1:PrimaryLineNumbers>5</cpq1:PrimaryLineNumbers>
    </cpq:request>
</cpq:cloneLineItems>
</soapenv:Body>
</soapenv:Envelope>

```

### Example Response

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:CloneLineItemsResponseD0="http://soap.sforce.com/schemas/class/
Apttus_CPQApi/CPQ">
  <soapenv:Body>
    <cloneLineItemsResponse>
      <result>
        <CloneLineItemsResponseD0:OriginalToCloneMap>
          <CloneLineItemsResponseD0:Key>1</CloneLineItemsResponseD0:Key>
          <CloneLineItemsResponseD0:Value>6</CloneLineItemsResponseD0:Value
        >
        </CloneLineItemsResponseD0:OriginalToCloneMap>
        <CloneLineItemsResponseD0:OriginalToCloneMap>
          <CloneLineItemsResponseD0:Key>2</CloneLineItemsResponseD0:Key>
          <CloneLineItemsResponseD0:Value>7</CloneLineItemsResponseD0:Value
        >
        </CloneLineItemsResponseD0:OriginalToCloneMap>
        <CloneLineItemsResponseD0:OriginalToCloneMap>
          <CloneLineItemsResponseD0:Key>5</CloneLineItemsResponseD0:Key>
          <CloneLineItemsResponseD0:Value>8</CloneLineItemsResponseD0:Value
        >
        </CloneLineItemsResponseD0:OriginalToCloneMap>
      </result>
    </cloneLineItemsResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Cloning Option Line Items on the Cart

This method allows you to clone the option line items and child line items (if applicable) on the Cart. This API is invoked when you click **Clone** action icon next to an option product on the Cart.

API	Signature
<b>cloneOptionLineItems</b>	<i>webservice static Apttus_CPQApi.CPQ.CloneLineItemsResponseDO cloneOptionLineItems(Apttus_CPQApi.CPQ.CloneLineItemsRequestDO request)</i>

Request Parameter		
Name	Type	Description
Request	Apttus_CPQApi.CPQ.CloneLineItemsRequestDO	This is the request data object.

Request Data Object - Apttus_CPQApi.CPQ.CloneLineItemsRequestDO		
Name	Type	Description
Cart ID	ID	The Id of the cart.
PrimaryLineNumber	Integer	The line numbers of the primary line items which have to be cloned using this API.

Response Data Object - Apttus_CPQApi.CPQ.CloneLineItemsResponseDO		
Name	Type	Description
OriginalToCloneMap	List<Apttus_CPQApi.CPQ.IntegerMapDO>	The list of primary line item numbers.

Response Data Object - Apttus_CPQApi.CPQ.IntegerMapDO		
Name	Type	Description
mapItem.Key	Integer	The line item numbers of the source primary line item from which new line items are cloned using this API.

Response Data Object - Apttus_CPQApi.CPQ.IntegerMapDO		
Name	Type	Description
mapItem.Value	Integer	The line item numbers of the cloned line items.

## Code Sample

Using the below code sample, you can clone the option line items after you configure your bundle products, add the attributes and arrive on the cart. This API accepts the Cart ID and the option line item numbers, and clones these line items. In the form of the response, this API provides the original line items in the `mapItem.Key` parameter and the newly cloned line items in the `mapItem.Value` parameter.

For example, once you configure your cart by adding option products using *addoptions* API, you can invoke this API in order to clone the option line items on the cart at any of the following:

- After the product is added on the Cart.
- Before making changes to pricing of the products.
- After updating the cart pricing using *updatePriceForCart* API.

```
/**
 * The below method demonstrates how to clone option line items in an existing cart
 (every quote has a cart)
 * Lets assume the Quote's Cart has a 3 products,
 * Laptop is a bundle product (line number 1) and Monitor and Wifi Router are
 standalone products (line number 2 and 3 respectively)
 * The input of this method is Quote Number and the line number of the Laptop bundle
 product
 */
public static void cloneOptionLineItems(String quoteNumber, List<Integer>
primaryLineNumbers)
{

    List<Apttus_Config2__ProductConfiguration__c> cart = [SELECT Id FROM
Apttus_Config2__ProductConfiguration__c WHERE Apttus_QPConfig__ProposalId__r.Name
= :quoteNumber LIMIT 1];
```

```

    if(!cart.isEmpty() && primaryLineNumbers != null && !
primaryLineNumbers.isEmpty())
    {

        // Create the request object
        Apttus_CPQApi.CPQ.CloneLineItemsRequestDO request = new
Apttus_CPQApi.CPQ.CloneLineItemsRequestDO();
        request.CartId = cart.get(0).Id;
        request.PrimaryLineNumbers = primaryLineNumbers;

        // Execute the cloneOptionLineItems routine
        Apttus_CPQApi.CPQ.CloneLineItemsResponseDO response =
Apttus_CPQApi.CPQWebService.cloneOptionLineItems(request);

        for(Apttus_CPQApi.CPQ.IntegerMapDO intMapDO : response.OriginalToCloneMap)
        {
            System.debug('Original Primary Line Number = ' + intMapDO.Key + ', Cloned
Primary Line Number = ' + intMapDO.Value);
        }
    }
}

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

### API Prerequisites

- [Creating a Cart from a Quote](#)
- [Adding a Bundle to a Cart](#)
- [Updating Price For A Cart](#)

## Response/Request XML

### Example Request

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"

```

```

xmlns:cpq="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
xmlns:cpq1="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ">
<soapenv:Header>
  <cpq:SessionHeader>
    <cpq:sessionId>00DZ000000NAEIA!
ASAAQGYIEG86MxZJB2SAYA0.SocSmGG7G3SzeD6x2lnQ_zQCcBu.SpLUCrnOBk0Ph00tnf_OAAnE0nfpZfkaS
9mA_1G_1kmP</cpq:sessionId>
  </cpq:SessionHeader>
</soapenv:Header>
<soapenv:Body>
  <cpq:cloneOptionLineItems>
    <cpq:request>
      <cpq1:CartId>a10Z0000002YqyaMAC</cpq1:CartId>
      <cpq1:PrimaryLineNumbers>1</cpq1:PrimaryLineNumbers>
      <cpq1:PrimaryLineNumbers>3</cpq1:PrimaryLineNumbers>
    </cpq:request>
  </cpq:cloneOptionLineItems>
</soapenv:Body>
</soapenv:Envelope>

```

### Example Response

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:CloneLineItemsResponseD0="http://soap.sforce.com/schemas/class/
Apttus_CPQApi/CPQ">
  <soapenv:Body>
    <cloneOptionLineItemsResponse>
      <result>
        <CloneLineItemsResponseD0:OriginalToCloneMap>
          <CloneLineItemsResponseD0:Key>1</CloneLineItemsResponseD0:Key>
          <CloneLineItemsResponseD0:Value>9</CloneLineItemsResponseD0:Value
        >
        </CloneLineItemsResponseD0:OriginalToCloneMap>
        <CloneLineItemsResponseD0:OriginalToCloneMap>
          <CloneLineItemsResponseD0:Key>3</CloneLineItemsResponseD0:Key>
          <CloneLineItemsResponseD0:Value>12</CloneLineItemsResponseD0:Valu
        e>
        </CloneLineItemsResponseD0:OriginalToCloneMap>
      </result>

```

```

    </cloneOptionLineItemsResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Comparing Products

This is used to retrieve products, based on product IDs, and compare them side by side. This is typically done from the Product Catalog, to decide between products before adding them to the cart.

**Prerequisite:** You must set up feature sets and associate them with the products you want to compare. There is no API for this process—you must configure this in your org. For more information, see [Configuring Product Comparison](#).

API	Signature
compareProducts	<i>WebService static List compareProducts(Apttus_CPQApi.CPQ.FeatureInfoRequestDO request)</i>

Parameters		
Name	Type	Description
request	Apttus_CPQApi.CPQ.FeatureInfoRequestDO	The request data object.
Request Data Object - Apttus_CPQApi.CPQ.FeatureInfoRequestDO		
Field	Type	Description
productIds	List<ID>	The list of IDs of the product you want to compare.

Response Data Object - Apttus_CPQApi.CPQ.FeatureInfoResponseDO		
Field	Type	Description
FeatureInfos	List<Apttus_CPQApi.FeatureSupport.FeatureInfo>	This returns the list of features for each of the products included in the comparison.
ProductId	ID	ID of the products, which have a feature set to be compared.
Data Object - Apttus_CPQApi.FeatureSupport.FeatureInfo		
Field	Type	Description
ProductFeatureSet	FeatureSet	Associated Product Feature Set
ProductFeatureSOld	Id	Id of associated ProductFeature__c Salesforce Object
Data Object - Apttus_CPQApi. FeatureSupport.FeatureSet		
Field	Type	Description
Description	String	Description of associated product feature set
Features	List<Apttus_CPQApi.FeatureSupport.ProductFeatureValue>	List of associated feature values
FeatureSetSOld	Id	Id of associated Apttus_Config2__FeatureSet__c Salesforce Object
Name	String	Name of the associated Product Feature Set
Sequence	Decimal	Order of the feature set



Data Object - Apttus_CPQApi.FeatureSupport.ProductFeatureValue		
Field	Type	Description
FeatureSOld	Id	ID of the associated feature set.
IsIncluded	Boolean	Whether it is included
Name	String	Name of the feature
ProdFeatureValueSOld	Id	Product Feature Value Sold
ProdId	Id	ID of the product.
Sequence	Decimal	The feature display sequence
Value	String	Product Feature Value

### Code Sample

The sample below enables you to fetch the IDs of the selected products. Two or more products should be selected to enable comparison. After the user selects two or more products such as a Laptop from different vendors, invoke this API.

```
public List<CPQ.FeatureInfoResponseDO> compareProducts(List<ID>productIds)
{
    //Prompt appropriate error message validation message when the productIds
    parameter has less than two products.

    if(productIds.size()< 2)
    {
        System.debug('Please select atleast two products to compare.');
```

```
        return null;
    }

    //If two or more products are selected execute the code below

    else{
```

```

    Apttus_CPQApi.CPQ.FeatureInfoRequestDO request = new
Apttus_CPQApi.CPQ.FeatureInfoRequestDO();
    request.ProductIds = productIds;
    //Invoke the compare products API
    List<Apttus_CPQApi.CPQ.FeatureInfoResponseDO> response =
Apttus_CPQApi.CPQWebService.compareProducts(request);
    return response;
}
}

```

## Code sample

The sample code below allows you to create a table in a Visualforce page to display the compared products:

```

1  'compareFeatureSets' in Visualforce code is the getter variable which hold
the
2  response from compareProducts apex method.
3
4
5  Visualforce code
6
7
8  <!-- Compare Products Panel -->
9
10 <apex:outputPanel layout="block" Id="compareProductsPanel" styleClass="gri
d-2-13 clearfix compareProductsPanel">
11 <table>
12     <apex:repeat var="featureSetName" value="{!compareFeatureSets}">
13         <tr> <th data-priority="1" colspan="{!colSpan}"><h4>{!
featureSetName}</h4></th></tr>
14         <apex:repeat var="featureValue" value="{!compareFeatureSets
[featureSetName][0].productFeatureSet.features}">
15             <apex:outputPanel layout="none" rendered="{!
hasIncludedProduct[compareFeatureSets[featureSetName]
[0].productFeatureSet.featureSetSOId]}">
16                 <tr><th class="label">{!featureValue.Name}</th>
17                     <apex:repeat var="product" value="{!productIds}">
18                         <apex:repeat var="prodFeatureInfo" value="{!
productFeatureInfosResponse[product]}">

```

```

19         <apex:outputPanel layout="none" rendered="{!
prodFeatureInfo.productFeatureSet.Name == featureSetName}">
20         <apex:repeat var="prodFeatureValue" value="{!
prodFeatureInfo.productFeatureSet.features}">
21         <apex:outputPanel layout="none" rendered="{!
prodFeatureValue.featureS0Id == featureValue.featureS0Id &&
prodFeatureValue.isIncluded}">
22             <td style="border:1px #CCCCCC solid;">{!
prodFeatureValue.Value}</td>
23         </apex:outputPanel>
24         <apex:outputPanel layout="none" rendered="{!
prodFeatureValue.featureS0Id == featureValue.featureS0Id && NOT
(prodFeatureValue.isIncluded)}">
25             <td style="border:1px #CCCCCC solid;"></td>
26         </apex:outputPanel>
27     </apex:repeat>
28 </apex:outputPanel>
29 </apex:repeat>
30 </apex:repeat>
31 </tr>
32 </apex:outputPanel>
33 </apex:repeat>
34 </apex:repeat>
35 </table>
36 </apex:outputPanel>

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## API Prerequisites

None.

## Response/Request XML

### Example Request

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"

```

```

xmlns:cpq="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
xmlns:cpq1="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ">
<soapenv:Header>
  <cpq:SessionHeader>
    <cpq:sessionId>00DZ000000NAEIA!
ASAAQHmIRqgn4R90i1yQjWTIVk4UZmsDe_.eK0Z9z6qLij7Tu.L_Yo8dRA_p80mhKMeRs4uzCZTadIgQ9fEbD
KciXEQYRyaA</cpq:sessionId>
  </cpq:SessionHeader>
</soapenv:Header>
<soapenv:Body>
  <cpq:compareProducts>
    <cpq:request>
      <cpq1:ProductIds>01ti0000006iMl3</cpq1:ProductIds>
      <cpq1:ProductIds>01ti0000006iMl8</cpq1:ProductIds>
    </cpq:request>
  </cpq:compareProducts>
</soapenv:Body>
</soapenv:Envelope>

```

### Example Response

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:FeatureInfoResponseD0="http://soap.sforce.com/schemas/class/Apttus_CPQApi/
CPQ"
  xmlns:FeatureInfo="http://soap.sforce.com/schemas/class/Apttus_CPQApi/
FeatureSupport"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <compareProductsResponse>
      <result>
        <FeatureInfoResponseD0:FeatureInfos>
          <FeatureInfo:ProductFeatureSet>
            <FeatureInfo:Description xsi:nil="true"/>
            <FeatureInfo:Features>
              <FeatureInfo:FeatureS0Id>a1wi0000000jcV3AAI</FeatureInfo:
FeatureS0Id>
              <FeatureInfo:IsIncluded>true</FeatureInfo:IsIncluded>
              <FeatureInfo:Name>AutoCPQ951Feature1</FeatureInfo:Name>

```

```

        <FeatureInfo:ProdFeatureValueS0Id>a1zi0000000WsYeAAK</
FeatureInfo:ProdFeatureValueS0Id>
        <FeatureInfo:ProdId>01ti00000006iMl3AAE</FeatureInfo:ProdI
d>
        <FeatureInfo:Sequence>1</FeatureInfo:Sequence>
        <FeatureInfo:Value>No</FeatureInfo:Value>
    </FeatureInfo:Features>
    <FeatureInfo:Features>
        <FeatureInfo:FeatureS0Id>a1wi0000000jcV4AAI</FeatureInfo:
FeatureS0Id>
        <FeatureInfo:IsIncluded>>true</FeatureInfo:IsIncluded>
        <FeatureInfo:Name>AutoCPQ951Feature1</FeatureInfo:Name>
        <FeatureInfo:ProdFeatureValueS0Id>a1zi0000000WsYfAAK</
FeatureInfo:ProdFeatureValueS0Id>
        <FeatureInfo:ProdId>01ti00000006iMl3AAE</FeatureInfo:ProdI
d>
        <FeatureInfo:Sequence>2</FeatureInfo:Sequence>
        <FeatureInfo:Value>Yes</FeatureInfo:Value>
    </FeatureInfo:Features>
    <FeatureInfo:FeatureSetS0Id>a1vi0000000psEiAAI</FeatureInfo:F
eatureSetS0Id>
        <FeatureInfo:Name>AutoCPQ951FeatureSet</FeatureInfo:Name>
        <FeatureInfo:Sequence>1</FeatureInfo:Sequence>
    </FeatureInfo:ProductFeatureSet>
    <FeatureInfo:ProductFeatureS0Id xsi:nil="true"/>
</FeatureInfoResponseD0:FeatureInfos>
<FeatureInfoResponseD0:ProductId>01ti00000006iMl3AAE</
FeatureInfoResponseD0:ProductId>
    </result>
    <result>
        <FeatureInfoResponseD0:FeatureInfos>
            <FeatureInfo:ProductFeatureSet>
                <FeatureInfo:Description xsi:nil="true"/>
                <FeatureInfo:Features>
                    <FeatureInfo:FeatureS0Id>a1wi0000000jcV3AAI</FeatureInfo:
FeatureS0Id>
                    <FeatureInfo:IsIncluded>>true</FeatureInfo:IsIncluded>
                    <FeatureInfo:Name>AutoCPQ951Feature1</FeatureInfo:Name>
                    <FeatureInfo:ProdFeatureValueS0Id>a1zi0000000WsYjAAK</
FeatureInfo:ProdFeatureValueS0Id>
                    <FeatureInfo:ProdId>01ti00000006iMl8AAE</FeatureInfo:ProdI
d>
                    <FeatureInfo:Sequence>1</FeatureInfo:Sequence>
                    <FeatureInfo:Value>Yes</FeatureInfo:Value>

```

```

        </FeatureInfo:Features>
        <FeatureInfo:Features>
            <FeatureInfo:FeatureS0Id>a1wi0000000jcV4AAI</FeatureInfo:
FeatureS0Id>
            <FeatureInfo:IsIncluded>>true</FeatureInfo:IsIncluded>
            <FeatureInfo:Name>AutoCPQ951Feature1</FeatureInfo:Name>
            <FeatureInfo:ProdFeatureValueS0Id>a1zi0000000WsYkAAK</
FeatureInfo:ProdFeatureValueS0Id>
            <FeatureInfo:ProdId>01ti00000006iMl8AAE</FeatureInfo:ProdI
d>
            <FeatureInfo:Sequence>2</FeatureInfo:Sequence>
            <FeatureInfo:Value>No</FeatureInfo:Value>
        </FeatureInfo:Features>
        <FeatureInfo:FeatureSetS0Id>a1vi0000000psEiAAI</FeatureInfo:F
eatureSetS0Id>
            <FeatureInfo:Name>AutoCPQ951FeatureSet</FeatureInfo:Name>
            <FeatureInfo:Sequence>1</FeatureInfo:Sequence>
        </FeatureInfo:ProductFeatureSet>
        <FeatureInfo:ProductFeatureS0Id xsi:nil="true"/>
    </FeatureInfoResponseD0:FeatureInfos>
    <FeatureInfoResponseD0:ProductId>01ti00000006iMl8AAE</
FeatureInfoResponseD0:ProductId>
    </result>
</compareProductsResponse>
</soapenv:Body>
</soapenv:Envelope>

```

## Associating Constraint Rules to a Cart

This associates constraints rules to a cart.

API	Signature
associateConstraintRules	<i>webservice static void associateConstraintRules(Id cartId, List existingPrimaryNumbers)</i>

Parameters		
Name	Type	Description
cartId	ID	The id of the cart.
existingPrimaryNumbers	List<Integer>	This is a collection of primary numbers for which rules are already attached. When existing primary numbers are supplied, rules are associated with the new lines; whereas when null or empty collection is supplied, rules are attached to all the line items.

### Code Sample

The sample below enables you to associate a constraint rule to a product added as a line item in a cart. The sample below fetches the line numbers of the products in the cart and associates the constraint rule to the new line numbers of the cart. If the list object for primary line items is null, then the constraint rules are associated with all the products in the line items of the cart. For example if the delivery and packaging charge constraint rule is already applied to the existing products in the cart, you will provide a list of *collection of primary line items comprising selected products* to which the rules are already applied, thus enabling the delivery and packaging constraint rule to be associated with the new line items added to the cart.

```

1  public void associateConstraintRules()
2  {
3      List<Apttus_Config2__LineItem__c> liSOList = [select
Apttus_Config2__ProductID__c,
4      Apttus_Config2__LineNumber__c from Apttus_Config2__LineItem__c where
5      Apttus_Config2__ConfigurationId__c = :cartID];
6
7      acrList = new List<Apttus_Config2__LineItem__c>();
8
9      List<Integer> primaryLines = new List<Integer>();
10     for(Apttus_Config2__LineItem__c liS0: liSOList)
11     {
12         for(Apttus_Config2__LineItem__c acrS0: acrList)
13         {
14             if(acrS0.Id == liS0.Id)

```

```

15         {
16
17         primaryLines.Add(lis0.Apttus_Config2__LineNumber__c.intValue());
18         }
19     }
20
21     for(Apttus_Config2__LineItem__c lis0: lis0List)
22     {
23         for(LineItemWrapperClass objLineItemWrapperClass : lstWrapItems)
24         {
25             if(objLineItemWrapperClass.Selected)
26             {
27                 acrList.add(lis0);
28                 break;
29             }
30         }
31     }
32
33     Apttus_CPQApi.CPQWebService.associateConstraintRules(cartId,
34     primaryLines);
35 }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## API Prerequisites

- [Creating a Cart from a Quote](#)

## Response/Request XML

### Example Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cpq="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService">
```



```

<soapenv:Header>
  <cpq:SessionHeader>
    <cpq:sessionId>00D4P000000z7dE!
AQQAQGL6XYy.QsjqQQ6RTBnh.1ApTbiqkGAdVz9BS70lxobcyXgHHplmGXAE7p_cf6ziWJ8tpQt_4Q4Bi2VtY
eMyzjhaPbf0</cpq:sessionId>
  </cpq:SessionHeader>
</soapenv:Header>
<soapenv:Body>
  <cpq:associateConstraintRules>
    <cpq:cartId>all4P00000Bg6eAQAR</cpq:cartId>
    <!--Zero or more repetitions:-->
    <cpq:existingPrimaryNumbers>1</cpq:existingPrimaryNumbers>
  </cpq:associateConstraintRules>
</soapenv:Body>
</soapenv:Envelope>

```

### Example Response

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="ht
tp://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService">
  <soapenv:Body>
    <associateConstraintRulesResponse/>
  </soapenv:Body>
</soapenv:Envelope>

```

## Applying Constraint Rules to a Cart

This processes and applies all the constraint rules.

### Best Practice

It is recommended that you use the [associateConstraintRules API](#) before using this API. Directly using this API may not trigger the constraint rule.

API	Signature
applyConstraintRules	<i>webService static void applyConstraintRules(Id cartId, Boolean finalCheck)</i>

Parameters		
Name	Type	Description
cartId	ID	The id of the cart.
finalCheck	Boolean	If this is set to true, runs rules that are marked as check on finalization.

### Code Sample

The sample below enables you to apply constraint rules to a cart with a specific cart ID. If you set the finalCheck flag as true, the constraint rules are run when you finalize the cart. If the flag is set as false, the rules are run before cart finalization. For example, if you want to add a Shipping Costs constraint rule only after the cart is finalized, set the finalCheck as true and the shipping costs constraint rule is applied once the user finalized the cart. If you want to apply constraint rules before the cart is finalized, set finalCheck as false. For example, if you set the finalCheck flag as false, installation charges are added along with a product selected before the cart is finalized.

```

1  public void applyConstraintRules()
2  {
3      // For rules that are not marked as Check on Finalization
4      Apttus_CPQApi.CPQWebService.applyConstraintRules(cartID, false);
5
6      // For rules that are marked as Check on Finalization
7      //Apttus_CPQApi.CPQWebService.applyConstraintRules(cartID, true);
8  }
```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## API Prerequisites

- [Creating a Cart from a Quote](#)
- [Associating Constraint Rules to a Cart](#)

## Response/Request XML

### Example Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cpq="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService">
  <soapenv:Header>
    <cpq:SessionHeader>
      <cpq:sessionId>00D4P000000z7dE!
AQQAQFudWwmQHka0g8qt7T4KJ9MmTK0J0550XmfoUk9bUEL_idltBYg5muQuM4Pm0HVjinAgttLfi55uyxVSv
F5yrkoH.rH4</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <cpq:applyConstraintRules>
      <cpq:cartId>a1l4P00000Bg7CkQAJ</cpq:cartId>
      <cpq:finalCheck>>false</cpq:finalCheck>
    </cpq:applyConstraintRules>
  </soapenv:Body>
</soapenv:Envelope>
```

### Example Response

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService">
  <soapenv:Body>
    <applyConstraintRulesResponse/>
  </soapenv:Body>
</soapenv:Envelope>
```

## Applying Constraint Rules to Deleted Products

This runs the rules that are related to the deleted products.

API	Signature
<code>applyConstraintRulesOnDelete</code>	<i>webservice static void applyConstraintRulesOnDelete(Id cartId, List deletedProductIds)</i>

Parameters		
Name	Type	Description
cartId	ID	The id of the cart.
deletedProductIds	List<ID>	This is a list of product ids of the line item deleted from the cart.

### Code Sample

The sample below enables you to delete any constraint rules applied to a product that is deleted by a customer. For example, for all products you have added an inclusion constraint rule which states whenever you add a product, an installation charge is included along with the product. When the user deletes the product from the cart, this API enables you to delete the associated installation constraint rule applied to the product. Using the Product\_ID parameter you can specify the product from which the constraint rule is to be disassociated from.

```

1 Apttus_CPQApi.CPQWebService.applyConstraintRulesOnDelete(CART_ID, new
  String[] {PRODUCT_IDS});
    
```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## Response/Request XML

### Example Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cpq="
http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService">
  <soapenv:Header>
    <cpq:SessionHeader>
      <cpq:sessionId>00D4P000000z7dE!
AQQQAQHIR1W9.1as89Auz.CvNYxqyg56gLLWgUtP5VZxidvTsb1DpQZpmyDuqZOiF4VctBp3jhhJIxG9oRQ4A
4F9h98N0inT</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <cpq:applyConstraintRulesOnDelete>
      <cpq:cartId>a1l4P00000Bg7D9QAJ</cpq:cartId>
      <cpq:deletedProductIds>01t4P0000080fLL</cpq:deletedProductIds>
    </cpq:applyConstraintRulesOnDelete>
  </soapenv:Body>
</soapenv:Envelope>
```


### Example Response

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="ht
tp://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService">
  <soapenv:Body>
    <applyConstraintRulesOnDeleteResponse/>
  </soapenv:Body>
</soapenv:Envelope>
```

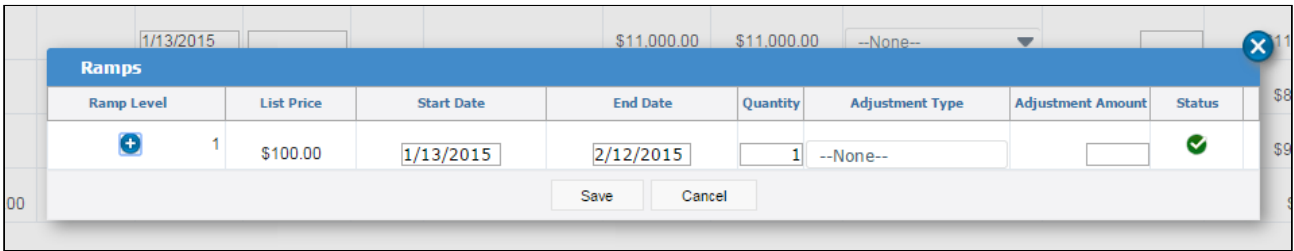
## Adding Price Ramps to a Cart (CPQ Web Service)

After you add a line item to a cart, this API enables you to add primary and secondary ramp line items for the line item. Once the ramp line items are created, you can also update the ramp line item details or delete ramp line item details using standard SOQL queries.

When you use CPQ out of the box, invoke the ramp using the red icon to the left of the primary line item.

 Product	Standard Price	\$100.00	\$100.00	1/13/2015	2/12/2015	<input type="text" value="1"/>	\$100.00	\$100.00	--None--	<input type="text" value=""/>	\$100	\$100.00	0.0000%	0.0000%
---	----------------	----------	----------	-----------	-----------	--------------------------------	----------	----------	----------	-------------------------------	-------	----------	---------	---------

Once you click the ramp icon, the ramp dialog appears:



The ramp dialog allows you to add, edit dates and quantity, make adjustments, save the changes, and cancel the changes.

Once the customer adds a ramp to a cart line item, they can do the following:

- Edit the start date, end date, quantity, adjustment type and adjustment amount based on the custom setting.
- The start date of a ramp line item defaults to the end date+1 of the previous line item.
- The end date of a ramp line item defaults to a date such that the difference between the start date and end date is the same as that of the previous line item.
- The user can add more ramp line items after or in between the ramp line items.
- The user can remove the new line before saving by clicking on the icon in the right most column.

Use the addMultiProducts API to add products to the cart.

This API adds one or more products (with default options) to the cart along with quantity, term, start date, and end date.

API	Signature	
addMultiProducts	<i>webService static Apttus_CPQApi.CPQ.AddMultiProductResponseDO                      addMultiProducts(Apttus_CPQApi.CPQ.AddMultiProductRequestDO request)</i>	
Parameters		
Name	Type	Description
request	Apttus_CPQApi.CPQ.AddMultiProductRequestDO	The request data object.

Request Data Object - Apttus_CPQApi.CPQ.AddMultiProductRequestDO		
Field	Type	Description
CartId	ID	The Id of the cart.
SelectedProducts	List <Apttus_CPQApi.CPQ.SelectedProductDO>	The list of selected product data objects.
Data Object - Apttus_CPQApi.CPQ.SelectedProductDO		
Field	Type	Description
AttributeValues	List	List of attributes values.
Comments	String	Comments associated with the record.
CopyBundleConfigurationFromSource	Boolean	You can use this to copy the configuration of the bundle from the source
CustomData	Apttus_Config2__LineItem__c	This can be used to include the list of custom fields you have added to the product.
CustomFields	List<String> CustomFields	List of custom fields created for your product.
EndDate	Date	The end date.
ProductId	ID	Id of the product bundle.
Quantity	Decimal	The bundle quantity.
RelatedLineItems	List	List of related line items for bundle
SellingTerm	Decimal	The bundle selling term.

Data Object - Apttus_CPQApi.CPQ.SelectedProductDO		
Field	Type	Description
SourceFields	List	List of the fields in the source bundle that you want to copy.
Sourceld	ID	ID of the source bundle.
StartDate	Date	The start date. You should ensure you use the correct date format.

Response Data Object - Apttus_CPQApi.CPQ.AddMultiProductResponseDO		
Field	Type	Description
LineNumbers	List<Decimal>	The list of line numbers added to the cart.

### Code Sample

The sample below enables you to add ramp line items after you have:

- Added Products to the cart using the AddMultiProducts APIs,
- Updated the Price for the added products using the updatePriceforCart API.
- Selected the products for which you want to add a ramp for.

Using the sample below you fetch the list of selected products for which you want to add a ramp. You also fetch the parameters for each of the selected products. For all the ramps you create, set PriceGroup as Price Ramp and PricingStatus as Pending. For a primary line item, set IsPrimaryLine\_\_c = true, IsPrimaryRampLine\_\_c = true, and PrimaryLineNumber\_\_c = 1.

```

1  public void createRampLineItems()
2  {
3      List<String> rampLineItems = new List<String>();
4
5      if(lstWrapItems.size() > 0)
6      {

```



```

7      // Create a list of selected products for which you want to create
a ramp for
8      for(LineItemWrapperClass objLineItemWrapperClass : lstWrapItems)
9      {
10         if(objLineItemWrapperClass.Selected)
11         {
12             rampLineItems.add(objLineItemWrapperClass.Name);
13         }
14     }
15
16     // Sort Ramp Line Items by name
17     rampLineItems.sort();
18
19     Integer rampLineItemIndex = 1;
20
21     for(String rampLineItemName : rampLineItems)
22     {
23         // Get Line Item parameters for the selected products
24         Apttus_Config2__LineItem__c lineItem = [SELECT
Apttus_Config2__ItemSequence__c, Apttus_Config2__PricingStatus__c,
Apttus_Config2__PriceGroup__c,
25         Apttus_Config2__IsPrimaryRampLine__c,
Apttus_Config2__IsPrimaryLine__c, Apttus_Config2__LineNumber__c,
Apttus_Config2__PrimaryLineNumber__c from Apttus_Config2__LineItem__c
WHERE
26         Name=:rampLineItemName];
27
28         //Set the parameters for each of the line items
29         lineItem.Apttus_Config2__PriceGroup__c = 'Price Ramp';
30         lineItem.Apttus_Config2__PricingStatus__c = 'Pending';
31         lineItem.Apttus_Config2__LineNumber__c = 1;
32         lineItem.Apttus_Config2__PrimaryLineNumber__c = 1;
33         lineItem.Apttus_Config2__ItemSequence__c = rampLineItemIndex;
34
35         //For a primary line item set the following
36         if(rampLineItemIndex == 1)
37         {
38             lineItem.Apttus_Config2__IsPrimaryLine__c = true;
39             lineItem.Apttus_Config2__IsPrimaryRampLine__c = true;
40         }
41
42         //For all secondary line items set the following parameters
43         else
44         {
45             lineItem.Apttus_Config2__IsPrimaryLine__c = false;

```

```

46         lineItem.Apttus_Config2__IsPrimaryRampLine__c = false;
47     }
48
49     // Update Line Items
50     update lineItem;
51
52     rampLineItemIndex++;
53 }
54 }
55 else
56 {
57     ApexPages.addMessage(new
58     ApexPages.Message(ApexPages.severity.info, 'No line items available.));
59 }

```

## Computing the Net Price for a Bundle

This API is used to calculate the price for the individual and summary line items of the cart. It runs the pricing rules, calculate bundle, line item, option level net prices and update the cart line with calculated prices. After the pricing is calculated for the cart line items, the resulting information is stored in the *Line Item* and *Summary Group* objects.

This API will also apply all the relevant rules and calculate the pricing including the following among others:

- Line Item level discount
- Total Price based on the specified quantity
- Applied Tiered Pricing
- Apply Ramp Pricing
- Apply Contractual Pricing
- Apply Related Pricing Rules

API	Signature
<b>computeNetPriceForBundle</b>	<i>webservice static Apttus_CPQApi.CPQ.ComputeNetPriceResponseDO computeNetPriceForBundle(Apttus_CPQApi.CPQ.ComputeNetPriceRequestDO request)</i>

Parameters		
Name	Type	Description
request	Apttus_CPQApi.CPQ.ComputeNetPriceRequestDO	The request data object.

Request Data Object - Apttus_CPQApi.CPQ.ComputeNetPriceRequestDO		
Field	Type	Description
CartId	ID	The Id of the cart.
LineNumber	Decimal	The bundle line number.

Response Data Object - Apttus_CPQApi.CPQ.ComputeNetPriceResponseDO		
Field	Type	Description
IsSuccess	Boolean	Indicates whether getting the total net price was successful.

### Code Sample

The sample below prompts the user with the appropriate messages if a cart does not exist or when no products exist in the cart. If the user clicks calculate net price and products exist in the cart, the API computes the net price for all the line numbers of products in the cart with a specific cartID. For example, if the user wants to check the aggregate price of all the products in the cart, you can create a **Total Net Price** button on your cart page and invoke the `computeNetPriceForBundle()` API on click of the button. The API will calculate the total price taking into account any discount or pricing rules associated with the line items in the cart.

```

1  public void computeNetPriceForBundle()
2  {
3      //If a cart does not exist, there is no cartID. Show the following
message to the user.
4      if(cartId == null)
5      {
6          ApexPages.addMessage(new
ApexPages.Message(ApexPages.severity.info, 'Please create cart.));
7      }
8
9      //If no products are added to the cart, the lineNumber of the cart is
null. Prompt the use to add products to the cart.
10     else if(lineNumber == null)
11     {
12         ApexPages.addMessage(new
ApexPages.Message(ApexPages.severity.info, 'Please add a bundle to the
cart.));
13     }
14     else
15     {
16         //Selected products are added to the cart. The request object
comprises the id of the cart and the line number in the cart
17         for(Decimal line: lineNumber)
18         {
19             Apttus_CPQApi.CPQ.ComputeNetPriceRequestDO request = new
Apttus_CPQApi.CPQ.ComputeNetPriceRequestDO();
20             request.CartId = cartId;
21             request.LineNumber = line;
22
23             Apttus_CPQApi.CPQ.ComputeNetPriceResponseDO priceResponse =
Apttus_CPQApi.CPQWebService.computeNetPriceForBundle(request);
24             ApexPages.addMessage(new
ApexPages.Message(ApexPages.severity.info, 'Success: ' +
priceResponse.IsSuccess));
25         }
26     }
27 }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

### API Prerequisites

- [Creating a Cart from a Quote](#)
- [Adding Products to a Cart](#)
- [Adding a Bundle to a Cart](#)

### Response/Request XML

#### Example Request

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:cpq="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:cpq1="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ">
  <soapenv:Header>
    <cpq:SessionHeader>
      <cpq:sessionId>00D4P000000z7dE!
AQQQAHIR1W9.1as89Auz.CvNYxqyg56gLLWgUtP5VZxidvTsb1DpQZpmyDuqZOiF4VctBp3jhhJIxG9oRQ4A
4F9h98N0inT</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <cpq:computeNetPriceForBundle>
      <cpq:request>
        <cpq1:CartId>a1L4P00000Bg7D9QAJ</cpq1:CartId>
        <cpq1:LineNumber>5</cpq1:LineNumber>
      </cpq:request>
    </cpq:computeNetPriceForBundle>
  </soapenv:Body>
</soapenv:Envelope>
```

```

Example Response

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="ht
tp://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
xmlns:ComputeNetPriceResponseDO="http://soap.sforce.com/schemas/class/Apttus_CPQApi/
CPQ">
  <soapenv:Body>
    <computeNetPriceForBundleResponse>
      <result>
        <ComputeNetPriceResponseDO:IsSuccess>true</ComputeNetPriceResponseDO:IsSu
ccess>
      </result>
    </computeNetPriceForBundleResponse>
  </soapenv:Body>
</soapenv:Envelope>
    
```

## Updating Price For A Cart

This API enables you to update the price for items in a given cart. Only line items in pending status are updated.

API	Signature
updatePriceForCart	<pre> webService static Apttus_CPQApi.CPQ.UpdatePriceResponseDO updatePriceForCart(Apttus_CPQApi.CPQ.UpdatePriceReques tDO request)                     </pre>

Parameters		
Name	Type	Description
Request	Apttus_CPQApi.CPQ.UpdatePriceReque stDO	The request data object.

Request Data Object - Apttus_CPQApi.CPQ.UpdatePriceRequestDO		
Field	Type	Description
CartId	ID	The Id of the cart to update the price for.

Response Data Object - Apttus_CPQApi.CPQ.UpdatePriceResponseDO		
Field	Type	Description
CompletedLineNumbers	List<Integer>	The list of line numbers representing line items that were updated in the current update price invocation.
IsPostPricePending	Boolean	Indicates whether the execution of post-price logic is pending.
IsPrePricePending	Boolean	Indicates whether the execution of pre-price logic is pending.
IsPricePending	Boolean	Indicates whether there are more items to update in future invocations. The caller needs to invoke the update price method in a separate transaction. This should be repeated until there are no more items pending.
IsTotalPricePending	Boolean	Indicates whether the calculation of the total price is pending.
PendingLineNumbers	List<Integer>	The list of line numbers in pending status after the method completes.

### Code Sample

The sample below enables you to update the net price in the cart for pending line items. In the sample below, with a single call of `updatePriceForCart` API, the API computes the net price for a single line item with status pending. `updatePriceForCart` API is invoked using `<apex:actionPoller>` command until compute net price operation is completed for all the

line items in the cart. For example, there are 10 line items in the cart and you invoke *updatePriceForCart* API 10 times at a regular interval of 5 seconds using `<apex:actionPoller>` command to complete the computing net price operation for all 10 line items.

```
1 <apex:actionPoller action="{!doUpdatePrice}" rerender="MessageID"
  interval="5" rendered="{!actionPollerActive}" enabled="{!
  actionPollerActive}"/> .
```

In the sample below the action poller is executed as soon as the page is loaded. The *hasPendingItems* flag returns false since no pending items have been added to the cart yet. If a valid cartID has been passed as an argument to the API and *isPricePending* returns true, run the action poller and invoke the *updatePriceforCart* API. The action poller does not run when *hasPendingItems*=false. You can invoke this API when the user adds discounts to the cart and clicks Reprice.

```
1 public void updatePriceforCart()
2 {
3     hasPendingItems = false;
4     actionPollerActive = true;
5
6     if(String.isBlank(cartId))
7     {
8         ApexPages.addMessage(new
9         ApexPages.Message(ApexPages.severity.info, 'Please create cart.));
10
11         objUpdatePriceRequestDO = new
12         Apttus_CpqApi.CPQ.UpdatePriceRequestDO();
13     }
14     else
15     {
16         // create the update price request
17         objUpdatePriceRequestDO = new
18         Apttus_CpqApi.CPQ.UpdatePriceRequestDO();
19
20         // add request parameters
21         objUpdatePriceRequestDO.CartId = cartId;
22
23         // update price for the cart
```



```

21     Apttus_CpqApi.CPQ.UpdatePriceResponseDO result =
Apttus_CpqApi.CPQWebService.updatePriceForCart(objUpdatePriceRequestDO);
22
23     hasPendingItems = result.IsPricePending;
24
25     // Start the action poller. If isPricePending=true returns true,
run the action poller and the updatePrice API is invoked again
26     if (hasPendingItems)
27     {
28         actionPollerActive = true;
29         flagUpdatePrice = true;
30     }
31 }
32 }
33
34 // invoke the following in a new transaction using action poller
35 public void doUpdatePrice()
36 {
37
38     // update price for the cart
39
40     if(String.isNotBlank(cartId) && flagUpdatePrice == true)
41     {
42         // create the update price request
43         objUpdatePriceRequestDO = new
Apttus_CpqApi.CPQ.UpdatePriceRequestDO();
44
45         // add request parameters
46         objUpdatePriceRequestDO.CartId = cartId;
47
48         Apttus_CpqApi.CPQ.UpdatePriceResponseDO result =
Apttus_CpqApi.CPQWebService.updatePriceForCart(objUpdatePriceRequestDO);
49
50         hasPendingItems = result.IsPricePending;
51         List<Integer> pendingLineNumbers = result.PendingLineNumbers;
52         List<Integer> completedLineNumbers = result.CompletedLineNumbers;
53
54         if( hasPendingItems == false && completedLineNumbers.size() != 0 )
55         {
56             // stop the action poller
57             actionPollerActive = false;
58         }
59         else if( hasPendingItems == true && pendingLineNumbers.size() > 0
60     )
    {

```

```

61         actionPollerActive = true;
62     }
63 }
64 }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## API Prerequisites

- [Creating a Cart from a Quote](#)
- [Adding Products to a Cart OR Adding a Bundle to a Cart](#)

## Response/Request XML

### Example Request

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:cpq="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:cpq1="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ">
  <soapenv:Header>
    <cpq:SessionHeader>
      <cpq:sessionId>00DZ000000NAEIA!
ASAAQLC3r1a0VIL_gk7v0A9K1n7dLEcCAANQHwNfXAPFKZEH292fAdeal5Nps8X2klNu98fcp6usnVFmzKARw
E99xq_dE8U6</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <cpq:updatePriceForCart>
      <cpq:request>
        <cpq1:CartId>a10Z0000002ZQVZ</cpq1:CartId>
      </cpq:request>
    </cpq:updatePriceForCart>
  </soapenv:Body>
</soapenv:Envelope>

```

**Example Response**

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:UpdatePriceResponseDO="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ">
  <soapenv:Body>
    <updatePriceForCartResponse>
      <result>
        <UpdatePriceResponseDO:CompletedLineNumbers>1</UpdatePriceResponseDO:CompletedLineNumbers>
        <UpdatePriceResponseDO:IsPostPricePending>>false</UpdatePriceResponseDO:IsPostPricePending>
      </result>
    </updatePriceForCartResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Price Breakup for a Cart or Specific Line Item

This method can be used to retrieve the price breakup for a cart or specific line item.

API	Signature
getPriceBreakup	<i>webservice static Apttus_CPQApi.CPQ.GetPriceBreakupResponseDO getPriceBreakup(Apttus_CPQApi.CPQ.GetPriceBreakupRequestDO request)</i>

Parameters		
Name	Type	Description
request	Apttus_CPQApi.CPQ.GetPriceBreakupRequestDO	The request data object.

Request Data Object - Apttus_CPQApi.CPQ.GetPriceBreakupRequestDO			
Name	Type	Required	Description
CartId	ID	Yes	The ID of the new cart object. Mandatory parameter.
LineItemId	ID	No	The ID of the line item. This can be null. If it is null the price breakup for all applicable line items in the cart is retrieved. If a value is provided, only the price breakup for that specific line item is retrieved.

Response Data Object - Apttus_CPQApi.CPQ.GetPriceBreakupResponseDO		
Name	Type	Description
HasPriceBreakups	Boolean	Indicates whether there are price breakups for the cart.
PriceBreakups	List <Apttus_CPQApi.CPQ.PriceBreakupCollDO>	The list of price breakup collection objects

Data Object - Apttus_CPQApi.CPQ.PriceBreakupCollDO		
Name	Type	Description
BreakupItems	List<Apttus_Config2__PriceBreakup__c>	The list of price breakup subjects associated with the line item.
LineItemId	ID	The ID of the line item associated with the price break subjects.

## Code Sample

Using the sample below you can enable the end-user to view the price breakup for a selected product. Suppose the user has selected a product Laptop, using the Price Breakup API, you can show the end-user the pricing breakup. For example, if the user has bought 25 laptops and for 20 quantities of the laptop a 5% discount is provided on the net price, and for the other 5 quantities, a discount of 10% is given. Using the *getPriceBreakup* API you can show the user the tiered pricing pattern applied to a product.

```

1  public void getPriceBreakup()
2  {
3      lstPriceBreakup = new List<PriceBreakUpWrapperClass>();
4      //For a selected product in a cart execute this loop.
5      for(LineItemWrapperClass objLineItemWrapperClass: lstWrapItems)
6      {
7          if(objLineItemWrapperClass.selected)
8          {
9              Apttus_CPQApi.CPQ.GetPriceBreakupRequestD0 priceBreakUpRequest
= new Apttus_CPQApi.CPQ.GetPriceBreakupRequestD0();
10             priceBreakUpRequest.CartId = cartId;
11             priceBreakUpRequest.LineItemId = objLineItemWrapperClass.Id;
12
13             //Pass the parameters to the API
14             Apttus_CPQApi.CPQ.GetPriceBreakupResponseD0
priceBreakUpResponse =
Apttus_CPQApi.CPQWebService.getPriceBreakup(priceBreakUpRequest);
15
16             //If the response returns true i.e. HasProducts=true execute
the loop below and fetch the priceBreakup list.
17             if(priceBreakUpResponse.HasPriceBreakups)
18             {
19                 lstPriceBreakUpCalls = new
List<Apttus_CPQApi.CPQ.PriceBreakupCollD0>();
20                 lstPriceBreakUpCalls = priceBreakUpResponse.PriceBreakups;
21
22                 for(Apttus_CPQApi.CPQ.PriceBreakupCollD0
objPriceBreakUpCall : lstPriceBreakUpCalls )
23                 {
24                     //priceBreakUpList = new
List<Apttus_Config2__PriceBreakup__c>();
25                     //priceBreakUpList = objPriceBreakUpCall.BreakupItems;
26
27                     String lineItemID = objPriceBreakUpCall.LineItemId;

```

```

28
29         List<Apttus_Config2__PriceBreakup__c>
    lIS0priceBreakUpList = [select id, Apttus_Config2__Sequence__c,
    Apttus_Config2__BreakupType__c,
30         Apttus_Config2__TierStartValue__c,
    Apttus_Config2__TierEndValue__c, Apttus_Config2__TierQuantity__c,
    Apttus_Config2__TierBasePrice__c, Apttus_Config2__TierExtendedPrice__c
    from Apttus_Config2__PriceBreakup__c where Apttus_Config2__LineItemId__c
    = :lineItemID];
31
32         for(Apttus_Config2__PriceBreakup__c
    lpriceBreakUpList : lIS0priceBreakUpList)
33             {
34                 PriceBreakUpWrapperClass
    objPriceBreakUpWrapperClass = new PriceBreakUpWrapperClass();
35                 objPriceBreakUpWrapperClass.BreakUpID =
    lpriceBreakUpList.Id;
36                 objPriceBreakUpWrapperClass.Sequence =
    lpriceBreakUpList.Apttus_Config2__Sequence__c;
37                 objPriceBreakUpWrapperClass.BreakupType =
    lpriceBreakUpList.Apttus_Config2__BreakupType__c;
38                 objPriceBreakUpWrapperClass.TierStartValue =
    lpriceBreakUpList.Apttus_Config2__TierStartValue__c;
39                 objPriceBreakUpWrapperClass.TierEndValue =
    lpriceBreakUpList.Apttus_Config2__TierEndValue__c;
40                 objPriceBreakUpWrapperClass.TierQty =
    lpriceBreakUpList.Apttus_Config2__TierQuantity__c;
41                 objPriceBreakUpWrapperClass.TierUnitPrice =
    lpriceBreakUpList.Apttus_Config2__TierBasePrice__c;
42                 objPriceBreakUpWrapperClass.TierUnitPrice =
    objPriceBreakUpWrapperClass.TierUnitPrice.setScale(2,
    System.RoundingMode.HALF_UP);
43                 objPriceBreakUpWrapperClass.TierExtendedPrice =
    lpriceBreakUpList.Apttus_Config2__TierExtendedPrice__c;
44                 objPriceBreakUpWrapperClass.TierExtendedPrice =
    objPriceBreakUpWrapperClass.TierExtendedPrice.setScale(2,
    System.RoundingMode.HALF_UP);
45
46                 lstPriceBreakup.add(objPriceBreakUpWrapperClass);
47             }
48
49     }
50 }

```

```

51         //If the product selected has no price breakups show an
appropriate message.
52         else
53         {
54             ApexPages.addMessage(new
ApexPages.Message(ApexPages.severity.info, 'No Price Breakup Available. ')
);
55         }
56     }
57 }
58 }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## API Prerequisites

- [Creating a Cart from a Quote](#)

## Response/Request XML

### Example Request

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cpq="
http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService" xmlns:cpq1="http://
soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ">
  <soapenv:Header>
    <cpq:SessionHeader>
      <cpq:sessionId>00D4P000000z7dE!
AQQQAHIR1W9.1as89Auz.CvNYxqyg56gLLWgUtP5VZxidvTsb1DpQZpmyDuqZOiF4VctBp3jhhJIxG9oRQ4A
4F9h98N0inT</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <cpq:getPriceBreakup>
      <cpq:request>
        <cpq1:CartId>a1l4P00000Bg7DTQAZ</cpq1:CartId>
        <cpq1:LineItemId>a1V4P0000053bKq</cpq1:LineItemId>
      </cpq:request>
    </cpq:getPriceBreakup>
  </soapenv:Body>
</soapenv:Envelope>

```

```

    </cpq:getPriceBreakup>
  </soapenv:Body>
</soapenv:Envelope>

```

### Example Response

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="ht
tp://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
xmlns:GetPriceBreakupResponseD0="http://soap.sforce.com/schemas/class/Apttus_CPQApi/
CPQ" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <getPriceBreakupResponse>
      <result>
        <GetPriceBreakupResponseD0:HasPriceBreakups>true</
GetPriceBreakupResponseD0:HasPriceBreakups>
        <GetPriceBreakupResponseD0:PriceBreakups>
          <GetPriceBreakupResponseD0:BreakupItems xsi:type="Apttus_Config2__Pric
eBreakup__c">
            <Id>a1W4P000000FIAW0UAP</Id>
            <Apttus_Config2__BreakupType__c>Tier Price</
Apttus_Config2__BreakupType__c>
            <Apttus_Config2__LineItemId__c>a1V4P00000053bKqUAI</
Apttus_Config2__LineItemId__c>
            <Apttus_Config2__Sequence__c>1</Apttus_Config2__Sequence__c>
            <Apttus_Config2__TierBasePrice__c>2204.25000</
Apttus_Config2__TierBasePrice__c>
            <Apttus_Config2__TierEndValue__c>10.00000</
Apttus_Config2__TierEndValue__c>
            <Apttus_Config2__TierQuantity__c>10.00000</
Apttus_Config2__TierQuantity__c>
            <Apttus_Config2__TierStartValue__c>0.00000</
Apttus_Config2__TierStartValue__c>
            <Name>PB-000000084</Name>
          </GetPriceBreakupResponseD0:BreakupItems>
          <GetPriceBreakupResponseD0:BreakupItems xsi:type="Apttus_Config2__Pric
eBreakup__c">
            <Id>a1W4P000000FIAW1UAP</Id>
            <Apttus_Config2__BreakupType__c>Tier Price</
Apttus_Config2__BreakupType__c>
            <Apttus_Config2__LineItemId__c>a1V4P00000053bKqUAI</
Apttus_Config2__LineItemId__c>

```



```

        <Apttus_Config2__Sequence__c>2</Apttus_Config2__Sequence__c>
        <Apttus_Config2__TierBasePrice__c>1469.50000</
Apttus_Config2__TierBasePrice__c>
        <Apttus_Config2__TierEndValue__c>25.00000</
Apttus_Config2__TierEndValue__c>
        <Apttus_Config2__TierQuantity__c>15.00000</
Apttus_Config2__TierQuantity__c>
        <Apttus_Config2__TierStartValue__c>10.00000</
Apttus_Config2__TierStartValue__c>
        <Name>PB-00000085</Name>
    </GetPriceBreakupResponseDO:BreakupItems>
    <GetPriceBreakupResponseDO:BreakupItems xsi:type="Apttus_Config2__Pric
eBreakup__c">
        <Id>a1W4P00000FIAW2UAP</Id>
        <Apttus_Config2__BreakupType__c>Tier Price</
Apttus_Config2__BreakupType__c>
        <Apttus_Config2__LineItemId__c>a1V4P0000053bKqUAI</
Apttus_Config2__LineItemId__c>
        <Apttus_Config2__Sequence__c>3</Apttus_Config2__Sequence__c>
        <Apttus_Config2__TierBasePrice__c>2939.00000</
Apttus_Config2__TierBasePrice__c>
        <Apttus_Config2__TierEndValue__c>40.00000</
Apttus_Config2__TierEndValue__c>
        <Apttus_Config2__TierQuantity__c>15.00000</
Apttus_Config2__TierQuantity__c>
        <Apttus_Config2__TierStartValue__c>25.00000</
Apttus_Config2__TierStartValue__c>
        <Name>PB-00000086</Name>
    </GetPriceBreakupResponseDO:BreakupItems>
    <GetPriceBreakupResponseDO:BreakupItems xsi:type="Apttus_Config2__Pric
eBreakup__c">
        <Id>a1W4P00000FIAW3UAP</Id>
        <Apttus_Config2__BreakupType__c>Total</
Apttus_Config2__BreakupType__c>
        <Apttus_Config2__LineItemId__c>a1V4P0000053bKqUAI</
Apttus_Config2__LineItemId__c>
        <Apttus_Config2__NetUnitPrice__c>2204.25000</
Apttus_Config2__NetUnitPrice__c>
        <Apttus_Config2__Sequence__c>4</Apttus_Config2__Sequence__c>
        <Apttus_Config2__TierBasePrice__c>2204.25000</
Apttus_Config2__TierBasePrice__c>
        <Apttus_Config2__TierQuantity__c>40.00000</
Apttus_Config2__TierQuantity__c>
        <Name>PB-00000087</Name>

```

```

        </GetPriceBreakupResponseDO:BreakupItems>
        <GetPriceBreakupResponseDO:LineItemId>a1V4P0000053bKqUAI</
GetPriceBreakupResponseDO:LineItemId>
        </GetPriceBreakupResponseDO:PriceBreakups>
    </result>
</getPriceBreakupResponse>
</soapenv:Body>
</soapenv:Envelope>
    
```

## Removing a Bundle from a Cart

This removes a product bundle and related line items and options from the cart.

API	Signature
removeBundle	<pre> webService static Apttus_CPQApi.CPQ.RemoveBundleResponseDO removeBundle(Apttus_CPQApi.CPQ.RemoveBundleRequestDO request)                     </pre>

Parameters		
Name	Type	Description
request	Apttus_CPQApi.CPQ.RemoveBundleRequestDO	The request data object.

Request Data Object - Apttus_CPQApi.CPQ.RemoveBundleRequestDO		
Field	Type	Description
CartId	ID	The Id of the cart.

Request Data Object - Apttus_CPQApi.CPQ.RemoveBundleRequestDO		
Field	Type	Description
LineNumber	Decimal	The bundle line number.

Response Data Object - Apttus_CPQApi.CPQ.RemoveBundleResponseDO		
Field	Type	Description
IsSuccess	Boolean	Indicates the success of the operation.

### Code Sample

The sample below enables you remove a selected bundle from the cart using the line number and a cart id. Invoke the API, when the user selects a bundle from the cart and click Remove or Delete. Based on the line number and the cartID the asset is removed from the cart. Update the new line items using the getLineItems function.

```

1  /**
2   * The below method demonstrates how to remove a bundle product from an
3   * existing cart (every quote has a cart)
4   * Lets assume the Quote's Cart has a bundle product called Laptop and its
5   * two options are Keyboard and Mouse
6   * The input of this method is Quote Number and the line number of the
7   * Laptop bundle product
8   */
9  public static void removeBundle(String quoteNumber, Integer lineNumber)
10 {
11     List<Apttus_Config2__ProductConfiguration__c> cart = [SELECT Id FROM
12     Apttus_Config2__ProductConfiguration__c WHERE
13     Apttus_QPConfig__ProposalId__r.Name = :quoteNumber LIMIT 1];
14
15     if(!cart.isEmpty() && lineNumber != null)
16     {

```

```

14         // Create the request object
15         Apttus_CPQApi.CPQ.RemoveBundleRequestDO request = new
Apttus_CPQApi.CPQ.RemoveBundleRequestDO();
16         request.CartId = cart.get(0).Id;
17         request.LineNumber = lineNumber;
18
19         // Execute the removeBundle routine
20         Apttus_CPQApi.CPQ.RemoveBundleResponseDO response =
Apttus_CPQApi.CPQWebService.removeBundle(request);
21
22         System.debug('Remove bundle from cart response status = ' +
response.IsSuccess);
23     }
24 }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## API Prerequisites

- [Creating a Cart from a Quote](#)
- [Adding a Bundle to a Cart](#)

## Response/Request XML

### Example Request

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cpq="
http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService" xmlns:cpq1="http://
soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ">
  <soapenv:Header>
    <cpq:SessionHeader>
      <cpq:sessionId>00D4P000000z7dE!
AQQAQFudWwmQHKaOg8qt7T4KJ9MmTK0J0550Xmf0Uk9bUEL_idltBYg5muQuM4Pm0HVjinAgttLfi55uyxVSv
F5yrkoH.rH4</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>

```

```

<soapenv:Body>
  <cpq:removeBundle>
    <cpq:request>
      <cpq1:CartId>a1l4P00000Bg7D9QAJ</cpq1:CartId>
      <cpq1:LineNumber>1</cpq1:LineNumber>
    </cpq:request>
  </cpq:removeBundle>
</soapenv:Body>
</soapenv:Envelope>

```

### Example Response

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:RemoveBundleResponseDO="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ">
  <soapenv:Body>
    <removeBundleResponse>
      <result>
        <RemoveBundleResponseDO:IsSuccess>>true</RemoveBundleResponseDO:IsSuccess>
      </result>
    </removeBundleResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Removing Multiple Bundles from a Cart

This removes one or more products (with default options) or bundles from the cart along with quantity, term, start date, and end date.

API	Signature
removeMultiBundles	<pre> webService static Apttus_CPQApi.CPQ.RemoveMultiBundlesResponseDO removeMultiBundles(Apttus_CPQApi.CPQ.RemoveMultiBundlesRequestDO request) </pre>

Parameters		
Name	Type	Description
request	Apttus_CPQApi.CPQ.RemoveMultiBundlesRequestDO	The request data object.

Request Data Object - Apttus_CPQApi.CPQ.RemoveMultiBundlesRequestDO		
Field	Type	Description
CartId	ID	The Id of the cart.
LineNumber	List<Decimal>	The List of LineNumbers that need to be removed from the cart.

Response Data Object - Apttus_CPQApi.CPQ.RemoveMultiBundlesResponseDO		
Field	Type	Description
IsSuccess	Boolean	Indicates the success of the operation.

### Code Sample

Use the sample below to remove one or more products of the cart. When the user selects multiple products on the cart and clicks Delete or Remove, invoke this API. The multiple products are deleted from the cart. Based on the line number and the cartID the selected assets are removed from the cart. Update the new line items using the getLineItems function.

1	/**
---	-----

```

2   * The below method demonstrates how to remove multiple products from an
3   existing cart (every quote has a cart)
4   * Lets assume the Quote's Cart has a 3 products,
5   * Laptop is a bundle product (line number 1) and Monitor and Wifi Router
6   are standalone products (line number 2 and 3 respectively)
7   * The input of this method is Quote Number and the line numbers of the
8   Laptop bundle product and Monitor and Wifi Router are standalone products
9   */
10  public static void removeMultiBundles(String quoteNumber, List<Integer>
11  lineNumbers) {
12
13      List<Apttus_Config2__ProductConfiguration__c> cart = [SELECT Id FROM
14  Apttus_Config2__ProductConfiguration__c WHERE
15  Apttus_QPConfig__Proposald__r.Name = :quoteNumber LIMIT 1];
16
17      if(!cart.isEmpty() && lineNumbers != null && !lineNumbers.isEmpty()) {
18
19          // Create the request object
20          Apttus_CPQApi.CPQ.RemoveMultiBundlesRequestDO request = new
21  Apttus_CPQApi.CPQ.RemoveMultiBundlesRequestDO();
22          request.CartId = cart.get(0).Id;
23          request.LineNumbers = lineNumbers;
24
25          // Execute the removeMultiBundles routine
26          Apttus_CPQApi.CPQ.RemoveMultiBundlesResponseDO response =
27  Apttus_CPQApi.CPQWebService.removeMultiBundles(request);
28
29          System.debug('Remove bundle from cart response status = ' +
30  response.IsSuccess);
31      }
32  }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## Response/Request XML

### Example Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cpq="
http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService" xmlns:cpq1="http://
soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ">
  <soapenv:Header>
    <cpq:SessionHeader>
      <cpq:sessionId>00D4P000000z7dE!
AQQQAHIR1W9.1as89Auz.CvNYxqyg56gLLWgUtP5VZxidvTsb1DpQZpmyDuqZ0iF4VctBp3jhhJIxG9oRQ4A
4F9h98N0inT</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <cpq:removeMultiBundles>
      <cpq:request>
        <cpq1:CartId>a1l4P00000Bg7D9QAJ</cpq1:CartId>
        <cpq1:LineNumbers>1</cpq1:LineNumbers>
        <cpq1:LineNumbers>3</cpq1:LineNumbers>
      </cpq:request>
    </cpq:removeMultiBundles>
  </soapenv:Body>
</soapenv:Envelope>
```

### Example Response

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="ht
tp://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
xmlns:RemoveMultiBundlesResponseD0="http://soap.sforce.com/schemas/class/
Apttus_CPQApi/CPQ">
  <soapenv:Body>
    <removeMultiBundlesResponse>
      <result>
        <RemoveMultiBundlesResponseD0:IsSuccess>>true</
RemoveMultiBundlesResponseD0:IsSuccess>
      </result>
    </removeMultiBundlesResponse>
  </soapenv:Body>
```



```
</soapenv:Envelope>
```

## Removing Line Items from the Cart

This API enables you to remove specific line items from the Cart. This API is invoked when you click **Remove** action icon next to any product on the Cart.

API	Signature	
removeLineItems	<i>webService static Apttus_CPQApi.CPQ.RemoveLineItemsResponseDO removeLineItems(Apttus_CPQApi.CPQ.RemoveLineItemsRequestDO request)</i>	
Request Parameter		
Name	Type	Description
Request	Apttus_CPQApi.CPQ.RemoveLineItemsRequestDO	This is the request data object.
Request Data Object - Apttus_CPQApi.CPQ.RemoveLineItemsRequestDO		
Name	Type	Description
Cart ID	ID	The Id of the cart.
PrimaryLineNumbers	Integer	The line item numbers of the primary line items which have to be removed using this API.
Response Data Object - Apttus_CPQApi.CPQ.RemoveLineItemsResponseDO		
Name	Type	Description
IsSuccess	Boolean	Indicates whether the operation of removing the line items was a success. True indicates success, false indicates that the line items were not removed from the Cart.

### Code sample

You can use this API after you realize that the end-user has added some products (bundle, standalone, or option) and you want to remove those products from the Cart at once.

Along with the primary line items, this API also removes the child line items under these primary line items (if applicable).

```

/**
 * The below method demonstrates how to remove multiple line items from an existing
 * cart (every quote has a cart)
 * Lets assume the Quote's Cart has 3 products,
 * a bundle product called Laptop (line number 1) and its two options are Keyboard
 * and Mouse
 * and Monitor and Wifi Router are standalone products (line number 2 and 3
 * respectively)
 * The input of this method is Quote Number and the line numbers of the Laptop bundle
 * product and Monitor standalone product
 */
public static void removeLineItems(String quoteNumber, List<Integer>
primaryLineNumbers)
{
    List<Apttus_Config2__ProductConfiguration__c> cart = [SELECT Id FROM
Apttus_Config2__ProductConfiguration__c WHERE Apttus_QPConfig__ProposalId__r.Name
= :quoteNumber LIMIT 1];

    if(!cart.isEmpty() && primaryLineNumbers != null && !
primaryLineNumbers.isEmpty())
    {
        // Create the request object
        Apttus_CPQApi.CPQ.RemoveLineItemsRequestDO request = new
Apttus_CPQApi.CPQ.RemoveLineItemsRequestDO();
        request.CartId = cart.get(0).Id;
        request.PrimaryLineNumbers = primaryLineNumbers;

        // Execute the removeLineItems routine
        Apttus_CPQApi.CPQ.RemoveLineItemsResponseDO response =
Apttus_CPQApi.CPQWebService.removeLineItems(request);

        System.debug('Remove line items from cart response status = ' +
response.IsSuccess);
    }
}

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

### API Prerequisites

- [Creating a Cart from a Quote](#)
- [Adding Products to a Cart OR Adding a Bundle to a Cart](#)

### Response/Request XML

#### Example Request

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:cpq="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:cpq1="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ">
  <soapenv:Header>
    <cpq:SessionHeader>
      <cpq:sessionId>00DZ000000NAEIA!
ASAAQGYIEG86MxZJB2SAYA0.SoCSmGG7G3SzeD6x2lnQ_zQCcBu.SpLUCrn0Bk0Ph00tnf_OAAnE0nfpZfkaS
9mA_1G_1kmP</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <cpq:removeLineItems>
      <cpq:request>
        <cpq1:CartId>a10Z0000002YqyaMAC</cpq1:CartId>
        <cpq1:PrimaryLineNumbers>2</cpq1:PrimaryLineNumbers>
        <cpq1:PrimaryLineNumbers>7</cpq1:PrimaryLineNumbers>
      </cpq:request>
    </cpq:removeLineItems>
  </soapenv:Body>
</soapenv:Envelope>
```

#### Example Response

```
<soapenv:Envelope
```

```

xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
xmlns:RemoveLineItemsResponseD0="http://soap.sforce.com/schemas/class/
Apttus_CPQApi/CPQ">
  <soapenv:Body>
    <removeLineItemsResponse>
      <result>
        <RemoveLineItemsResponseD0:IsSuccess>true</RemoveLineItemsResponseD0:
IsSuccess>
      </result>
    </removeLineItemsResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Removing Options from a Bundle

This API removes options from the bundle.

API	Signature
removeOptions	<i>webService static Apttus_CPQApi.CPQ.RemoveOptionsResponseDO removeOptions(Id cartId, Integer lineNumber, List componentIds)</i>

Request Parameter		
Name	Type	Description
cartID	ID	The ID of the cart the bundle belongs to.
lineNumber	Integer	The line number of the bundle line items.
componentIds	List<ID>	The list of component IDs of the relevant option line items

Response Data Object - Apttus_CPQApi.CPQ.RemoveOptionsResponseDO		
Field	Type	Description
IsSuccess	Boolean	Indicated whether the removal was successful. The values returned are <i>true</i> and <i>false</i> .
RemovedIds	List<ID>	The IDs of removed option line items.

### Code Sample

The sample code below enables you to remove options from the bundle and get the removal status and list of Ids of the removed options.

```

1  /**
2   * The below method demonstrates how to remove options from bundle
3   */
4  Public void deleteOptions(Id cartId, Integer lineNumber, List<Id>
componentIds) {
5      Apttus_CPQApi.CPQ.RemoveOptionsResponseDO response =
Apttus_CPQApi.CPQ.RemoveOptionsResponseDO();
6      Response = Apttus_CPQApi.CPQWebService.removeOptions(cartId,
lineNumber, componentIds);
7      System.debug('Has removed? ' + Response.IsSuccess);
8      System.debug('Options removed ' + Response.RemovedIds);
9  }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

### API Prerequisites

- [Query Product Option Component Id](#)

## Response/Request XML

### Example Request

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:cpq="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService">
  <soapenv:Header>
    <cpq:SessionHeader>
      <cpq:sessionId>00DZ000000NAEIA!
ASAAQHmIRqgn4R90i1yQjWTIVk4UZmsDe_.eK0Z9z6qLij7Tu.L_Yo8dRA_p80mhKMeRs4uzCZTadIgQ9fEbD
KciXEQYRyaA</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <cpq:removeOptions>
      <cpq:cartId>a10Z0000002ZQVZ</cpq:cartId>
      <cpq:lineNumber>1</cpq:lineNumber>
      <cpq:componentIds>a1YZ0000003pIGMA2</cpq:componentIds>
    </cpq:removeOptions>
  </soapenv:Body>
</soapenv:Envelope>
```

### Example Response

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:RemoveOptionsResponseD0="http://soap.sforce.com/schemas/class/
Apttus_CPQApi/CPQ">
  <soapenv:Body>
    <removeOptionsResponse>
      <result>
        <RemoveOptionsResponseD0:IsSuccess>true</RemoveOptionsResponseD0:IsSu
ccess>
        <RemoveOptionsResponseD0:RemovedIds>a19Z0000006sPeBIAU</
RemoveOptionsResponseD0:RemovedIds>
      </result>
    </removeOptionsResponse>
```

```
</soapenv:Body>
</soapenv:Envelope>
```

## Retrieving Constraint Rules Results

This API retrieves the constraints rules applied to a cart.

API	Signature
getConstraintRuleResult	<pre>webService static Apttus_CPQApi.CPQ.ConstraintResultDO getConstraintRuleResult(Id cartId)</pre>

Parameters		
Name	Type	Description
CartId	ID	The cart ID

Response Data Object - Apttus_CPQApi.CPQ.ConstraintResultDO		
Field	Type	Description
CartId	ID	The Id of the cart.
ConstraintRuleActions	List<Apttus_CPQApi.CPQ.AppliedActionDO>	The list of constraint rule action applied to the cart.
HasPendingError	Boolean	Indicates whether there are pending rule actions that are error type.
HasPendingWarning	Boolean	Indicates whether there are pending rule actions that are warning type.
NeedMoreProcessing	Boolean	Indicates whether the rule required more processing.

Response Data Object - Apttus_CPQApi.CPQ.AppliedActionDO		
Field	Type	Description
ActionIntent	String	The action intent of the applied constraint rule.
ActionType	String	The action type of the applied constraint rule.
AffectedPrimaryNumbers	List	The primary line number of the line item the constraint rule is applied on.
AffectedProductIds	List	IDs of the line item the constraint rule is applied on.
AppliedActionId	ID	The ID of the constraint rule action applied on the cart.
AppliedRuleInfoId	ID	The Applied Rule Info ID of the constraint rule action
CartId	ID	The ID of the cart.
ConstraintRuleActionId	ID	The ID of the constraint rule action.
CriteriaFields	List	The criteria fields in the constraint rule action.
IsAutoExecuted	Boolean	Indicates whether the auto-execution is enabled for the constraint rule.
IsHideMessage	Boolean	Indicates whether the messages are hidden.
IsIgnoredByUser	Boolean	Indicates whether ignoring the constraint rule is allowed.
IsPending	Boolean	Indicates whether any action is pending from the user to execute the constraint rule.



Response Data Object - Apttus_CPQApi.CPQ.AppliedActionDO		
Field	Type	Description
IsShowPrompt	Boolean	Indicates whether prompt is required.
IsTargetOptionProduct	Boolean	Indicates whether the target include or exclude products should be present in the shopping cart as options.
IsTargetPrimaryProduct	Boolean	Indicates whether the include or exclude products should be present in the shopping cart as primary lines.
MatchCountRule	String	
Message	String	The message added to the constraint rule action.
MessageType	String	The type of the message defined in the constraint rule action.
SuggestedProductIds	List	The IDs of the product suggested by rule action definition to be included in the cart.
TargetBundleNumber	Integer	The parent bundle number of the included or excluded products. For the products that are not option, the number is set as zero which represents the cart as the parent.
TriggeringPrimaryNumbers	List	The primary numbers of the line items that have triggered the rule action.
TriggeringProductIds	List	The list of ids representing the products that caused the rule to trigger

## Code Sample

The sample below enables you to fetch the constraint rules applied to a cart with a specific Cart ID. Use this API when you want to view the constraint rules applied to a cart. For example, to generate a report about the constraint rules applied to a cart, you can

fetch the constraint rules using this API.

```

1  public void getConstraintRuleResult()
2  {
3      Integer numErrors = 0;
4
5      Apttus_CPQApi.CPQ.ConstraintResultDO constraintResult =
Apttus_CPQApi.CPQWebService.getConstraintRuleResult(cartID);
6
7      List<Apttus_CPQApi.CPQ.AppliedActionDO> appliedActionDOList =
constraintResult.ConstraintRuleActions;
8
9      for(Apttus_CPQApi.CPQ.AppliedActionDO
appliedActDO:appliedActionDOList)
10     {
11         ApexPages.addMessage(new
ApexPages.Message(ApexPages.severity.info, appliedActDO.Message));
12         if(appliedActDO.MessageType.equals('Error') &&
appliedActDO.IsPending)
13             {
14                 numErrors++;
15             }
16     }
17 }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## Response/Request XML

### Example Request

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cpq="
http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService">
  <soapenv:Header>
    <cpq:SessionHeader>

```

```

    <cpq:sessionId>00D4P000000z7dE!
AQQAQAHIR1W9.1as89Auz.CvNYxqyg56gLLWgUtP5VZxidvTsb1DpQZpmyDuqZOiF4VctBp3jhhJIxG9oRQ4A
4F9h98N0inT</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <cpq:getConstraintRuleResult>
      <cpq:cartId>a1l4P00000Bg7D9QAJ</cpq:cartId>
    </cpq:getConstraintRuleResult>
  </soapenv:Body>
</soapenv:Envelope>

```

### Example Response

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="ht
tp://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
xmlns:ConstraintResultD0="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <getConstraintRuleResultResponse>
      <result>
        <ConstraintResultD0:CartId xsi:nil="true"/>
        <ConstraintResultD0:ConstraintRuleActions>
          <ConstraintResultD0>ActionIntent>Show Message</ConstraintResultD0:Acti
onIntent>
          <ConstraintResultD0>ActionType>Exclusion</ConstraintResultD0:ActionTyp
e>
          <ConstraintResultD0:AffectedPrimaryNumbers>13</ConstraintResultD0:Affe
ctedPrimaryNumbers>
          <ConstraintResultD0:AffectedProductIds>01t4P00000080fIrQAK</
ConstraintResultD0:AffectedProductIds>
          <ConstraintResultD0:AppliedActionId>a0y4P000000GHs7nQAD</
ConstraintResultD0:AppliedActionId>
          <ConstraintResultD0:AppliedRuleInfoId>a0z4P000000JyUToQAN</
ConstraintResultD0:AppliedRuleInfoId>
          <ConstraintResultD0:CartId>a1l4P00000Bg7D9QAJ</ConstraintResultD0:Cart
Id>
          <ConstraintResultD0:ConstraintRuleActionId>a1H4P0000005JdfVUAS</
ConstraintResultD0:ConstraintRuleActionId>
          <ConstraintResultD0:IsAutoExecuted>>false</ConstraintResultD0:IsAutoExe
cuted>
          <ConstraintResultD0:IsHideMessage>>false</ConstraintResultD0:IsHideMess
age>

```

```

        <ConstraintResultDO:IsIgnoredByUser>false</ConstraintResultDO:IsIgnoredByUser>
        <ConstraintResultDO:IsPending>true</ConstraintResultDO:IsPending>
        <ConstraintResultDO:IsShowPrompt>false</ConstraintResultDO:IsShowPrompt>
        <ConstraintResultDO:IsTargetOptionProduct>true</ConstraintResultDO:IsTargetOptionProduct>
        <ConstraintResultDO:IsTargetPrimaryProduct>true</ConstraintResultDO:IsTargetPrimaryProduct>
        <ConstraintResultDO:MatchCountRule xsi:nil="true"/>
        <ConstraintResultDO:Message>The MSM410 Access Point is not compatible with the selected blade server</ConstraintResultDO:Message>
        <ConstraintResultDO:MessageType>Error</ConstraintResultDO:MessageType>
        <ConstraintResultDO:SuggestedProductIds>01t4P0000080fIrQAK</ConstraintResultDO:SuggestedProductIds>
        <ConstraintResultDO:TargetBundleNumber>0</ConstraintResultDO:TargetBundleNumber>
        <ConstraintResultDO:TriggeringPrimaryNumbers>14</ConstraintResultDO:TriggeringPrimaryNumbers>
        <ConstraintResultDO:TriggeringProductIds>01t4P0000080fIzQAK</ConstraintResultDO:TriggeringProductIds>
        </ConstraintResultDO:ConstraintRuleActions>
        <ConstraintResultDO:HasPendingError>true</ConstraintResultDO:HasPendingError>
        <ConstraintResultDO:HasPendingWarning>false</ConstraintResultDO:HasPendingWarning>
        <ConstraintResultDO:NeedMoreProcessing>false</ConstraintResultDO:NeedMoreProcessing>
    </result>
</getConstraintRuleResultResponse>
</soapenv:Body>
</soapenv:Envelope>

```

## Retrieving Incentives on the Cart

You can use this API to retrieve the incentives that your users want to apply automatically or manually to the products in their shopping cart or a particular bundle.

API	Signature
<b>getIncentivesForCart</b>	<i>webservice static Apttus_CPQApi.CPQ.GetIncentivesForCartResponseDO getIncentivesForCart(Apttus_CPQApi.CPQ.GetIncentivesForCartRequestDO request)</i>

Parameters		
Name	Type	Description
request	Apttus_CPQApi.CPQ.GetIncentivesForCartRequestDO	The request data object.
Request Data Object - Apttus_CPQApi.CPQ.GetIncentivesForCartRequestDO		
BundleLineItemid	Id	The Id of the bundle on which promotions are to be applied.
CartID	ID	The Id of the cart, containing products on which the promotions are to be applied.
InclCoupons	Boolean	Indicates whether the coupons are included in the response.
Response Data Object - Apttus_CPQApi.CPQ.GetIncentivesForCartResponseDO		
Field	Type	Description
AppliedIncentives	List <AppliedIncentives>	A list of incentives applied on the given bundle line item.
IncentiveCoupons	List <IncentiveCoupons>	A list of coupons available on the bundle line item.
IncentiveItems	List <IncentiveItems>	A list of incentives by line item.
Incentives	List<Incentives>	A list of incentives available for the line item.

Response Data Object - Apttus_CPQApi.CPQ.GetIncentivesForCartResponseDO		
Field	Type	Description
getAppliedIncentivesByItem()	Map	A list of incentives applied on a line item.
getCouponsByIncentive()	Map	A list of coupons in an incentive.
getIncentivesByItem()	Map	A list of incentives available for a line item.
Data Object - Apttus_Config2__Incentive__c		
Field	Type	Description
AccountScopeOper	Picklist	Consists of options such as Include and Exclude. These allow you to define a promotion by choosing to include or exclude an account.
AccountScope	Look up	Enables you to define promotions for a particular Account. While defining a promotion, you can choose to include or exclude the Account based on which the promotion is applied.
AccountTypeScopeOper	Picklist	Consists of options such as Include and Exclude. These allow you to define a promotion by choosing to include or exclude an account type.
AccountTypeScope	Look up	Enables you to define promotions for a particular Account type. While defining a promotion, you can choose to include or exclude the Account Type based on which the promotion is applied.

Data Object - Apttus_Config2__Incentive__c		
Field	Type	Description
Active	Boolean	Can be set to <i>Yes</i> or <i>No</i> . Set it to <i>Yes</i> when you want a promotion to be applied and active for the criteria you specify. The <b>Active</b> flag on the promotions will be set using a workflow linked to approvals similar to what you would do for any other custom object approval. For example, you can write a workflow rule to set the active status to true after the promotion is Approved (if approval is setup). Approvals set up would be similar to custom object approvals for any object.
Active	Boolean	Can be set to <i>Yes</i> or <i>No</i> . Set it to <i>Yes</i> when you want a promotion to be applied and active for the criteria you specify. The <b>Active</b> flag on the promotions will be set using a workflow linked to approvals similar to what you would do for any other custom object approval. For example, you can write a workflow rule to set the active status to true after the promotion is Approved (if approval is setup). Approvals set up would be similar to custom object approvals for any object.
AutoApply	Boolean	Can be set to <i>Yes</i> or <i>No</i> . Set it to <i>Yes</i> when you want a promotion to be automatically applied when a promotion criteria is fulfilled. Set it to <i>No</i> when you want the user to manually enter an Incentive Code to apply promotions.
CountryScopeOper	Picklist	Consists of options such as Include and Exclude. These allow you to control if you want to include or exclude a country for a particular promotion.

Data Object - Apttus_Config2__Incentive__c		
Field	Type	Description
CountryScope	Look up	Enables you to define promotions for a particular country. While defining a promotion you can choose to include or exclude a country. The country picklist must be defined for the Account object.
EffectiveDate	Date	Start Date from which the promotion is applicable.
ExpirationDate	Date	Date on which the promotion expires. It can be blank. It needs to be greater than the Start Date.
IncentiveCode	String	This is the code that is used for applying the incentive. This code is captured on the order or proposal line items when an incentive is applied to the line item. By default, this value is set to Incentive Number, but is editable. The marketing team can also use this for marketing campaigns. When a promotion is not auto-applied, (that is, when the <b>Auto Apply ?</b> check box is deselected), the user can use the incentive code to avail a promotional offer.
IncentiveNumber	Auto Number	System generated unique number for the Incentive. This differs from the Salesforce record id. When an active incentive is changed over time, the Incentive number remains same.
Name	String	User-specific name for the Incentive. This promotion name is displayed to the user wherever the Incentive information is displayed. For example, whenever a user looks-up the promotion information applicable to a specific product, the promotion name you provide here is displayed.



Data Object - Apttus_Config2__Incentive__c		
Field	Type	Description
PriceListScopeOper	Picklist	Based on the selected Price List, you can control whether to include or exclude that particular price list and its products.
PriceListScope	Look up	Consists of the price list for which you have defined promotions.
ProductFamilyScope	Look up	Consists of a product family for which you have defined promotions.
ProductFamilyScopeOper	Picklist	Based on the selected Product Family, you can control whether to include or exclude that particular product family.
ProductScopeOper	Picklist	Based on the selected Product, you can control whether to include or exclude that product.
ProductScope	Look up	Consists of the products for which you have defined promotions.
RegionScopeOper	Picklist	Consists of options such as Include and Exclude. These allow you to control if you want to include or exclude a region for a particular promotion.
RegionScope	Look up	Enables you to define promotions for a particular region. While defining a promotion, you can choose to include or exclude a region. You must define the region picklist for the Account object.
UseType	Picklist	This picklist indicates whether the incentive is of type Promotion or Rebate. In our case, select <i>Promotion</i> .
Sequence	Number	Defines the sequence in which incentives are applied, when there are more than one incentives.

Data Object - Apttus_Config2__Incentive__c		
Field	Type	Description
Status	Picklist	Specifies the status of the Promotion, which is dependent on the promotion life cycle. After a promotion is approved, you can set the status of the approval as active.  Approvals on Promotions: Admins can set up approval processes and other workflows on promotions. Marketing managers can send promotions for approval and approvers can view the promotion, approve or reject the promotions with or without comments, and activate it.
StopProcessingMoreIncentives	Boolean	Indicates if CPQ need not process other incentives if there are multiple incentives applicable at the same time.

### Code Sample

This code sample enables you to retrieve incentives and coupons available in cart or a line item. You can also retrieve incentives that are applied on the cart or a line item.

```
// get incentive for cart
Apttus_CPQApi.CPQ.GetIncentivesForCartRequestDO request = new
    Apttus_CPQApi.CPQ.GetIncentivesForCartRequestDO();
request.CartId = configS0.Id;
request.BundleLineItemId = bdLineItemS0.Id;
request.InclCoupons = true;
Apttus_CPQApi.CPQ.GetIncentivesForCartResponseDO result2 =
    Apttus_CPQApi.CPQWebService.getIncentivesForCart(request);

// incentives by item
Map<ID, List<Apttus_Config2_Incentive_c>> incentivesByItem =
    result2.getIncentivesByItem();
// applied incentives by item
```

```

Map<ID, List<Apttus_Config2_AdjustmentLineItem_c>> appliedIncentivesByItem =
result2.getAppliedIncentivesByItem();
// incentive coupons by incentive

Map<ID, List<Apttus_Config2_IncentiveCoupon_c>> couponsByIncentive =
result2.getCouponsByIncentive();

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## Response/Request XML

### Example Request

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cpq="
http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService" xmlns:cpq1="http://
soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ">
  <soapenv:Header>
    <cpq:SessionHeader>
      <cpq:sessionId>00DZ000000NAEIA!
ASAAQFRdZcUFQ8LrHGF_qYNN9deP0ee.07JRqgeS3IF8IILzILBdA0PySGSki f1VBtALP6pryVNOXfhn0faHO
pGsc9GFVLL0</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <cpq:getIncentivesForCart>
      <cpq:request>
        <cpq1:CartId>a10Z0000002YdG6</cpq1:CartId>
        <cpq1:InclCoupons>true</cpq1:InclCoupons>
      </cpq:request>
    </cpq:getIncentivesForCart>
  </soapenv:Body>
</soapenv:Envelope>

```

### Example Response

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"

```

```

xmlns="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
xmlns:GetIncentivesForCartResponseD0="http://soap.sforce.com/schemas/class/
Apttus_CPQApi/CPQ"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<soapenv:Body>
  <getIncentivesForCartResponse>
    <result>
      <GetIncentivesForCartResponseD0:Incentives xsi:type="Apttus_Config2__
Incentive__c">
        <Id>a400v00000005FtMEAU</Id>
        <Apttus_Config2__AccountScope0per__c>in</
Apttus_Config2__AccountScope0per__c>
        <Apttus_Config2__AccountScope__c>All</
Apttus_Config2__AccountScope__c>
        <Apttus_Config2__AccountTypeScope0per__c>in</
Apttus_Config2__AccountTypeScope0per__c>
        <Apttus_Config2__AccountTypeScope__c>All</
Apttus_Config2__AccountTypeScope__c>
        <Apttus_Config2__Active__c>true</Apttus_Config2__Active__c>
        <Apttus_Config2__ApplicationMethod__c>Buy X Get X</
Apttus_Config2__ApplicationMethod__c>
        <Apttus_Config2__AutoApply__c>false</Apttus_Config2__AutoApply__c>
        <Apttus_Config2__ContextType__c>Line Item</
Apttus_Config2__ContextType__c>
        <Apttus_Config2__CountryScope0per__c>in</
Apttus_Config2__CountryScope0per__c>
        <Apttus_Config2__CountryScope__c>All</
Apttus_Config2__CountryScope__c>
        <Apttus_Config2__EffectiveDate__c>2018-10-15</
Apttus_Config2__EffectiveDate__c>
        <Apttus_Config2__ExpirationDate__c>2025-10-31</
Apttus_Config2__ExpirationDate__c>
        <Apttus_Config2__IncentiveCode__c>ICT-002510</
Apttus_Config2__IncentiveCode__c>
        <Apttus_Config2__IncentiveNumber__c>ICT-002510.0</
Apttus_Config2__IncentiveNumber__c>
        <Apttus_Config2__PriceListScope0per__c>in</
Apttus_Config2__PriceListScope0per__c>
        <Apttus_Config2__PriceListScope__c>All</
Apttus_Config2__PriceListScope__c>
        <Apttus_Config2__ProductFamilyScope0per__c>in</
Apttus_Config2__ProductFamilyScope0per__c>

```

```

        <Apttus_Config2__ProductFamilyScope__c>All</
Apttus_Config2__ProductFamilyScope__c>
        <Apttus_Config2__ProductScope0per__c>in</
Apttus_Config2__ProductScope0per__c>
        <Apttus_Config2__ProductScope__c>All</
Apttus_Config2__ProductScope__c>
        <Apttus_Config2__RegionScope0per__c>in</
Apttus_Config2__RegionScope0per__c>
        <Apttus_Config2__RegionScope__c>All</
Apttus_Config2__RegionScope__c>
        <Apttus_Config2__Sequence__c>1</Apttus_Config2__Sequence__c>
        <Apttus_Config2__Status__c>Draft</Apttus_Config2__Status__c>
        <Apttus_Config2__StopProcessingMoreIncentives__c>true</
Apttus_Config2__StopProcessingMoreIncentives__c>
        <Apttus_Config2__SubUseType__c>Promotion</
Apttus_Config2__SubUseType__c>
        <Apttus_Config2__UseType__c>Promotion</Apttus_Config2__UseType__c>
        <Name>002I-Auto Single Rate Type Promotion</Name>
    </GetIncentivesForCartResponseD0:Incentives>
    <GetIncentivesForCartResponseD0:Incentives xsi:type="Apttus_Config2__
Incentive__c">
        <Id>a400v0000005FtHEAU</Id>
        <Apttus_Config2__AccountScope0per__c>in</
Apttus_Config2__AccountScope0per__c>
        <Apttus_Config2__AccountScope__c>All</
Apttus_Config2__AccountScope__c>
        <Apttus_Config2__AccountTypeScope0per__c>in</
Apttus_Config2__AccountTypeScope0per__c>
        <Apttus_Config2__AccountTypeScope__c>All</
Apttus_Config2__AccountTypeScope__c>
        <Apttus_Config2__Active__c>true</Apttus_Config2__Active__c>
        <Apttus_Config2__ApplicationMethod__c>Buy X Get X</
Apttus_Config2__ApplicationMethod__c>
        <Apttus_Config2__AutoApply__c>false</Apttus_Config2__AutoApply__c>
        <Apttus_Config2__ContextType__c>Line Item</
Apttus_Config2__ContextType__c>
        <Apttus_Config2__CountryScope0per__c>in</
Apttus_Config2__CountryScope0per__c>
        <Apttus_Config2__CountryScope__c>All</
Apttus_Config2__CountryScope__c>
        <Apttus_Config2__EffectiveDate__c>2018-10-14</
Apttus_Config2__EffectiveDate__c>
        <Apttus_Config2__ExpirationDate__c>2024-10-18</
Apttus_Config2__ExpirationDate__c>

```

```

        <Apttus_Config2__IncentiveCode__c>ICT-002509</
Apttus_Config2__IncentiveCode__c>
        <Apttus_Config2__IncentiveNumber__c>ICT-002509.0</
Apttus_Config2__IncentiveNumber__c>
        <Apttus_Config2__PriceListScope0per__c>in</
Apttus_Config2__PriceListScope0per__c>
        <Apttus_Config2__PriceListScope__c>All</
Apttus_Config2__PriceListScope__c>
        <Apttus_Config2__ProductFamilyScope0per__c>in</
Apttus_Config2__ProductFamilyScope0per__c>
        <Apttus_Config2__ProductFamilyScope__c>All</
Apttus_Config2__ProductFamilyScope__c>
        <Apttus_Config2__ProductScope0per__c>in</
Apttus_Config2__ProductScope0per__c>
        <Apttus_Config2__ProductScope__c>All</
Apttus_Config2__ProductScope__c>
        <Apttus_Config2__RegionScope0per__c>in</
Apttus_Config2__RegionScope0per__c>
        <Apttus_Config2__RegionScope__c>All</
Apttus_Config2__RegionScope__c>
        <Apttus_Config2__Sequence__c>1</Apttus_Config2__Sequence__c>
        <Apttus_Config2__Status__c>Draft</Apttus_Config2__Status__c>
        <Apttus_Config2__StopProcessingMoreIncentives__c>true</
Apttus_Config2__StopProcessingMoreIncentives__c>
        <Apttus_Config2__SubUseType__c>Promotion</
Apttus_Config2__SubUseType__c>
        <Apttus_Config2__UseType__c>Promotion</Apttus_Config2__UseType__c>
        <Name>002H-Auto Single Rate Type Promotion</Name>
    </GetIncentivesForCartResponseD0:Incentives>
</result>
</getIncentivesForCartResponse>
</soapenv:Body>
</soapenv:Envelope>

```

## Retrieving Asset Line Items

Gets the list of asset line item objects matching the parameters used as the search criteria.

API	Signature
<code>getAssetsForSearchText</code>	<pre>webservice static Apttus_CPQApi.CPQ.AssetSearchResultDO getAssetsForSearchText(Id accountId, List locationIds, String searchText)</pre>

 You cannot use the parameters *accountId* and *locationIds* simultaneously.

Parameters		
Name	Type	Description
<code>accountId</code>	ID	The id of the account associated with the asset.
<code>locationIds</code>	List <ID>	The ids of the different locations associated with the assets.
<code>searchText</code>	String	The search text.

Response Data Object - Apttus_CPQApi.CPQ.AssetSearchResultDO		
Name	Type	Description
<code>AssetItems</code>	List < Apttus_Config2__AssetLineItem__c >	This is a list of the asset line items. This returns the asset line item records and the values for the fields that belong to each record.
<code>HasAssetItems</code>	Boolean	Indicates whether there are asset line items, that match the parameters used for the search.

## Code Sample

When you generate a quote for an existing customer, it is possible that the customer has some assets associated with the account. You can provide a search field on the cart page that enables the customer to search for existing assets and view them. When the user types the search criteria for fetching the assets and clicks Search, invoke the API. This sample should enable the user to search a product by product name, product code, product description, category name, and configuration type. The search for get assets is similar to the product search in the catalog page and installed products page.

```

1  public void searchAssets()
2  {
3      Apttus_CPQApi.CPQ.AssetSearchResultDO assetResults =
Apttus_CPQApi.CPQWebService.getAssetsForSearchText(accountId, null,
assetSearchText);
4      List<AssetWrapperClass> lstAssetWrapper = new
List<AssetWrapperClass>();
5
6      // if assets items that are searched by a keyword exist (i.e
hasassetitems=true)
7      // and some value is returned, show the following as a part of the
list
8      if (assetResults.HasAssetItems == true)
9      {
10         List<Apttus_Config2__AssetLineItem__c> aliSOList =
assetResults.AssetItems;
11         for (Apttus_Config2__AssetLineItem__c aliSO : aliSOList)
12         {
13             AssetWrapperClass objAssetWrapperClass = new AssetWrapperClass
();
14             objAssetWrapperClass.AssetId = aliSO.ID;
15             objAssetWrapperClass.ProductId =
aliSO.Apttus_Config2__Description__c;
16             objAssetWrapperClass.Quantity =
aliSO.Apttus_Config2__Quantity__c;
17             objAssetWrapperClass.StartDate =
aliSO.Apttus_Config2__StartDate__c;
18             objAssetWrapperClass.EndDate =
aliSO.Apttus_Config2__EndDate__c;
19             lstAssetWrapper.add(objAssetWrapperClass);
20         }
21     }
22
23     // if there are no assets by the search text that the user entered,

```



```

24     // display an appropriate message.
25     else
26     {
27         lstAssetWrapper.clear();
28         ApexPages.addMessage(new
ApexPages.Message(ApexPages.severity.info, 'No result found. Please try
again. '));
29     }
30 }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## Response/Request XML

### Example Request

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cpq="
http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService">
  <soapenv:Header>
    <cpq:SessionHeader>
      <cpq:sessionId>00D4P000000z7dE!
AQQAQGL6XYy.QsjqQQ6RTBnh.1ApTbiqkGAdVz9BS70lxobcyXgHHplmGXAE7p_cf6ziWJ8tpQt_4Q4Bi2VtY
eMyzjhaPbf0</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <cpq:getAssetsForSearchText>
      <cpq:accountId>0014P000028PFFS</cpq:accountId>
      <!--<cpq:locationIds></cpq:locationIds-->
      <cpq:searchText></cpq:searchText>
    </cpq:getAssetsForSearchText>
  </soapenv:Body>
</soapenv:Envelope>

```

**Example Response**

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
xmlns:AssetSearchResultD0="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <getAssetsForSearchTextResponse>
      <result>
        <AssetSearchResultD0:AssetItems xsi:type="Apttus_Config2__AssetLineItem__c">
          <Id>a114P000004ie6cQAA</Id>
          <Apttus_Config2__AccountId__c>0014P000028PFFSQA4</Apttus_Config2__AccountId__c>
          <Apttus_Config2__AssetStatus__c>Activated</Apttus_Config2__AssetStatus__c>
          <Apttus_Config2__BasePriceMethod__c>Per Unit</Apttus_Config2__BasePriceMethod__c>
          <Apttus_Config2__BasePrice__c>160.00000</Apttus_Config2__BasePrice__c>
          <Apttus_Config2__BusinessObjectId__c>a0c4P00000G3AikQAF</Apttus_Config2__BusinessObjectId__c>
          <Apttus_Config2__ChargeType__c>Standard Price</Apttus_Config2__ChargeType__c>
          <Apttus_Config2__Description__c>SecureCloud Enterprise</Apttus_Config2__Description__c>
          <Apttus_Config2__EndDate__c>2017-06-19</Apttus_Config2__EndDate__c>
          <Apttus_Config2__Frequency__c>Monthly</Apttus_Config2__Frequency__c>
          <Apttus_Config2__IsPrimaryLine__c>true</Apttus_Config2__IsPrimaryLine__c>
          <Apttus_Config2__IsPrimaryRampLine__c>false</Apttus_Config2__IsPrimaryRampLine__c>
          <Apttus_Config2__IsReadOnly__c>false</Apttus_Config2__IsReadOnly__c>
          <Apttus_Config2__IsUsageTierModifiable__c>false</Apttus_Config2__IsUsageTierModifiable__c>
          <Apttus_Config2__ItemSequence__c>1</Apttus_Config2__ItemSequence__c>
          <Apttus_Config2__LineNumber__c>1</Apttus_Config2__LineNumber__c>
          <Apttus_Config2__LineType__c>Product/Service</Apttus_Config2__LineType__c>
          <Apttus_Config2__MustUpgrade__c>false</Apttus_Config2__MustUpgrade__c>
          <Apttus_Config2__NetPrice__c>729600.00000</Apttus_Config2__NetPrice__c>
          <Apttus_Config2__PriceType__c>Recurring</Apttus_Config2__PriceType__c>

```

```

        <Apttus_Config2__ProductId__c>01t4P00000884AHQAY</
Apttus_Config2__ProductId__c>
        <Apttus_Config2__ProductId__r xsi:type="Product2">
            <Id>01t4P00000884AHQAY</Id>
            <Apttus_Config2__Customizable__c>>false</
Apttus_Config2__Customizable__c>
            <Apttus_Config2__HasAttributes__c>>true</
Apttus_Config2__HasAttributes__c>
            <Apttus_Config2__HasDefaults__c>>false</
Apttus_Config2__HasDefaults__c>
            <Apttus_Config2__HasOptions__c>>true</Apttus_Config2__HasOptions__c>
            <Family>CSD Products</Family>
        </Apttus_Config2__ProductId__r>
        <Apttus_Config2__Quantity__c>400.00000</Apttus_Config2__Quantity__c>
        <Apttus_Config2__SellingFrequency__c>Monthly</
Apttus_Config2__SellingFrequency__c>
        <Apttus_Config2__SellingTerm__c>12.00000</
Apttus_Config2__SellingTerm__c>
        <Apttus_Config2__StartDate__c>2016-06-20</Apttus_Config2__StartDate__c>
        <Name>SecureCloud Enterprise</Name>
    </AssetSearchResultDO:AssetItems>
    <AssetSearchResultDO:AssetItems xsi:type="Apttus_Config2__AssetLineItem__
c">
        <Id>a114P000004ie6dQAA</Id>
        <Apttus_Config2__AccountId__c>0014P0000028PFFSQA4</
Apttus_Config2__AccountId__c>
        <Apttus_Config2__AssetStatus__c>Activated</
Apttus_Config2__AssetStatus__c>
        <Apttus_Config2__BasePriceMethod__c>Per Unit</
Apttus_Config2__BasePriceMethod__c>
        <Apttus_Config2__BasePrice__c>131328.00000</
Apttus_Config2__BasePrice__c>
        <Apttus_Config2__BusinessObjectId__c>a0c4P00000G3AikQAF</
Apttus_Config2__BusinessObjectId__c>
        <Apttus_Config2__ChargeType__c>Subscription Fee</
Apttus_Config2__ChargeType__c>
        <Apttus_Config2__Description__c>SecureCloud Premier Support</
Apttus_Config2__Description__c>
        <Apttus_Config2__EndDate__c>2017-06-19</Apttus_Config2__EndDate__c>
        <Apttus_Config2__Frequency__c>Yearly</Apttus_Config2__Frequency__c>
        <Apttus_Config2__IsPrimaryLine__c>true</
Apttus_Config2__IsPrimaryLine__c>
        <Apttus_Config2__IsPrimaryRampLine__c>>false</
Apttus_Config2__IsPrimaryRampLine__c>

```

```

    <Apttus_Config2__IsReadOnly__c>>false</Apttus_Config2__IsReadOnly__c>
    <Apttus_Config2__IsUsageTierModifiable__c>>false</
Apttus_Config2__IsUsageTierModifiable__c>
    <Apttus_Config2__ItemSequence__c>1</Apttus_Config2__ItemSequence__c>
    <Apttus_Config2__LineNumber__c>2</Apttus_Config2__LineNumber__c>
    <Apttus_Config2__LineType__c>Product/Service</
Apttus_Config2__LineType__c>
    <Apttus_Config2__MustUpgrade__c>>false</Apttus_Config2__MustUpgrade__c>
    <Apttus_Config2__NetPrice__c>131328.00000</Apttus_Config2__NetPrice__c>
    <Apttus_Config2__PriceType__c>Recurring</Apttus_Config2__PriceType__c>
    <Apttus_Config2__ProductId__c>01t4P000000884tTQAQ</
Apttus_Config2__ProductId__c>
    <Apttus_Config2__ProductId__r xsi:type="Product2">
      <Id>01t4P000000884tTQAQ</Id>
      <Apttus_Config2__Customizable__c>>false</
Apttus_Config2__Customizable__c>
      <Apttus_Config2__HasAttributes__c>>false</
Apttus_Config2__HasAttributes__c>
      <Apttus_Config2__HasDefaults__c>>false</
Apttus_Config2__HasDefaults__c>
      <Apttus_Config2__HasOptions__c>>false</Apttus_Config2__HasOptions__c>
      <Family>CSD Products</Family>
    </Apttus_Config2__ProductId__r>
    <Apttus_Config2__Quantity__c>1.00000</Apttus_Config2__Quantity__c>
    <Apttus_Config2__SellingFrequency__c>Yearly</
Apttus_Config2__SellingFrequency__c>
    <Apttus_Config2__SellingTerm__c>1.00000</
Apttus_Config2__SellingTerm__c>
    <Apttus_Config2__StartDate__c>2016-06-20</Apttus_Config2__StartDate__c>
    <Name>SecureCloud Premier Support</Name>
  </AssetSearchResultDO:AssetItems>
  <AssetSearchResultDO:AssetItems xsi:type="Apttus_Config2__AssetLineItem__
c">
    <Id>a114P000004ie6eQAA</Id>
    <Apttus_Config2__AccountId__c>0014P0000028PFFSQA4</
Apttus_Config2__AccountId__c>
    <Apttus_Config2__AssetStatus__c>Activated</
Apttus_Config2__AssetStatus__c>
    <Apttus_Config2__BasePriceMethod__c>Per Unit</
Apttus_Config2__BasePriceMethod__c>
    <Apttus_Config2__BasePrice__c>500.00000</Apttus_Config2__BasePrice__c>
    <Apttus_Config2__BusinessObjectId__c>a0c4P00000G3AikQAF</
Apttus_Config2__BusinessObjectId__c>

```

```

        <Apttus_Config2__ChargeType__c>Subscription Fee</
Apttus_Config2__ChargeType__c>
        <Apttus_Config2__Description__c>SecureTraining Online</
Apttus_Config2__Description__c>
        <Apttus_Config2__EndDate__c>2020-06-19</Apttus_Config2__EndDate__c>
        <Apttus_Config2__Frequency__c>Yearly</Apttus_Config2__Frequency__c>
        <Apttus_Config2__IsPrimaryLine__c>true</
Apttus_Config2__IsPrimaryLine__c>
        <Apttus_Config2__IsPrimaryRampLine__c>>false</
Apttus_Config2__IsPrimaryRampLine__c>
        <Apttus_Config2__IsReadOnly__c>>false</Apttus_Config2__IsReadOnly__c>
        <Apttus_Config2__IsUsageTierModifiable__c>>false</
Apttus_Config2__IsUsageTierModifiable__c>
        <Apttus_Config2__ItemSequence__c>1</Apttus_Config2__ItemSequence__c>
        <Apttus_Config2__LineNumber__c>3</Apttus_Config2__LineNumber__c>
        <Apttus_Config2__LineType__c>Product/Service</
Apttus_Config2__LineType__c>
        <Apttus_Config2__MustUpgrade__c>>false</Apttus_Config2__MustUpgrade__c>
        <Apttus_Config2__NetPrice__c>5000.00000</Apttus_Config2__NetPrice__c>
        <Apttus_Config2__PriceType__c>Recurring</Apttus_Config2__PriceType__c>
        <Apttus_Config2__ProductId__c>01t4P00000884t9QAA</
Apttus_Config2__ProductId__c>
        <Apttus_Config2__ProductId__r xsi:type="Product2">
            <Id>01t4P00000884t9QAA</Id>
            <Apttus_Config2__Customizable__c>>false</
Apttus_Config2__Customizable__c>
            <Apttus_Config2__HasAttributes__c>>false</
Apttus_Config2__HasAttributes__c>
            <Apttus_Config2__HasDefaults__c>>false</
Apttus_Config2__HasDefaults__c>
            <Apttus_Config2__HasOptions__c>>false</Apttus_Config2__HasOptions__c>
            <Family>CSD Products</Family>
        </Apttus_Config2__ProductId__r>
        <Apttus_Config2__Quantity__c>10.00000</Apttus_Config2__Quantity__c>
        <Apttus_Config2__SellingFrequency__c>Yearly</
Apttus_Config2__SellingFrequency__c>
        <Apttus_Config2__SellingTerm__c>1.00000</
Apttus_Config2__SellingTerm__c>
        <Apttus_Config2__StartDate__c>2019-06-20</Apttus_Config2__StartDate__c>
        <Name>SecureTraining Online</Name>
    </AssetSearchResultDO:AssetItems>
    <AssetSearchResultDO:HasAssetItems>>true</AssetSearchResultDO:HasAssetItem
s>
</result>

```

```

        </getAssetsForSearchTextResponse>
    </soapenv:Body>
</soapenv:Envelope>
    
```

## Setting Incentives for Cart

You can use this API to set the incentives for a cart.

API	Signature
setIncentivesForCart	<pre> webService static Apttus_CPQApi.CPQ.SetIncentivesForCartResponseDO setIncentivesForCart(Apttus_CPQApi.CPQ.SetIncentivesForC artRequestDO request)                     </pre>

Request Parameter		
Name	Type	Description
request	Apttus_CPQApi.CPQ.SetIncentivesForCartRequestDO	The Set Incentive request data object.

Request Data Object - Apttus_CPQApi.CPQ.SetIncentivesForCartRequestDO		
Name	Type	Description
CartId	Id	The ID of the cart for which you want to set the incentives.
IncentiveCodes	List	The list incentives code that you want to set for cart or line item.
LineItemId	Id	The ID of the line item for which you want to set the incentive.

Data Object - Apttus_CPQApi.CPQApi.CPQ.SetIncentivesForCartResponseDO		
Name	Type	Description
IsSuccess	Boolean	Indicates whether the incentives were successfully set on the cart or line item.

### Code Sample

The sample code below enables you to set incentives for a cart or a line item.

```

1 // set incentives for cart
2 Apttus_CPQApi.CPQ.SetIncentivesForCartRequestDO request = new
   Apttus_CPQApi.CPQ.SetIncentivesForCartRequestDO();
3 request.CartId = configS0.Id;
4 request.IncentiveCodes.add(incentiveS0.Apttus_Config2__IncentiveCode__c);
5
6 Apttus_CPQApi.CPQ.SetIncentivesForCartResponseDO result =
   Apttus_CPQApi.CPQWebService.setIncentivesForCart(request);
7
8 // set incentives for cart line item
9 Apttus_CPQApi.CPQ.SetIncentivesForCartRequestDO request4 = new
   Apttus_CPQApi.CPQ.SetIncentivesForCartRequestDO();
10 request4.LineItemId = lineItemS0.Id;
11 request4.IncentiveCodes.add(incentiveS0.Apttus_Config2__IncentiveCode__c);
12
13 Apttus_CPQApi.CPQ.SetIncentivesForCartResponseDO result4 =
   Apttus_CPQApi.CPQWebService.setIncentivesForCart(request4);

```

## Retrieving Incentives Applied on the Cart

You can use this API to fetch the incentives that are applied on a cart.

API	Signature
<b>getAppliedIncentiveCodesForCart</b>	<i>webService static</i> <i>Apttus_CPQApi.CPQ.GetAppliedIncentiveCodesForCartResponseDO</i> <i>getAppliedIncentiveCodesForCart(Apttus_CPQApi.CPQ.GetAppliedIncentiveCodesForCartRequestDO request)</i>

Request Parameter		
Name	Type	Description
request	Apttus_CPQApi.CPQ.GetAppliedIncentiveCodesForCartRequestDO	The incentive code request data object

Request Data Object - Apttus_CPQApi.CPQ.GetAppliedIncentiveCodesForCartRequestDO		
Name	Type	Description
CartId	Id	The ID of the cart from which you want to fetch the applied promotion

Data Object - Apttus_CPQApi.CPQ.SetIncentivesForCartResponseDO		
Name	Type	Description
IncentiveCodes	List <IncentiveCodes>	The list of the incentives codes that are applied on the cart

### Code Sample

The sample code below enables you to fetch the incentives applied on the cart.

```

1  Apttus_CPQApi.CPQ.GetAppliedIncentiveCodesForCartRequestDO request3 = new
    Apttus_CPQApi.CPQ.GetAppliedIncentiveCodesForCartRequestDO();
2  request3.CartId = configS0.Id;
3
4  Apttus_CPQApi.CPQ.GetAppliedIncentiveCodesForCartResponseDO result3 =
    Apttus_CPQApi.CPQWebService.getAppliedIncentiveCodesForCart(request3);
    
```



## Abandoning a Cart

This deletes the selected cart in *Draft* status.

API	Signature
<b>abandonCart</b>	<i>webservice static</i> <i>Apttus_CPQApi.CPQ.AbandonCartResponseDO</i> <i>abandonCart(Apttus_CPQApi.CPQ.AbandonCartRequestDO</i> <i>request)</i>

Parameters		
Name	Type	Description
request	Apttus_CPQApi.CPQ.AbandonCartRequestDO	The request data object.

Request Data Object - Apttus_CPQApi.CPQ.AbandonCartRequestDO		
Field	Type	Description
CartId	ID	The Id of the cart.

Response Data Object - Apttus_CPQApi.CPQ.AbandonCartResponseDO		
Field	Type	Description
IsSuccess	Boolean	Indicates whether the cart is deleted successfully.

## Code Sample

The sample below enables you to send an API request for abandon cart along with the cart ID. The cart whose id has been passed will be abandoned when the request is successful. The response message in the sample indicates whether the request was successful. When the user clicks Abandon Cart the *abandonCart* API will be invoked and the cartID associated with the request is abandoned.

```

1  public void abandonCart()
2  {
3      Apttus_CPQApi.CPQ.AbandonCartRequestDO request = new
4      Apttus_CPQApi.CPQ.AbandonCartRequestDO();
5      request.CartId = cartId;
6
7      Apttus_CPQApi.CPQ.AbandonCartResponseDO response =
8      Apttus_CPQApi.CPQWebService.abandonCart(request);
9      ApexPages.addMessage(new ApexPages.Message(ApexPages.severity.info,
10     'Success: ' + response.IsSuccess));
11 }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## API Prerequisites

- [Creating a Cart from a Quote](#)

## Request/Response XML

### Example Request

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:cpq="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"

```

```

xmlns:cpq1="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ">
<soapenv:Header>
  <cpq:SessionHeader>
    <cpq:sessionId>00DZ000000NAEIA!
ASAAQN7k3xhhvMe.j.8gpR.ijcGq77HJeF4MEDfbxbBAqZ8r4WWNTv30Vb6o1bjtHJLbq5mvHcuAH_ie6sTC3
DzrWgnLdUuD</cpq:sessionId>
  </cpq:SessionHeader>
</soapenv:Header>
<soapenv:Body>
  <cpq:abandonCart>
    <cpq:request>
      <cpq1:CartId>a10Z0000002YqyaMAC</cpq1:CartId>
    </cpq:request>
  </cpq:abandonCart>
</soapenv:Body>
</soapenv:Envelope>

```

### Example Response

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:AbandonCartResponseD0="http://soap.sforce.com/schemas/class/Apttus_CPQApi/
CPQ">
  <soapenv:Body>
    <abandonCartResponse>
      <result>
        <AbandonCartResponseD0:IsSuccess>>true</AbandonCartResponseD0:IsSuccess>
      </result>
    </abandonCartResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Finalizing a Cart

This finalizes the cart, synchronizing the cart line items with the quote/proposal.

API	Signature
<b>finalizeCart</b>	<i>webservice static Apttus_CPQApi.CPQ.FinalizeCartResponseDO finalizeCart(Apttus_CPQApi.CPQ.FinalizeCartRequestDO request)</i>

Parameters		
Name	Type	Description
request	Apttus_CPQApi.CPQ.FinalizeCartRequestDO	The request data object.

Request Data Object - Apttus_CPQApi.CPQ.FinalizeCartRequestDO		
Field	Type	Description
CartId	ID	The Id of the cart.

Response Data Object - Apttus_CPQApi.CPQ.FinalizeCartResponseDO		
Field	Type	Description
IsSuccess	Boolean	Indicates whether finalizing the cart items was successful.

### Code Sample

The sample enables you to finalize a cart with a specific cartID and synchronizes the products (line items) added to the cart with the quote or proposal used to generate the

cart. You can create a button finalize cart which the user will click once the products in the cart are adjusted and final. Invoke this API on click of the button.

```

1  public void finalizeCart()
2      {
3          // create the finalize cart request
4          Apttus_CpqApi.CPQ.FinalizeCartRequestDO request = new
Apttus_CpqApi.CPQ.FinalizeCartRequestDO();
5
6          // add request parameters
7          request.CartId = CartId;
8
9          // finalize the cart
10         Apttus_CpqApi.CPQ.FinalizeCartResponseDO response =
Apttus_CpqApi.CPQWebService.finalizeCart(request);
11     }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## API Prerequisites

- [Creating a Cart from a Quote](#)
- [Adding Products to a Cart OR Adding a Bundle to a Cart](#)
- [Updating Price For A Cart](#)

## Response/Request XML

### Example Request

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:cpq="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:cpq1="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ">
  <soapenv:Header>
    <cpq:SessionHeader>

```

```

    <cpq:sessionId>00DZ000000NAEIA!
ASAAQN7k3xhhvMe.j.8gpR.ijcGq77HJeF4MEDfbxbBAqZ8r4WWNTv30Vb6o1bjtHJLbq5mvHcuAH_ie6sTC3
DzrWgnLdUuD</cpq:sessionId>
  </cpq:SessionHeader>
</soapenv:Header>
<soapenv:Body>
  <cpq:finalizeCart>
    <cpq:request>
      <cpq1:CartId>a10Z0000002YqyaMAC</cpq1:CartId>
    </cpq:request>
  </cpq:finalizeCart>
</soapenv:Body>
</soapenv:Envelope>

```

### Example Response

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:FinalizeCartResponseD0="http://soap.sforce.com/schemas/class/Apttus_CPQApi/
CPQ">
  <soapenv:Body>
    <finalizeCartResponse>
      <result>
        <FinalizeCartResponseD0:IsSuccess>>true</FinalizeCartResponseD0:IsSucc
ess>
      </result>
    </finalizeCartResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Synchronizing a Cart

This is used to sync the shopping cart with the product configuration on the quote record.

API	Signature
synchronizeCart	<i>webservice static Apttus_CPQApi.CPQ.SynchronizeCartResponseDO synchronizeCart(Apttus_CPQApi.CPQ.SynchronizeCartRequest DO request)</i>

Parameters		
Name	Type	Description
request	Apttus_CPQApi.CPQ.SynchronizeCartRequestDO	The request data object.

Request Data Object - Apttus_CPQApi.CPQ.SynchronizeCartRequestDO		
Name	Type	Description
CartId	ID	The ID of the cart (Product Configuration) you want to synchronize.

Response Data Object - Apttus_CPQApi,CPQ.SynchronizeCartResponseDO		
Name	Type	Description
IsSuccess	Boolean	Indicates whether synchronizing the cart items was successful.

### Code Sample

The sample below enables you to synchronize the cart items with a quote. When the customer has finalized the cart, you can synchronize the products in the cart with the quote

used to generate the cart. Invoke this API when the user has finalized the cart. The cartID with the updated products is provided as a parameter to the API.

```

1  public void synchronizeCart()
2  {
3      //Pass the cartID as a parameter to the API to synchronize your quote
      //with the updated cart items.
4      Apttus_CPQApi.CPQ.SynchronizeCartRequestDO request = new
      Apttus_CPQApi.CPQ.SynchronizeCartRequestDO();
5      request.CartId = cartId;
6
7      Apttus_CPQApi.CPQ.SynchronizeCartResponseDO response =
      Apttus_CPQApi.CPQWebService.synchronizeCart(request);
8      ApexPages.addMessage(new ApexPages.Message(ApexPages.severity.info,
      'Success: ' + response.IsSuccess));
9  }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## API Prerequisites

- [Creating a Cart from a Quote](#)
- [Finalizing a Cart](#)

## Response/Request XML

### Example Request

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:cpq="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:cpq1="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ">
  <soapenv:Header>
    <cpq:SessionHeader>

```



```

    <cpq:sessionId>00D4P00000z7dE!
AQQQAHIR1W9.1as89Auz.CvNYxqyg56gLLWgUtP5VZxidvTsb1DpQZpmyDuqZOiF4VctBp3jhhJIxG9oRQ4A
4F9h98N0inT</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <cpq:synchronizeCart>
      <cpq:request>
        <cpq1:CartId>a1l4P00000Bg7D9</cpq1:CartId>
      </cpq:request>
    </cpq:synchronizeCart>
  </soapenv:Body>
</soapenv:Envelope>

```

### Example Response

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="ht
tp://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
xmlns:SynchronizeCartResponseD0="http://soap.sforce.com/schemas/class/Apttus_CPQApi/
CPQ">
  <soapenv:Body>
    <synchronizeCartResponse>
      <result>
        <SynchronizeCartResponseD0:IsSuccess>>true</SynchronizeCartResponseD0:IsSu
ccess>
      </result>
    </synchronizeCartResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Updating Quote Terms

This method enables you to update the expected start date and expected end date of the quote, along with its selling term. This method does the following:

- Updates *Expected Start Date* and *Expected End Date* for the Quote/Proposal object.
- Retrieves the finalized cart associated with the quote.
- Creates a new cart by checking out the finalized cart.
- Updates Expected Start Date & Expected End Date in the cart object. Note this is for the new *Product Configuration* cart object.
- Updates Start Date, End Date, and Selling Term on line item terms that have a valid start date, i.e. when the start date is not null.

- Selling Term may be set to null to clear out the existing selling term which triggers a recalculation of the selling term using the start and end dates. Alternatively, if start date and selling term are provided, the end date is automatically calculated.
- Updates the net price and total price for the cart.
- The ID of the new cart is returned to the caller.
- It is the caller's responsibility to finalize the cart.

The field *QuoteLineItemColl*, which is part of the *Apttus\_CPQApi.CPQ.UpdateQuoteTermRequestDO*, can accept a list of quote line item sObjects which can hold the data to override one or more line items. For this the line items need to have the following fields populated:

- Start Date
- End Date
- Selling Term
- Derived Form

Line items that match the *Derived From* value gets overridden values from the Quote Line items. The remaining fields will default to values provided in the request object. A null value in the *Start Date* field in the request will prevent the values from defaulting to the line items from the request object.

API	Signature
updateQuoteTerm	<i>WebService static</i> <i>Apttus_CPQApi.CPQ.UpdateQuoteTermResponseDO</i> <i>updateQuoteTerm(Apttus_CPQApi.CPQ.UpdateQuoteTermRequestDO request)</i>

Parameters		
Name	Type	Description
request	Apttus_CPQApi.CPQ.UpdateQuoteTermRequestDO	This is the request call made by the method.

Request Data Object - Apttus_CPQApi.CPQ.UpdateQuoteTermRequestDO			
Field	Type	Required	Description
EndDate	Date	No	The expected end of the quote.
LineItemColl	Apttus_CPQApi.CPQ.LineItemCollDO	No	The list of line items
QuoteId	ID	Yes	The Id of the quote you want to change the dates and selling term for.
QuoteLineItemCollDO	Apttus_CPQApi.CPQ.QuoteLineItemCollDO	No	The collection of Line items to be updated.
ReCalcSellingTerm	Boolean	No	You can use this method enabled recalculating the selling term
SellingTerm	Decimal	No	The selling term of the quote is typically measured in years, months, or days.
StartDate	Date	Yes	The expected start date of the quote.

Data Object - Apttus_CPQApi.CPQ.LineItemCollDO		
Field	Type	Description
LineItems	List	The list of line items

Data Object - Apttus_CPQApi.CPQ.QuoteLineItemCollDO		
Field	Type	Description
LineItems	List	The list of line items

Response Data Object - Apttus_CPQApi.CPQ.UpdateQuoteTermResponseDO		
Field	Type	Description
CartId	ID	Id of the cart updated.

### Code Sample

The sample below enables you update the end date of a cart after the date of acceptance of the proposal. The auto-roll up end date is the sum of the proposal acceptance start date and selling term. When the recipient of the proposal accepts the quote and updates the status, invoke this API to update the start date and selling term of the products associated to the quote. In the sample below, the user enters the quote name in a search field and clicks Search. If a valid quote exists by the name, the quote is fetched. If the user clicks the update Quote Terms button, the API is invoked and the end date is updated.

```

1  public void autoroll()
2  {
3      //Search a quote by its name and fetch its ID
4      List<Apttus_Proposal__Proposal__c> listQuote = [Select Id from
Apttus_Proposal__Proposal__c where Name=:quoteNumber LIMIT 1];
5
6      //If a valid quote exists with the name execute the if loop
7      if(listQuote.size() > 0)
8      {
9          Apttus_CPQApi.CPQ.UpdateQuoteTermRequestDO updateQTRRequest = new
Apttus_CPQApi.CPQ.UpdateQuoteTermRequestDO();
10         updateQTRRequest.QuoteId = listQuote[0].Id;
11
12         //Start date=current date entered in the from field of the
proposal.
13         updateQTRRequest.StartDate = fromdate;
14
15         //End date=startdate+selling term
16         updateQTRRequest.EndDate =
updateQTRRequest.StartDate.addMonths(sellingTerm);
17         /**updateQTRRequest.EndDate = endDate;**/
18
19         /**updateQTRRequest.SellingTerm = 12;*/
20

```

```

21         //Execute the API to update quote term
22         Apttus_CPQApi.CPQ.UpdateQuoteTermResponseDO updateQTReponse =
Apttus_CPQApi.CPQWebService.updateQuoteTerm(updateQTRequest);
23
24         ID cartID = updateQTReponse.cartID;
25
26         //Invoke the finalize cart request to update end date with the
cart
27         Apttus_CPQApi.CPQ.FinalizeCartRequestDO finalizeRequest = new
Apttus_CPQApi.CPQ.FinalizeCartRequestDO();
28         finalizeRequest.CartId = cartId;
29
30         //Invoke the finalize cart API to update the cart details
31         Apttus_CPQApi.CPQ.FinalizeCartResponseDO finalizeResponse =
Apttus_CPQApi.CPQWebService.finalizeCart(finalizeRequest);
32     }
33 }

```

The sample below enables you to update the start date and end date of a cart after the date of acceptance of the proposal. When the recipient of the proposal accepts the quote and updates the status, invoke this API to update the start date, end date and selling term of the products associated to the quote. In the sample below, the user enters the quote name in a search field and clicks Search. If a valid quote exists by the name, the quote is fetched. If the user clicks the update Quote Terms button, the API is invoked and the start date is updated. The validity i.e the selling term has to be updated on a pro-rata basis.

```

1     public void prorata()
2     {
3         //Search a quote by its name and fetch its ID
4         List<Apttus_Proposal__Proposal__c> listQuote = [Select Id from
Apttus_Proposal__Proposal__c where Name=:quoteNumber LIMIT 1];
5
6
7         //If a valid quote exists with the name execute the if loop
8         if(listQuote.size() > 0)
9         {
10
11             Apttus_Proposal__Proposal__c propSO = [select id,
Apttus_Proposal__ExpectedStartDate__c, Apttus_Proposal__ExpectedEndDate__c
from Apttus_Proposal__Proposal__c      where id = :listQuote[0].Id limit 1]
;
12             Date startDate = propSO.Apttus_Proposal__ExpectedStartDate__c;

```

```

13         Date endDate = propSO.Apttus_Proposal__ExpectedEndDate__c;
14
15         Apttus_CPQApi.CPQ.UpdateQuoteTermRequestDO updateQTRequest = new
Apttus_CPQApi.CPQ.UpdateQuoteTermRequestDO();
16         updateQTRequest.QuoteId = listQuote[0].Id;
17         //Fetch the start date of the proposal
18         updateQTRequest.StartDate = fromDate;
19
20         /**updateQTRequest.EndDate =
updateQTRequest.StartDate.addMonths(12);**/
21         //Fetch the end date of the proposal
22         updateQTRequest.EndDate = endDate ;
23         updateQTRequest.SellingTerm = null;
24
25         //Selling Term to be recalculated based on the start date and end
date
26         updateQTRequest.ReCalcSellingTerm = true;
27         /*updateQTRequest.SellingTerm = 12;*/
28
29         //Execute the API to update quote term
30         Apttus_CPQApi.CPQ.UpdateQuoteTermResponseDO updateQTReponse =
Apttus_CPQApi.CPQWebService.updateQuoteTerm(updateQTRequest);
31         ID cartID = updateQTReponse.cartID;
32
33         //Invoke the finalize cart request to update selling term of
product with the cart
34         Apttus_CPQApi.CPQ.FinalizeCartRequestDO finalizeRequest = new
Apttus_CPQApi.CPQ.FinalizeCartRequestDO();
35         finalizeRequest.CartId = cartId;
36         Apttus_CPQApi.CPQ.FinalizeCartResponseDO finalizeResponse =
Apttus_CPQApi.CPQWebService.finalizeCart(finalizeRequest);
37     }
38 }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## API Prerequisites

- [Finalizing a Cart](#)

## Response/Request XML

### Example Request

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:cpq="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:cpq1="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQ">
  <soapenv:Header>
    <cpq:SessionHeader>
      <cpq:sessionId>00DZ000000NAEIA!
      ASAAQLC3r1a0VIL_gk7v0A9K1n7dLEcCAANQHwNfXAPFKZEH292fAdeal5Nps8X2k1Nu98fcp6usnVFmzKARw
      E99xq_dE8U6</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <cpq:updateQuoteTerm>
      <cpq:request>
        <cpq1:EndDate>2020-12-30</cpq1:EndDate>
        <cpq1:QuoteId>a0eZ0000005pG5NIAU</cpq1:QuoteId>
        <cpq1:QuoteLineItemColl>
          <cpq1:LineItems>
            <cpq:Apttus_QPConfig__EndDate__c>2020-12-30</cpq:Apttus_QPCon
            fig__EndDate__c>
            <cpq:Apttus_QPConfig__SellingTerm__c>1</cpq:Apttus_QPConfig__
            SellingTerm__c>
            <cpq:Apttus_QPConfig__StartDate__c>2020-11-01</cpq:Apttus_QPC
            onfig__StartDate__c>
            <cpq:Apttus_QPConfig__DerivedFromId__c>a19Z0000006sPdw</cpq:A
            pttus_QPConfig__DerivedFromId__c>
          </cpq1:LineItems>
        </cpq1:QuoteLineItemColl>
        <cpq1:ReCalcSellingTerm>>false</cpq1:ReCalcSellingTerm>
        <cpq1:SellingTerm>1</cpq1:SellingTerm>
        <cpq1:StartDate>2020-11-01</cpq1:StartDate>
      </cpq:request>
    </cpq:updateQuoteTerm>
  </soapenv:Body>
</soapenv:Envelope>

```

```

</soapenv:Body>
</soapenv:Envelope>

```

### Example Response

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService"
  xmlns:UpdateQuoteTermResponseD0="http://soap.sforce.com/schemas/class/
Apttus_CPQApi/CPQ">
  <soapenv:Body>
    <updateQuoteTermResponse>
      <result>
        <UpdateQuoteTermResponseD0:CartId>a10Z00000002hsIXMAY</
UpdateQuoteTermResponseD0:CartId>
      </result>
    </updateQuoteTermResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

## Modifying Assets (Deprecated)

This method enables you to complete standard asset based ordering actions of renew, terminate, amend, upgrade and increment.

This method creates the cart line item and associates it to the corresponding asset id. Actions can be performed on a single line item or a selected set of asset line items. Actions can also be performed on standalone product or bundle product.

When doing a renewal, you can use any of the standard adjustment types, including % *Uplift* which is used to pass an uplift percentage that is applied to the *base price* of the renewed line item, resulting in an updated *net renewal* price.

### Note

*CartId*, *AssetLineItemId\_c*, and *AssetActions* are required parameters, the use of other parameters depends on your business case.



API	Signature
modifyAsset	<i>webservice static Apttus_CPQApi.CPQ.ModifyAssetResponseDO modifyAsset(Apttus_CPQApi.CPQ.ModifyAssetRequestDO request)</i>

Parameters		
Name	Type	Description
request	Apttus_CPQApi.CPQ.ModifyAssetRequestDO	This is the request call made by the method.

Request Data Object - Apttus_CPQApi.CPQ.ModifyAssetRequestDO		
Field	Type	Description
AssetActions	List<Apttus_CPQApi.CPQ.AssetActionDO>	The list of asset actions, which is either <i>Renew, Amend, Increment, Cancel, Upgrade</i> .
CartId	ID	The Id of cart.

Response Data Object - Apttus_CPQApi.CPQ.ModifyAssetResponseDO		
Field	Type	Description
AssetActions	List<Apttus_CPQApi.CPQ.AssetActionDO>	The list of asset actions, which is either <i>Renew, Amend, Increment, Cancel, Upgrade</i> .
LineNumber	List<Decimal>	The asset line number.

Data Object - Apttus_CPQApi.CPQ.AssetActionDO		
Field	Type	Description
AdjustmentAmount	Decimal	Enter the value (number) for the adjustment. How the value is interpreted is based on the Adjustment Type. For example, 10 with an Adjustment Type of <i>% Uplift</i> means the price will be cut in half, whereas that same value of 10 with an Adjustment Type of <i>Price Override</i> means the price will become 10.
AdjustmentType	String	This is the type of adjustment that will be used. You can select from types such as % Discount, Discount Amount, and Price Override. Your specific implementation may have more or less types.
AssetAction	String	This is the specific asset action for the asset line item, referenced by AssetLineItemId.
AssetLineItemId	ID	This is the ID for the asset line item that you are going to renew, amend, increment, or cancel. This is mandatory.
Comments	String	Text string containing comments.
CustomData	Apttus_Config2__LineItem__c	This can be used to include custom fields in line item object.
CustomFields	List	This can be used to include custom fields you have added to the asset object.
EndDate	Date	This is the end date for the asset action. If you are terminating an asset you only need an end date.
Message	String	Text string containing comments.
Pending	Boolean	This indicates whether an asset action is pending.

Data Object - Apttus_CPQApi.CPQ.AssetActionDO		
Field	Type	Description
Quantity	Decimal	The amount of assets.
SellingTerm	Decimal	This is the selling term of the asset line item that you want to renew, amend, or increment.
StartDate	Date	This is the start date for the asset action.

### Code Sample

You can amend a list of standalone or bundle products. You can change the dates, quantity, or some options for a product using the Amend API. You can also cancel or change options in an Option Group using Amend.

The Sample below enables you to change amend the start and end date for selected assets of an account. The response returns the lineitem number of the asset to be updated. The assetAction status, the status of the asset is modified and updated. When the customer clicks Amend, you can invoke the modifyAssets API and choose to amend the date, quantity.

```

1  public void Amend()
2  {
3  //Create an assetAction variable with value as amend
4      string assetAction = 'Amend';
5
6      List<Apttus_CPQApi.CPQ.AssetActionDO> assetActDOList = new
7      List<Apttus_CPQApi.CPQ.AssetActionDO>();
8
9  //If Asset is selected in the cart page, execute the loop
10 for (AssetWrapperClass objAssetWrapper : lstAssetWrapper)
11     {
12         if (objAssetWrapper.selected)
13         {
14             Apttus_CPQApi.CPQ.AssetActionDO assetActDO = new
15             Apttus_CPQApi.CPQ.AssetActionDO();
16             assetActDO.AssetAction = assetAction;
17             assetActDO.AssetLineItemId = objAssetWrapper.AssetId ;
18         }
19     }
20 //Change the end date to five months from start date

```

```

17         assetActD0.EndDate = Date.today().addMonths(5);
18 //assetAction pending is true
19         assetActD0.Pending = true;
20         assetActD0.Quantity = objAssetWrapper.Quantity;
21 //Amend the start date to include today's date
22         assetActD0.StartDate = Date.today().addMonths(0);
23
24         List<String> customFields = new List<String>();
25         customFields.add('Apttus_Config2__Comments__c');
26         customFields.add('Apttus_Config2__PricingStatus__c');
27
28         assetActD0.CustomFields = customFields;
29
30         Apttus_Config2__LineItem__c liS0 = new
Apttus_Config2__LineItem__c();
31         liS0.Apttus_Config2__Comments__c = 'Comments Added my Modify
Asset API Code';
32         liS0.Apttus_Config2__PricingStatus__c = 'Complete';
33         assetActD0.CustomData = liS0;
34
35         assetActD0List.add(assetActD0);
36     }
37 }
38
39 //Execute the API for a specific cart using CartID
40     Apttus_CPQApi.CPQ.ModifyAssetRequestD0 modifyRequest = new
Apttus_CPQApi.CPQ.ModifyAssetRequestD0();
41     modifyRequest.CartId = cartID;
42     modifyRequest.AssetActions = assetActD0List;
43
44     Apttus_CPQApi.CPQ.ModifyAssetResponseD0 modifyReponse =
Apttus_CPQApi.CPQWebservice.modifyAsset(modifyRequest);
45
46     List<Decimal> lineNumberList = modifyReponse.LineNumbers;
47
48     List<Apttus_CPQApi.CPQ.AssetActionD0> assetList =
modifyReponse.AssetActions;
49
50 //Signifies the status of the assetAction
51     Boolean bIsPending = false;
52
53     Integer iCounter = 0;
54

```

```

55     while (iCounter < 1000)
56     {
57         bIsPending = false;
58         for(Apttus_CPQApi.CPQ.AssetActionDO objAsset: assetList)
59         {
60             if(objAsset.Pending == true)
61             {
62                 bIsPending = true;
63                 break;
64             }
65         }
66         //If the asset action is pending execute the loop below to change the
        //status
67         if(bIsPending == true)
68         {
69             modifyReponse =
Apttus_CPQApi.CPQWebservice.modifyAsset(modifyRequest);
70             lineNumberList = modifyReponse.LineNumbers;
71             assetList = modifyReponse.AssetActions;
72         }
73         else
74         {
75             break;
76         }
77         iCounter++;
78     }
79
80     bindAssets();
81 }
82 //This function enables you to bind updated assets to an account
//dynamically.
83 public void bindAssets()
84 {
85     List<Apttus_Config2__AssetLineItem__c> aliSOList = [select id,
Apttus_Config2__Description__c,
86     Apttus_Config2__Quantity__c, Apttus_Config2__StartDate__c,
Apttus_Config2__EndDate__c from Apttus_Config2__AssetLineItem__c
87     where Apttus_Config2__AccountID__c = :accountID];
88
89     lstAssetWrapper = new List<AssetWrapperClass>();
90
91     for(Apttus_Config2__AssetLineItem__c aliSO : aliSOList)
92     {
93         AssetWrapperClass objAssetWrapperClass = new AssetWrapperClass ();
94         objAssetWrapperClass.AssetId = aliSO.ID;

```

```

95         objAssetWrapperClass.ProductId =
aliSO.Apttus_Config2__Description__c;
96         objAssetWrapperClass.Quantity = aliSO.Apttus_Config2__Quantity__c;
97         objAssetWrapperClass.StartDate =
aliSO.Apttus_Config2__StartDate__c;
98         objAssetWrapperClass.EndDate = aliSO.Apttus_Config2__EndDate__c;
99
100        lstAssetWrapper.Add(objAssetWrapperClass);
101    }
102 }

```

You can renew a list of standalone or bundled products. You can choose to renew recurring items, overall bundle, and recurring options. You can also cancel or change the quantity, price, or an entire option. The sample below enables you to renew the assets. When the user selects an asset and clicks renew, invoke this API and use the sample below to extend the end date of the assets by 12 months.

```

1  public void ReNew()
2  {
3  //Declare assetAction variable with value as Renew
4      string assetAction = 'Renew';
5      List<Apttus_CPQApi.CPQ.AssetActionDO> assetActDOList = new
List<Apttus_CPQApi.CPQ.AssetActionDO>();
6
7      //For a selected asset you can execute the loop
8      for(AssetWrapperClass objAssetWrapper : lstAssetWrapper)
9      {
10         if(objAssetWrapper.selected)
11         {
12             Apttus_CPQApi.CPQ.AssetActionDO assetActDO = new
Apttus_CPQApi.CPQ.AssetActionDO();
13             assetActDO.AssetAction = assetAction;
14             assetActDO.AssetLineItemId = objAssetWrapper.AssetId ;
15
16             //Enables you to set end date as 12 months from the existing
end date
17             assetActDO.EndDate = objAssetWrapper.EndDate.addMonths(12);
18             assetActDO.Pending = true;
19             assetActDO.Quantity = objAssetWrapper.Quantity;

```

```

20
21     //Increment start date by One day from the existing start date
22     assetActD0.StartDate = objAssetWrapper.EndDate.addDays(1);
23
24     List<String> customFields = new List<String>();
25     customFields.add('Apttus_Config2__Comments__c');
26     customFields.add('Apttus_Config2__PricingStatus__c');
27     assetActD0.CustomFields = customFields;
28
29     //Fetch asset line item number
30     Apttus_Config2__LineItem__c liS0 = new
Apttus_Config2__LineItem__c();
31     liS0.Apttus_Config2__Comments__c = 'Comments Added my Modify
Asset API Code';
32     liS0.Apttus_Config2__PricingStatus__c = 'Complete';
33     assetActD0.CustomData = liS0;
34     assetActD0List.add(assetActD0);
35     }
36     }
37     //Invoke the modify assets API and specify the cart for which the API
is invoked.
38     Apttus_CPQApi.CPQ.ModifyAssetRequestD0 modifyRequest = new
Apttus_CPQApi.CPQ.ModifyAssetRequestD0();
39     modifyRequest.CartId = cartID;
40     modifyRequest.AssetActions = assetActD0List;
41
42     Apttus_CPQApi.CPQ.ModifyAssetResponseD0 modifyReponse =
Apttus_CPQApi.CPQWebservice.modifyAsset(modifyRequest);
43     List<Decimal> lineNumberList = modifyReponse.LineNumbers;
44     List<Apttus_CPQApi.CPQ.AssetActionD0> assetList =
modifyReponse.AssetActions;
45     Boolean bIsPending = false;
46     Integer iCounter = 0;
47     while (iCounter < 1000)
48     {
49         bIsPending = false;
50         for(Apttus_CPQApi.CPQ.AssetActionD0 objAsset: assetList)
51         {
52             if(objAsset.Pending == true)
53             {
54                 bIsPending = true;
55                 break;
56             }
57         }
58     }

```

```

59         //If asset Action is pending execute the following loop
60         if(bIsPending == true)
61         {
62             modifyReponse =
Apttus_CPQApi.CPQWebservice.modifyAsset(modifyRequest);
63             lineNumberList = modifyReponse.LineNumbers;
64             assetList = modifyReponse.AssetActions;
65         }
66         else
67         {
68             break;
69         }
70         iCounter++;
71     }
72     bindAssets();
73 }
74
75 //This function enables you to bind updated asset.
76 public void bindAssets()
77 {
78     List<Apttus_Config2__AssetLineItem__c> aliSOList = [select id,
Apttus_Config2__Description__c,
79     Apttus_Config2__Quantity__c, Apttus_Config2__StartDate__c,
Apttus_Config2__EndDate__c from Apttus_Config2__AssetLineItem__c
80     where Apttus_Config2__AccountID__c = :accountID];
81
82     lstAssetWrapper = new List<AssetWrapperClass>();
83
84     for(Apttus_Config2__AssetLineItem__c aliS0 : aliSOList)
85     {
86         AssetWrapperClass objAssetWrapperClass = new AssetWrapperClass ();
87         objAssetWrapperClass.AssetId = aliS0.ID;
88         objAssetWrapperClass.ProductId =
aliS0.Apttus_Config2__Description__c;
89         objAssetWrapperClass.Quantity = aliS0.Apttus_Config2__Quantity__c;
90         objAssetWrapperClass.StartDate =
aliS0.Apttus_Config2__StartDate__c;
91         objAssetWrapperClass.EndDate = aliS0.Apttus_Config2__EndDate__c;
92
93         lstAssetWrapper.Add(objAssetWrapperClass);
94     }
95 }

```



You can increment the quantity or the dates of the asset items using the increment asset action. For example, the user selects an asset item and clicks Increment. You can use the sample below to update the quantity and the end date by five months and bind the updated assets to the account.

```

1  public void Increment()
2  {
3      //Declare the assetAction variable with value as Increment
4      string assetAction = 'Increment';
5
6      List<Apttus_CPQApi.CPQ.AssetActionDO> assetActDOList = new
7      List<Apttus_CPQApi.CPQ.AssetActionDO>();
8
9      for(AssetWrapperClass objAssetWrapper : lstAssetWrapper)
10     {
11         if(objAssetWrapper.selected)
12         {
13             //For a selected asset increment the quantity by 5 and
14             //increment the end date by 5.
15             Apttus_CPQApi.CPQ.AssetActionDO assetActDO = new
16             Apttus_CPQApi.CPQ.AssetActionDO();
17             assetActDO.AssetAction = assetAction;
18             assetActDO.AssetLineItemId = objAssetWrapper.AssetId ;
19             assetActDO.EndDate = Date.today().addMonths(5);
20             assetActDO.Pending = true;
21             assetActDO.Quantity = objAssetWrapper.Quantity + 5;
22             assetActDO.StartDate = Date.today().addMonths(0);
23
24             List<String> customFields = new List<String>();
25             customFields.add('Apttus_Config2__Comments__c');
26             customFields.add('Apttus_Config2__PricingStatus__c');
27
28             assetActDO.CustomFields = customFields;
29             //Line number of the asset items
30             Apttus_Config2__LineItem__c liSO = new
31             Apttus_Config2__LineItem__c();
32             liSO.Apttus_Config2__Comments__c = 'Comments Added my Modify
33             Asset API Code';
34             liSO.Apttus_Config2__PricingStatus__c = 'Complete';
35             assetActDO.CustomData = liSO;

```

```

31         assetActDOList.add(assetActDO);
32     }
33 }
34 //Invoke the modifyAsset API to increment the quantity and end date
35 Apttus_CPQApi.CPQ.ModifyAssetRequestDO modifyRequest = new
Apttus_CPQApi.CPQ.ModifyAssetRequestDO();
36 modifyRequest.CartId = cartID;
37 modifyRequest.AssetActions = assetActDOList;
38
39 Apttus_CPQApi.CPQ.ModifyAssetResponseDO modifyReponse =
Apttus_CPQApi.CPQWebservice.modifyAsset(modifyRequest);
40 List<Decimal> lineNumberList = modifyReponse.LineNumbers;
41 List<Apttus_CPQApi.CPQ.AssetActionDO> assetList =
modifyReponse.AssetActions;
42 Boolean bIsPending = false;
43 Integer iCounter = 0;
44 while (iCounter < 1000)
45 {
46     bIsPending = false;
47     for(Apttus_CPQApi.CPQ.AssetActionDO objAsset: assetList)
48     {
49         if(objAsset.Pending == true)
50         {
51             bIsPending = true;
52             break;
53         }
54     }
55
56     //If asset action is pending, execute this loop
57     if(bIsPending == true)
58     {
59         modifyReponse =
Apttus_CPQApi.CPQWebservice.modifyAsset(modifyRequest);
60         lineNumberList = modifyReponse.LineNumbers;
61         assetList = modifyReponse.AssetActions;
62     }
63     else
64     {
65         break;
66     }
67     iCounter++;
68 }
69

```

```

70     bindAssets();
71 }
72
73 //Bind updated asset items
74 public void bindAssets()
75 {
76     List<Apttus_Config2__AssetLineItem__c> aliSOList = [select id,
Apttus_Config2__Description__c,
77     Apttus_Config2__Quantity__c, Apttus_Config2__StartDate__c,
Apttus_Config2__EndDate__c from Apttus_Config2__AssetLineItem__c
78     where Apttus_Config2__AccountID__c = :accountID];
79
80     lstAssetWrapper = new List<AssetWrapperClass>();
81
82     for(Apttus_Config2__AssetLineItem__c aliS0 : aliSOList)
83     {
84         AssetWrapperClass objAssetWrapperClass = new AssetWrapperClass ();
85         objAssetWrapperClass.AssetId = aliS0.ID;
86         objAssetWrapperClass.ProductId =
aliS0.Apttus_Config2__Description__c;
87         objAssetWrapperClass.Quantity = aliS0.Apttus_Config2__Quantity__c;
88         objAssetWrapperClass.StartDate =
aliS0.Apttus_Config2__StartDate__c;
89         objAssetWrapperClass.EndDate = aliS0.Apttus_Config2__EndDate__c;
90
91         lstAssetWrapper.Add(objAssetWrapperClass);
92     }
93 }

```

The terminate asset action enables you to terminate or cancel the selected asset items. The sample below enables you to terminate the asset and change its status by setting the end date as current date.

```

1 public void Terminate()
2 {
3     //Declaring variable assetAction and assigning it a value Cancel
4     string assetAction = 'Cancel';
5     List<Apttus_CPQApi.CPQ.AssetActionDO> assetActDOList = new
List<Apttus_CPQApi.CPQ.AssetActionDO>();
6
7     //To terminate a selected Asset, set the end date as today
8     for(AssetWrapperClass objAssetWrapper : lstAssetWrapper)

```

```

9      {
10         if(objAssetWrapper.selected)
11         {
12             Apttus_CPQApi.CPQ.AssetActionDO assetActDO = new
Apttus_CPQApi.CPQ.AssetActionDO();
13             assetActDO.AssetAction = assetAction;
14             assetActDO.AssetLineItemId = objAssetWrapper.AssetId ;
15             assetActDO.EndDate = Date.today().addMonths(0);
16             assetActDO.Pending = true;
17             assetActDO.Quantity = objAssetWrapper.Quantity;
18             //assetActDO.StartDate = Date.today().addMonths(0);
19
20             List<String> customFields = new List<String>();
21             customFields.add('Apttus_Config2__Comments__c');
22             customFields.add('Apttus_Config2__PricingStatus__c');
23
24             assetActDO.CustomFields = customFields;
25
26             Apttus_Config2__LineItem__c liSO = new
Apttus_Config2__LineItem__c();
27             liSO.Apttus_Config2__Comments__c = 'Comments Added my Modify
Asset API Code';
28             liSO.Apttus_Config2__PricingStatus__c = 'Complete';
29             assetActDO.CustomData = liSO;
30
31             assetActDOList.add(assetActDO);
32         }
33     }
34     //For a specific cart ID, invoke the API and modify the assets based
on the terminate function above.
35     Apttus_CPQApi.CPQ.ModifyAssetRequestDO modifyRequest = new
Apttus_CPQApi.CPQ.ModifyAssetRequestDO();
36     modifyRequest.CartId = cartID;
37     modifyRequest.AssetActions = assetActDOList;
38
39     Apttus_CPQApi.CPQ.ModifyAssetResponseDO modifyReponse =
Apttus_CPQApi.CPQWebservice.modifyAsset(modifyRequest);
40
41     List<Decimal> lineNumberList = modifyReponse.LineNumbers;
42
43     List<Apttus_CPQApi.CPQ.AssetActionDO> assetList =
modifyReponse.AssetActions;
44

```

```

45     Boolean bIsPending = false;
46
47     Integer iCounter = 0;
48
49     while (iCounter < 1000)
50     {
51         bIsPending = false;
52         for(Apptus_CPQApi.CPQ.AssetActionDO objAsset: assetList)
53         {
54             if(objAsset.Pending == true)
55             {
56                 bIsPending = true;
57                 break;
58             }
59         }
60         //if asset action is pending execute the loop below
61         if(bIsPending == true)
62         {
63             modifyReponse =
Apttus_CPQApi.CPQWebservice.modifyAsset(modifyRequest);
64             lineNumberList = modifyReponse.LineNumbers;
65             assetList = modifyReponse.AssetActions;
66         }
67         else
68         {
69             break;
70         }
71         iCounter++;
72     }
73
74     bindAssets();
75 }
76
77 //Use the function below to bind the updated asset items.
78 public void bindAssets()
79 {
80     List<Apttus_Config2__AssetLineItem__c> aliSOList = [select id,
Apttus_Config2__Description__c,
81     Apttus_Config2__Quantity__c, Apttus_Config2__StartDate__c,
Apttus_Config2__EndDate__c from Apttus_Config2__AssetLineItem__c
82     where Apttus_Config2__AccountID__c = :accountID];
83
84     lstAssetWrapper = new List<AssetWrapperClass>();
85
86     for(Apttus_Config2__AssetLineItem__c aliS0 : aliSOList)

```

```

87     {
88         AssetWrapperClass objAssetWrapperClass = new AssetWrapperClass ();
89         objAssetWrapperClass.AssetId = aliSO.ID;
90         objAssetWrapperClass.ProductId =
aliSO.Apttus_Config2__Description__c;
91         objAssetWrapperClass.Quantity = aliSO.Apttus_Config2__Quantity__c;
92         objAssetWrapperClass.StartDate =
aliSO.Apttus_Config2__StartDate__c;
93         objAssetWrapperClass.EndDate = aliSO.Apttus_Config2__EndDate__c;
94
95         lstAssetWrapper.Add(objAssetWrapperClass);
96     }
97 }

```

The upgrade asset action enables you to update selected asset items. The sample below enables you to upgrade the asset. Using the sample below you can fetch the asset items using the `getAssetsForSearchText` API and upgrade the selected assetLineItems using `AssetAction = Upgrade`. You can upgrade the date and the quantity for an asset line item.

```

public static void processForUpgrade(ID cartId, ID accountId)
{
    //Declaration
    Set<ID> assetLineItemIds = new Set<ID>();

    Apttus_CPQApi.CPQ.AssetSearchResultDO result
    = Apttus_CPQApi.CPQWebService.getAssetsForSearchText(accountId, null, null);

    Apttus_CPQApi.CPQ.ModifyAssetRequestDO assetRequest = new
Apttus_CPQApi.CPQ.ModifyAssetRequestDO();
    assetRequest.CartId = cartId;

    for (Apttus_Config2__AssetLineItem__c assetItemSO : result.AssetItems)
    {
        Apttus_CPQApi.CPQ.AssetActionDO assetAction = new
Apttus_CPQApi.CPQ.AssetActionDO();
        assetAction.AssetId = 'Upgrade';
        assetAction.AssetId = assetItemSO.Id;
        assetAction.Quantity = assetItemSO.Apttus_Config2__Quantity__c.intValue();
        assetAction.EndDate = Date.today();
        assetAction.Pending = true;
    }
}

```

```

        assetLineItemIds.add(assetItemS0.Id);
        assetRequest.AssetActions.add(assetAction);
    }

    Apttus_CPQApi.CPQ.ModifyAssetResponseDO assetResponse =
    Apttus_CPQApi.CPQWebService.modifyAsset(assetRequest);

    Set<Decimal> lineNumbers = new Set<Decimal>(assetResponse.LineNumbers);

    for (Apttus_Config2__LineItem__c lineS0 : [Select Id,
    Apttus_Config2__AssetLineItemId__c from Apttus_Config2__LineItem__c Where
    Apttus_Config2__ConfigurationId__c = :cartId])
    {

    }
}

```

## Computing Shipping for Cart Line Items

This API enables you to calculate the shipping amount for an entire order and does not display the breakup for each line item.

API	Signature
computeShippingForCart	<pre> webService static Apttus_CPQApi.CPQ.ComputeShippingResponseDO computeShippingForCart(Apttus_CPQApi.CPQ.ComputeShippingRequestDO request) </pre>

Parameters		
Name	Type	Description
request	Apttus_CPQApi.CPQ.ComputeShippingRequestDO	The compute shipping request.

Request Data Object - Apttus_CPQApi.CPQ.ComputeShippingRequestDO		
Field	Type	Description
CartID	ID	The id of the cart.

Response Data Object - Apttus_CPQApi.CPQ.ComputeShippingResponseDO		
Field	Type	Description
ShippingResults	Map<ID, Apttus_Config2.CustomClass.ShippingResult>	The shipping result object populated by the line item id. This is only available when called from Apex.
TotalShippingItem	Apttus_Config2__LineItem__c	The line item with total taxes.

Data Object - Apttus_Config2.CustomClass.ShippingResult		
Field	Type	Description
Handback	Object	<p>Any Apex Object passed by the calling class and passed back to the calling class in the result.</p> <p>Handback is used only in case when the shipping charge needs to be calculated for an object other than Cart line item e.g. Billing Schedule line items.</p> <ul style="list-style-type: none"> <li>• For calculating shipping charges for a cart, there is no need to pass the Handback object. It can be blank</li> <li>• For calculating the shipping charges for objects other than line items, "Item" field will be blank and the actual object will be passed as handback object to identify the object for which shipping charges are calculated.</li> </ul>



Data Object - Apttus_Config2.CustomClass.ShippingResult		
Field	Type	Description
ShippingCharge	Decimal	Shipping amount total of all the cart line items. This is the total shipping amount for the line item. Line item is identified by the sequence in which the shipping result is added. This needs to be calculated and populated by the implementation.

### Code Sample

The sample below enables you to compute shipping for a cart with a valid cartID. Using the sample below, you can compute shipping for the entire cart using the values fetched from the shipping callback class. If the shipping charge is location dependent, ensure that the account associated with the order or proposal has a valid shipping to and billing to address. For more information about Tax and Shipping scenarios refer, [Updating Taxes and Shipping for an Order](#).

```

1  public void computeShippingForCart()
2  {
3      if(String.isNotBlank(cartId))
4      {
5          // Start timer
6          startTime = System.currentTimeMillis();
7
8          // Create the request
9          Apttus_CPQApi.CPQ.ComputeShippingRequestDO request = new
Apttus_CPQApi.CPQ.ComputeShippingRequestDO();
10
11         // Add request parameters
12         request.CartId = cartId;
13
14         // Compute Shipping For Cart
15         Apttus_CPQApi.CPQ.ComputeShippingResponseDO result =
Apttus_CPQApi.CPQWebService.computeShippingForCart(request);
16
17         // End timer
18         ResponseTime('computeShippingForCart');
19
20         // Get Total Tax Item

```

```

21         ApexPages.addMessage(new
    ApexPages.Message(ApexPages.severity.info, 'Total Shipping Item : ' +
result.TotalShippingItem));
22
23         Map<ID, Apttus_Config2.CustomClass.ShippingResult> shippingResults
= new Map<ID, Apttus_Config2.CustomClass.ShippingResult>();
24
25         shippingResults = result.ShippingResults;
26
27         for(Id sId : shippingResults.keySet()) {
28
29             Apttus_Config2.CustomClass.ShippingResult shippingResult =
shippingResults.get(sId);
30
31             ApexPages.addMessage(new
    ApexPages.Message(ApexPages.severity.info, 'Shipping Charges : ' +
shippingResult.ShippingCharge));
32         }
33     }
34     else
35     {
36         ApexPages.addMessage(new
    ApexPages.Message(ApexPages.severity.info, 'Invalid or blank cart Id.));
37     }
38 }

```

## Computing Taxes for Cart Line Items

This API enables you to calculate tax breakups for cart line item.

API	Signature
<b>computeTaxForCart</b>	<i>webservice static Apttus_CPQApi.CPQ.ComputeTaxResponseDO computeTaxForCart(Apttus_CPQApi.CPQ.ComputeTaxRequestDO request)</i>

Parameters		
Name	Type	Description
request	Apttus_CPQApi.CPQ.ComputeTaxRequestDO	The compute tax request object.

Request Data Object - Apttus_CPQApi.CPQ.ComputeTaxRequestDO		
Field	Type	Description
CartID	ID	The id of the cart.

Response Data Object - Apttus_CPQApi.CPQ.ComputeTaxResponseDO		
Field	Type	Description
TaxResults	Map<ID, Apttus_Config2.CustomClass.TaxResult>	The Tax Result object populated by the line item id. This is only available when called from Apex.
TotalTaxItem	Apttus_Config2__LineItem__c	The line item with total taxes.

Data Object - Apttus_Config2.CustomClass.TaxResult		
Field	Type	Description
Handback	Object	<p>Any Apex Object passed by the calling class and passed back to the calling class in the result.</p> <p>Handback is used only in case when the tax needs to be calculated for an object other than Cart line item e.g. Billing Schedule line items.</p> <ul style="list-style-type: none"> <li>• For calculating tax for cart line items, there is no need to pass the Handback object. It can be blank</li> <li>• For calculating the tax for object other than line items, "Item" field will be blank and the actual object will be passed as handback object to identify object for which tax is calculated.</li> </ul>
TaxAmount	Decimal	<p>This is the total amount for the line item. This is sum of the breakup tax amounts for the line item. Line item is identified by the sequence in which the tax result is added. This needs to be calculated and populated by the implementation.</p>
TaxBreakups	List<Apttus_Config2__TaxBreakup_c>	<p>Tax breakup provide the different components of the tax for a given line item. This needs to be implemented and provided by the implementation. Tax breakup has structure specified below.</p>

Data Object - Apttus_Config2__TaxBreakup_c		
Field	Type	Description
BreakupType	String	If you specify the breakup type as Detail, then the quote or order line items show the detailed tax break up for a cart line item.
TaxAmount	Decimal	The tax amount calculated based on the formulae defined in the callback.
TaxAppliesTo	String	The field on which the tax is applied to. For example, net price.
TaxRate	Decimal	Tax percentage to be applied
TaxType	String	Type of tax to be applied. For example, sales, custom, excise.

### Tax Breakup Type Example

Breakup Type	Tax Type	Tax Rate	Tax Applies to	Tax Amount
Detail	State Tax	10	Net Price	100
Detail	City Tax	1	Net Price	10
Total				110

### Code Sample

The sample below enables you to compute tax for a cart with a valid cartID. Using the sample below, you can compute tax for the entire cart using the values fetched from the tax callback class. The fields Taxable, Tax Inclusive, and Tax Code are inherited from the quote or order line item. The tax code must be populated on the Price list item for each product and charge type combination. The Account and Account Location has a Tax Exempt indicator that overrides everything. For example, if the Account has a Tax Exempt flag enabled, then Tax is not calculated for the account. The system gives preference to location unless it is blank. If a product is not taxable or tax inclusive, the system will skip that line item.

For more information about Tax and Shipping scenarios refer, [Updating Taxes and Shipping for an Order](#).

```
1 public void computeTaxForCart()
2 {
3     if(String.isNotBlank(cartId))
4     {
5
6         // Create the request
7         Apttus_CPQApi.CPQ.ComputeTaxRequestDO request = new
Apttus_CPQApi.CPQ.ComputeTaxRequestDO();
8
9         // Add request parameters
10        request.CartId = cartId;
11
12        // Compute Tax for cart
13        Apttus_CPQApi.CPQ.ComputeTaxResponseDO result =
Apttus_CPQApi.CPQWebService.computeTaxForCart(request);
14
15
16        // Get Total Tax Item
17        ApexPages.addMessage(new
ApexPages.Message(ApexPages.severity.info, 'Total Tax Item : ' +
result.TotalTaxItem));
18
19        Map<ID, Apttus_Config2.CustomClass.TaxResult> taxResults = new
Map<ID, Apttus_Config2.CustomClass.TaxResult>();
20        lstTaxBreakUpWrapper = new List<TaxBreakUpWrapper>();
21
22        taxResults = result.TaxResults;
23
24        for(ID tId : taxResults.keySet())
25        {
26            Apttus_Config2.CustomClass.TaxResult taxResult =
taxResults.get(tId);
27
28            // Tax Amount
29            ApexPages.addMessage(new
ApexPages.Message(ApexPages.severity.info, 'Tax Amount : ' +
taxResult.TaxAmount));
30
```

```

31         // Tax BreakUps
32         for(Apttus_Config2__TaxBreakup__c taxBreakUp :
taxResult.TaxBreakups)
33             {
34
35                 TaxBreakUpWrapper temp = new TaxBreakUpWrapper();
36                 temp.TaxBreakUpId = taxBreakUp.Id;
37                 temp.TaxType = taxBreakUp.Apttus_Config2__TaxType__c;
38                 temp.TaxAppliedTo =
taxBreakUp.Apttus_Config2__TaxAppliesTo__c;
39                 temp.TaxRate = taxBreakUp.Apttus_Config2__TaxRate__c;
40                 temp.TaxAmount = taxBreakUp.Apttus_Config2__TaxAmount__c;
41                 temp.Sequence = taxBreakUp.Apttus_Config2__Sequence__c;
42
43                 lstTaxBreakUpWrapper.add(temp);
44             }
45         }
46     }
47     else
48     {
49         ApexPages.addMessage(new
ApexPages.Message(ApexPages.severity.info, 'Invalid or blank cart Id.));
50     }
51 }

```

## Adding a Miscellaneous Item to the Cart

This API creates and assigns a miscellaneous item to the cart.

API	Signature
<b>addMiscItem</b>	<i>webservice static Apttus_CPQApi.CPQ.AddMiscItemResponseDO addMiscItem(Apttus_CPQApi.CPQ.AddMiscItemRequestDO request)</i>

Parameters		
Name	Type	Description
request	Apttus_CPQApi.CPQ.AddMiscItemRequestDO	This is the request data object.

Request Data Object - Apttus_CPQApi.CPQ.AddMiscItemRequestDO		
Field	Type	Description
AllocateGroupAdjustment	Boolean	Indicates whether group adjustment should be allocated.
AllowRemoval	Boolean	Indicates whether removing the miscellaneous item is allowed.
Amount	Decimal	The Base Price for the miscellaneous item.
CartId	Id	The id of the cart where you want to add the miscellaneous item.
ChargeType	String	The charge type of the miscellaneous item.
Description	String	Description of the miscellaneous item.



Response Data Object - Apttus_CPQApi.CPQ.AddMiscItemResponseDO		
Field	Type	Description
MiscItem	Apttus_Config2__LineItem__c	The miscellaneous item that is created.

### Code Sample

The sample code below enables you to add a miscellaneous item to a cart by passing proposal, charge type, description, and base amount.

```

1  /**
2   * The below method demonstrates how to add a misc to an existing cart
3   * (every quote has a cart)
4   * Let's assume that the quote cart is already added Laptop as a bundle
5   * product and its two options are Keyboard and Mouse.
6   * Inside this method, we will add delivery charge as misc item to the
7   * cart.
8   */
9  public Apttus_CPQApi.CPQ.AddMiscItemResponseDO addMiscItem(String
10 proposalName, String chargeType, String description, Decimal amount)
11 {
12     Apttus_Config2__ProductConfiguration__c cart = [SELECT Id
13                                                     FROM
14 Apttus_Config2__ProductConfiguration__c
15                                                     WHERE
16 Apttus_QPConfig__ProposalId__r.Name = :proposalName LIMIT 1];
17
18     // Create the request object
19     Apttus_CPQApi.CPQ.AddMiscItemRequestDO request = new
20 Apttus_CPQApi.CPQ.AddMiscItemRequestDO();
21
22     // add misc attributes to the request
23     request.CartId = cart.Id;
24     request.ChargeType = ChargeType;
25     request.Description = Description;
26     request.Amount = amount;

```

```

20     request.AllocateGroupAdjustment = true;
21     request.AllowRemoval = true;
22
23     // create the response object
24     Apttus_CPQApi.CPQ.AddMiscItemResponseDO response =
Apttus_CPQApi.CPQWebService.addMiscItem(request);
25
26     return response;
27 }
28

```

## Creating Coupons for Incentives

You can use this API to create coupons for incentives.

i When you generate coupons using the API, if any duplicate coupon code is generated, the job fails and you need to execute the job again.

API	Signature
createCouponsForIncentives	<i>webService static  Apttus_CPQApi.CPQ.CreateCouponsForIncentiveResponse  DO  createCouponsForIncentives(Apttus_CPQApi.CPQ.CreateC  ouponsForIncentiveRequestDO request)</i>

Request Parameter		
Name	Type	Description
request	Apttus_CPQApi.CPQ.CreateCouponsForIncentiveRequestDO	The request data object.

Request Data Object - Apttus_CPQApi.CPQ.CreateCouponsForIncentiveRequestDO		
Name	Type	Description
IncentiveIds	List	The IDs of the incentives for which you want to generate coupons.
NumCoupons	List	The number of coupons you want to generate. For example, if you enter 10000, CPQ will generate 10000 coupons for the specified incentive.

Response Parameter Data Object - Apttus_CPQApi.CPQ.CreateCouponsForIncentiveResponseDO		
Name	Type	Description
Coupons	List	The list of generated coupon codes.
IsAsync	Boolean	Indicates whether coupons were created in async mode or not. <ul style="list-style-type: none"> <li>• If there are &lt;10k coupons, CPQ returns Async as False.</li> <li>• If there are &gt;10k coupons, CPQ returns Async as True.</li> <li>• If there are multiple incentives, CPQ returns Async as True.</li> </ul>
JobId	Id	The ID of the job created to generate coupons.

### Code Sample

The sample code below demonstrates the coupons created in sync mode and async mode.

1	<code>//Sync Test</code>
2	

```

3  Apttus_CPQApi.CPQ.CreateCouponsForIncentiveRequestDO request = new
    Apttus_CPQApi.CPQ.CreateCouponsForIncentiveRequestDO();
4  request.IncentiveIds.add(incentiveS0.Id);
5  // add corresponding number of coupons parameter
6  request.NumCoupons.add(5);
7
8  Apttus_CPQApi.CPQ.CreateCouponsForIncentiveResponseDO result =
    Apttus_CPQApi.CPQWebService.createCouponsForIncentives(request);
9
10 //Async Test
11
12 Apttus_CPQApi.CPQ.CreateCouponsForIncentiveRequestDO request2 = new
    Apttus_CPQApi.CPQ.CreateCouponsForIncentiveRequestDO();
13 request2.IncentiveIds.add(incentiveS0.Id);
14 request2.IncentiveIds.add(incentiveS02.Id);
15
16 // add corresponding number of coupons parameter
17 request2.NumCoupons.add(5);
18 request2.NumCoupons.add(5);
19
20 Apttus_CPQApi.CPQ.CreateCouponsForIncentiveResponseDO result2 =
    Apttus_CPQApi.CPQWebService.createCouponsForIncentives(request2);

```

## Retrieving Products Included by Auto-inclusion Constraint Rule

This processes and applies all the constraint rules.

**Best Practice**

It is recommended that you use the [associateConstraintRules API](#) before using this API. Directly using this API may not trigger the constraint rule.

API	Signature
applyConstraintRules	<i>webService static void applyConstraintRules(Id cartId, Boolean finalCheck)</i>

Parameters		
Name	Type	Description
cartId	ID	The id of the cart.
finalCheck	Boolean	If this is set to true, runs rules that are marked as check on finalization.

### Code Sample

The sample below enables you to apply constraint rules to a cart with a specific cart ID. If you set the finalCheck flag as true, the constraint rules are run when you finalize the cart. If the flag is set as false, the rules are run before cart finalization. For example, if you want to add a Shipping Costs constraint rule only after the cart is finalized, set the finalCheck as true and the shipping costs constraint rule is applied once the user finalized the cart. If you want to apply constraint rules before the cart is finalized, set finalCheck as false. For example, if you set the finalCheck flag as false, installation charges are added along with a product selected before the cart is finalized.

```

1  public void applyConstraintRules()
2  {
3      // For rules that are not marked as Check on Finalization
4      Apttus_CPQApi.CPQWebService.applyConstraintRules(cartID, false);
5
6      // For rules that are marked as Check on Finalization
7      //Apttus_CPQApi.CPQWebService.applyConstraintRules(cartID, true);
8  }
```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## API Prerequisites

- [Creating a Cart from a Quote](#)
- [Associating Constraint Rules to a Cart](#)

## Response/Request XML

### Example Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:cpq="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService">
  <soapenv:Header>
    <cpq:SessionHeader>
      <cpq:sessionId>00D4P000000z7dE!
AQQAQFudWwmQHka0g8qt7T4KJ9MmTK0J0550XmfoUk9bUEL_idltBYg5muQuM4Pm0HVjinAgttLfi55uyxVSv
F5yrkoH.rH4</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <cpq:applyConstraintRules>
      <cpq:cartId>a1l4P00000Bg7CkQAJ</cpq:cartId>
      <cpq:finalCheck>>false</cpq:finalCheck>
    </cpq:applyConstraintRules>
  </soapenv:Body>
</soapenv:Envelope>
```

### Example Response

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns="http://soap.sforce.com/schemas/class/Apttus_CPQApi/CPQWebService">
  <soapenv:Body>
    <applyConstraintRulesResponse/>
  </soapenv:Body>
</soapenv:Envelope>
```

## Adding Line Items

You can use this API to add line items to a specified cart. The line items can be for bundles, standalone products, and options.

**i** This API does not create the associated Product Attribute Value record, you must call the **createNewAttributes** API right after executing this API.

API	Signature
<b>addLineItems</b>	<i>webService static Apttus_CPQApi.CPQ.AddMultiProductResponseDO addLineItems(Id cartId, List lineItems, Apttus_CPQApi.CPQ.AddLineItemsRequestDO optionalParams)</i>

Parameters		
Field	Type	Description
cartId	ID	The ID of the cart.
lineItems	List<Apttus_Config2__LineItem__c>	List of LineItems SObject records to be added to the Cart.
optionalParams	Apttus_CPQApi.CPQ.AddLineItemsRequestDO	Additional parameter that must not be used for this API.

Response Data Object - Apttus_CPQApi.CPQ.AddMultiProductResponseDO		
Field	Type	Description
LineNumbers	List<Decimal>	Contains List of Line Numbers of the newly added Line Items

### Code Sample

The sample code below enables you to add a bundle and an option line item. You must provide values for the following mandatory fields:

- For bundle
  - ConfigurationId\_\_c
  - ProductId\_\_c
- For option
  - ConfigurationId\_\_c
  - OptionId\_\_c
  - ParentBundleNumber\_\_c

You can set relevant values on any field when passing the Line Item record in the request. However, you should not set system fields here such as Line Number, Primary Line Number, and Pricing related fields. CPQ manages these fields internally.

```

1  // prepare bundle line for input
2  Product2 bundleProd = [SELECT Id FROM Product2 WHERE Name = 'Patient
  Accounting'];
3  Apttus_Config2__LineItem__c bundleLineItemS0 = new
  LineItem__c(Apttus_Config2__ConfigurationId__c = getCartS0().Id);
4  bundleLineItemS0.Apttus_Config2__ProductId__c = bundleProd.Id;
5
6  // prepare option line for input
7  Product2 optionProd = [SELECT Id FROM Product2 WHERE Name = 'Dell 360'];
8  Apttus_Config2__LineItem__c optionLineItemS0 = new
  Apttus_Config2__LineItem__c(Apttus_Config2__ConfigurationId__c =
  getCartS0().Id);
9  optionLineItemS0.Apttus_Config2__ProductId__c = bundleProd.Id;
10 optionLineItemS0.Apttus_Config2__OptionId__c = optionProd.Id;
11 optionLineItemS0.Apttus_Config2__ParentBundleNumber__c = 1;

```



```

12
13 // add bundle line item
14 List<Decimal> bundleLineNumbers = CPQWebService.addLineItems (
15     getCartSO().Id,
16     new List<Apttus_Config2__LineItem__c>{bundleLineItemSO});
17
18 // add option line item
19 List<Decimal> optionLineNumbers = Apttus_CPQApi.CPQWebService.addLineItems
20     (
21     getCartSO().Id,
22     new List<Apttus_Config2__LineItem__c>{optionLineItemSO});

```

## Rebuilds the Category Hierarchy

This API is used to rebuilds the category hierarchy for the given list of hierarchy ids.

API	Signature
rebuildCategoryHierarchies	<i>WebService static void rebuildCategoryHierarchies(List&lt;ID&gt; hierarchyIds)</i>

Parameters			
Name	Type	Required	Description
hierarchyIds	List<Id>	Yes	The list of hierarchy ids to rebuild.

### Response Data

Void, doesn't return any valueResponse

### Code Sample

The following sample code retrieves the list of category records for the given category IDs, rebuilds the categories, and updates the data accordingly.

```

1 public void rebuildCategories(List<Id> categoryIds)
2     {

```

```

3
Apttus_CPQApi.CPQWebService.rebuildCategoryHierachies(categoryIds);
4
    }
```

## CPQ Web Service (Apttus\_Config2)

The CPQ web service APIs account for the standard actions to configure, price, and quote.

You can invoke APIs in CPQ Web Service(Apttus\_Config2) from the following command:

```

Apttus_Config2.CPQWebService.<Name of the Function>
where the name of the function is API Name and it's parameters.
```

You can use the CPQ web service APIs to complete the following tasks:

- [Creating Product Line Items](#)
- [Registering Split Cart Parameters](#)
- [Finalizing the Configuration Version](#)
- [Finalizing the Configuration Version with Synchronization Tasks](#)
- [Abandoning a Configuration](#)
- [Cloning Primary Line Items](#)
- [Creating Bundle Line Items](#)
- [Creating and Updating Related Line Items](#)
- [Saving Related Line Items](#)
- [Splitting Related Line Items](#)

## Creating Product Line Items

This API creates product line items.

API	Signature
<b>createProductLineItems</b>	<i>webService static List createProductLineItems(Apttus_Config2__ProductConfigura tion__c configSO, Apttus_Config2.CPQStruct.ProductCollDO prodCollDO)</i>

Request Parameter		
Name	Type	Description
configSO	Apttus_Config2__ProductConfiguration__c	The ID of product configuration subject.
prodCollDO	Apttus_Config2.CPQStruct.ProductCollDO	The product details request data object

Request Data Object - Apttus_Config2.CPQStruct.ProductCollDO		
Name	Type	Description
Products	Apttus_Config2.CPQStruct.ProductLineItemDO	The bundle product request data object
HasOptionProducts	Boolean	Indicate whether has option products or not.
ProductIds	List<ID>	The list of product Ids

Data Object - Apttus_Config2.CPQStruct.ProductLineItemDO		
Name	Type	Description
AddedBy	String	Indicate how was the product added. For example, added by constraint rule, APIs.
Comments	String	The comments added in the product.
CustomData	Apttus_Config2__LineItem__c	The custom data based on list of custom fields to initialize Line Item record.
CustomFields	List<String>	List of Custom Fields.
EndDate	Date	The end date of the product.

Data Object - Apttus_Config2.CPQStruct.ProductLineItemDO		
Name	Type	Description
productOptionComponentId	ID	The ID of the Product Option Component (only for option products).
ProductSO	Product2	The product subject.
Quantity	Decimal	The quantity of the product.
SellingTerm	Decimal	The selling term of the product.
StartDate	Date	The start date of the product.

Response Parameters		
Field	Type	Description
productLineItems	List<Apttus_Config2__LineItem__c>	The list of IDs for the new product line items.

### Code Sample

The sample code below enables you to create product line items for an existing cart. After the API is executed, the list of line items is returned without committing the changes into Salesforce. These line items can be used for further modification and later associate to cart by performing insert operation.

```

1  /**
2   * The below method demonstrates how to create bundle lineitems with
   * option (every quote has a cart)
3   * Let's assume the Quote's Cart is not blank and Laptop is a bundle
   * product and its two options are Keyboard and Mouse
4   * The input of this method is Quote Number and the bundle and related
   * option product names

```

```

5      * Inside the method we will create the line items for the Laptop bundle
6      product and also its two options Keyboard and Mouse
7      */
8      public List<Apttus_Config2__LineItem__c> createProductLineItems(String
9      proposalID, List<String> bundleProductName)
10     {
11         Apttus_Config2__ProductConfiguration__c cart = [SELECT
12         Apttus_Config2__PriceListId__c,
13                                     Id,
14         Apttus_Config2__SummaryGroupType__c,
15         Apttus_Config2__ContractNumbers__c,
16         Apttus_Config2__EffectiveDate__c,
17         Apttus_Config2__PurchaseId__c,
18         Apttus_Config2__ExpectedStartDate__c,
19         Apttus_Config2__ExpectedEndDate__c,
20         Apttus_Config2__EffectivePriceListId__c,
21         Apttus_Config2__PriceListId__r.Apttus_Config2__CostModelId__c
22                                     FROM
23         Apttus_Config2__ProductConfiguration__c
24                                     WHERE
25         Apttus_QPConfig__ProposalId__r.Name = :proposalID
26                                     LIMIT 1];
27
28         List<Apttus_Config2.CPQStruct.ProductLineItemDO> prodItems = new
29         List<Apttus_Config2.CPQStruct.ProductLineItemDO>();
30         for(Product2 ProductS0 : [SELECT ID,
31                                     Apttus_Config2__Uom__c,
32                                     Name,
33                                     Apttus_Config2__Customizable__c,
34                                     Apttus_Config2__HasDefaults__c,
35                                     Apttus_Config2__HasAttributes__c,
36                                     Apttus_Config2__HasOptions__c
37                                     FROM Product2
38                                     WHERE Name IN :bundleProductName])
39     {

```

```

34         prodItems.add(new
Apttus_Config2.CPQStruct.ProductLineItemDO(ProductSO, 10, 1, null, null,
'Comment from API'));
35     }
36
37     Apttus_Config2.CPQStruct.ProductCollDO prodCollDO = new
Apttus_Config2.CPQStruct.ProductCollDO(prodItems);
38     List<Apttus_Config2__LineItem__c> productLineItems =
Apttus_Config2.CPQWebService.createProductLineItems(cart,prodCollDO);
39
40     return productLineItems;
41 }
    
```

## Registering Split Cart Parameters

The global method *setSplitCartInfo* is used to group the line items in the cart for pricing. You can define the parameters, namely the threshold and the split criteria based on which you want to split the line items in the cart through the API. If you do not define the parameters, the values of settings **Split Cart Threshold** and **Split Cart Criteria Fields** in Config System Properties are used to split the line items instead.

API	Signature
<b>setSplitCart Info</b>	<i>static void setSplitCartInfo(Id cartId, Apttus_Config2.CustomClass.SplitCartInfo splitInfo)</i>

Parameters			
Name	Type	Required?	Description
cartId	ID	Yes	The id of the Cart you want to split.
splitInfo	Apttus_Config2.CustomClass. CustomClass.SplitCartInfo	Yes	The request object.

Request Object - Apttus_Config2.CustomClass.SplitCartInfo		
Field	Type	Description
SplitCriteriaFields	List	The list of API names of the fields to use as criteria.
SplitThreshold	Integer	The number of line items in a group.

### Code Sample

The sample below enables you to set the values of **Split Cart Threshold** and **Split Cart Criteria Fields** to define the information required to split a cart.

```

1 //Setting Split info
2 Apttus_Config2.CustomClass.SplitCartInfo splitInfo = new
  Apttus_Config2.CustomClass.SplitCartInfo();
3 splitInfo.SplitCriteriaFields.add('Apttus_Config2_LocationId_c');
4 splitInfo.SplitThreshold = 10;
5 //Calling SplitInfo
6 // cartID can be Main Cart Id or Config Cart Id
7 ID cartID = 'a1I2f00000029Iy';
8 Apttus_Config2.CPQWebService.setSplitCartInfo(cartID , splitInfo);

```

## Finalizing the Configuration Version

This API finalizes the configuration version and related line items. After the configuration is finalized, an email notification is sent when concurrent access enabled for the cart. The cart should be configured and priced before invoking this API. You can create a button to finalize the cart which the user can click to invoke this API once the products in the cart are adjusted and final.

API	Signature
finalizeConfiguration	<i>webService static void finalizeConfiguration(Id configId)</i>

Request Parameter		
Name	Type	Description
configId	ID	The ID of product configuration you want to finalize.

## Code Sample

The sample code below enables you to finalize a cart with a specific configuration and synchronizes the products (line items) added to the cart with the quote or proposal used to generate the cart.

```
1  /**
2   * Below method demonstrate to finalize the cart by passing the
3   * configuration request Id.
4   */
5  public void finalizeCart(Id configId)
6  {
7      // finalize the cart
8      Apttus_Config2.CPQWebService.finalizeConfiguration(configId);
9  }
```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## API Prerequisites

- [Creating a Cart from a Quote](#)
- [Updating Price](#)



## Response/Request XML

### Example Request

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:cpq="http://soap.sforce.com/schemas/class/Apttus_Config2/CPQWebService">
  <soapenv:Header>
    <cpq:SessionHeader>
      <cpq:sessionId>00DZ000000NAEIA!
ASAAQN7k3xhhvMe.j.8gpR.ijcGq77HJeF4MEDfbxbBAqZ8r4WWNTv30Vb6o1bjtHJLbq5mvHcuAH_ie6sTC3
DzrWgnLdUuD</cpq:sessionId>
    </cpq:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <cpq:finalizeConfiguration>
      <cpq:configId>a10Z0000002YrHD</cpq:configId>
    </cpq:finalizeConfiguration>
  </soapenv:Body>
</soapenv:Envelope>
```

### Example Response

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_Config2/CPQWebService">
  <soapenv:Body>
    <finalizeConfigurationResponse/>
  </soapenv:Body>
</soapenv:Envelope>
```

## Finalizing the Configuration Version with Synchronization Tasks

This API finalizes the configuration version and finalizes the cart. This API also execute async Action Callback. Based on cart type (large cart/normal cart) the callback will be executed synchronously or asynchronously. The cart should be configured and priced before

invoking this API. You can create a button to finalize the cart which the user can click to invoke this API once the products in the cart are adjusted and final.

API	Signature
<b>finalizeConfigurationAndSync</b>	<i>webService static void finalizeConfigurationAndSync(Id configId, Id requestId, String finalizeClassName)</i>

Request Parameter		
Name	Type	Description
configId	ID	The ID of product configuration you want to finalize.
requestId	ID	The ID of the request of the configuration you want to finalize.
finalizeClassName	String	The finalize Callback class name.

### Code Sample

The sample code below enables you to finalize a cart with the configuration you provided. After the finalization product line items added to the cart are synchronized with the quote or proposal used to generate the cart. The action defined in the callback class you mentioned in the **finalizeClassName** parameter is also executed. Based on cart type (large cart/normal cart) the callback that implements *CustomClass.IActionCallback* and *CustomClass.IActionCallback2* interfaces are executed synchronously or asynchronously.

```

1  /**
2   * Below method demonstrate to finalize the cart by passing the
   configuration request Id.
3  */
4
5  public void finalizeConfigurationAndSync (Id configId, Id requestId,
String finalizeClassName)
```

```

6  {
7      // finalize the cart
8      Apttus_Config2.CPQWebService.finalizeConfigurationAndSync(configId,
9      requestId, finalizeClassName);
    }

```

## Abandoning a Configuration

This API abandons the product configuration using the ID. The record is kept or deleted based on the configuration in the **Keep Abandoned Carts** field in Config System Properties. If **Keep Abandoned Carts** is enabled, the API changes the status of the product configuration to *Abandoned*, removes unreferenced totaling groups, and writes activity logs by creating a task. The API cancels the Quote Collaboration requests along with abandoning the cart. The API also executes any callbacks that must be executed before deleting a line item.

API	Signature
abandonConfiguration	<i>webService static void abandonConfiguration(Id configId)</i>

Request Parameter		
Name	Type	Description
configId	ID	The ID of product configuration you want to abandon.

### Code Sample

The below sample code allows you to delete the configuration record and the related line item associated with the configuration. In this case, the **Keep Abandoned Carts** field is disabled.

```

1  /**
2  * The below method demonstrates how to delete product configuration.

```

```

3  */
4  public void abandonConfigurationID (String proposalName)
5  {
6      Apttus_Config2__ProductConfiguration__c productConfig = [SELECT Id
7                                                                FROM
8                                                                Apttus_Config2__ProductConfiguration__c
9                                                                WHERE
10     Apttus_QPConfig__ProposalId__r.Name = :proposalName
11                                                                LIMIT 1];
12     Apttus_Config2.CPQWebService.abandonConfiguration (productConfig.Id);
13 }

```

## Cloning Primary Line Items

This global method clones line items from one configuration to another using Line Number.

API	Signature
clonePrimaryItemColl	<i>static Id clonePrimaryItemColl(Id configId, Integer lineNumber, Id targetConfigId)</i>

Request Parameter		
Name	Type	Description
configId	Id	The ID of source product configuration associated with the line items.
lineNumber	Integer	The Line Number of the line items you want to clone.
targetConfigId	Id	The ID of the product configuration you want to clone the line items into.

Response Parameters		
Field	Type	Description
clonedPrimaryLineNumber	Id	The Line number of the cloned primary line item collection

### Code Sample

The sample code below enables you to clone line items associated with a Line Number and add them to the target configuration. This returns the line number of the cloned line items.

```

1  /**
2   * Clones the given primary line item collection and adds it onto the
   target configuration
3   * @param parentConfigId the id of the product configuration associated
   with the line items.
4   * @param primaryLineNumber the line number associated with the item
   collection
5   * @param childConfigId the id of child product configuration
6   * @return the Id of the new primary line item collection
7   */
8
9   public Id clonePrimaryItemCollection(Id parentConfigId, Integer
primaryLineNumber, Id childConfigId)
10  {
11      Id clonedPrimaryLineNumber;
12      clonedPrimaryLineNumber =
Apttus_Config2.CPQWebService.clonePrimaryItemColl(parentConfigId,
primaryLineNumber, childConfigId);
13      return clonedPrimaryLineNumber;
14  }
```

## Creating Bundle Line Items

This API creates the bundle and option line items based on the bundle product data object.

API	Signature
<b>createBundleLineItems</b>	<i>webservice static List createBundleLineItems(Apttus_Config2__ProductConfigurati on__c configSO, Apttus_Config2.CPQStruct.ProductBundleDO bundleDO)</i>

Request Parameter		
Name	Type	Description
configSO	Apttus_Config2__ProductConfigurati on__c	The ID of product configuration subject.
bundleDO	Apttus_Config2.CPQStruct.ProductBu ndleDO	The bundle details request data object

Request Data Object - Apttus_Config2.CPQStruct.ProductBundleDO		
Name	Type	Description
BundleProduct	Apttus_Config2.CPQStruct.Pro ductLineItemDO	The bundle product request data object
HasOptionProducts	Boolean	Indicate whether has option products or not.
OptionProductIds	List<ID>	The list of option product Ids
OptionProducts	ListList<ProductLineItemDO>	The list of option product data objects.

Data Object - Apttus_Config2.CPQStruct.ProductLineItemDO		
Name	Type	Description
AddedBy	String	Indicate how was the product added. For example, added by constraint rule, APIs.
Comments	String	The comments added in the product.
CustomData	Apttus_Config2__LineItem__c	The custom data based on list of custom fields to initialize Line Item record.
CustomFields	List<String>	List of Custom Fields.
EndDate	Date	The end date of the product.
productOptionComponentId	ID	The ID of the Product Option Component (only for option products).
ProductSO	Product2	The product subject.
Quantity	Decimal	The quantity of the product.
SellingTerm	Decimal	The selling term of the product.
StartDate	Date	The start date of the product.
Response Parameters		
Field	Type	Description
lineItems	List<Apttus_Config2__LineItem__c>	The list of IDs for the new bundle line items.

### Code Sample

The sample code below enables you to create bundle and option line items for an existing cart. This API returns the list of line items without committing changes into Salesforce.

These line items can be used for further modification and later associate to cart by performing insert operation.

```

1  /**
2   * The below method demonstrates how to create bundle lineitems with
3   * option (every quote has a cart)
4   * Let's assume the Quote's Cart is not blank and Laptop is a bundle
5   * product and its two options are Keyboard and Mouse
6   * The input of this method is Quote Number and the bundle and related
7   * option product names
8   * Inside the method we will return the line items for the Laptop bundle
9   * product and also its two options Keyboard and Mouse
10  */
11  public List<Apttus_Config2__LineItem__c> createBundleLineItems(String
12  proposalID, String bundleProductName, List<String> optionProductNames)
13  {
14      Apttus_Config2__ProductConfiguration__c cart = [SELECT
15      Apttus_Config2__PriceListId__c,
16
17      Id,
18
19      Apttus_Config2__SummaryGroupType__c,
20
21      Apttus_Config2__ContractNumbers__c
22      FROM
23      Apttus_Config2__ProductConfiguration__c
24      WHERE
25      Apttus_QPConfig__ProposalId__r.Name = :proposalID
26      LIMIT 1];
27
28      //Add bundle product data object
29      Product2 bundleProductS0 = [SELECT ID,
30
31      Apttus_Config2__Uom__c,
32      Name,
33      Apttus_Config2__Customizable__c,
34
35      Apttus_Config2__HasDefaults__c,
36      Apttus_Config2__HasAttributes__c,
37      Apttus_Config2__HasOptions__c
38      FROM Product2
39      WHERE Name = :bundleProductName
40      LIMIT 1];

```



```

28     Apttus_Config2.CPQStruct.ProductLineItemDO BundleProduct = new
    Apttus_Config2.CPQStruct.ProductLineItemDO( bundleProductS0,
29                                     10, //
    quantity assigned to the product
30                                     1, //
    selling term for the product
31                                     null, //
    Start date
32                                     null, //
    End date
33                                     'Comment
    from API');
34     List<Apttus_Config2.CPQStruct.ProductLineItemDO> optionProducts = new
    List<Apttus_Config2.CPQStruct.ProductLineItemDO>();
35     for (Product2 optionProductS0 : [SELECT ID,
36                                     Apttus_Config2__Uom__c,
37                                     Name,
38                                     Apttus_Config2__Customizable__c,
39                                     Apttus_Config2__HasDefaults__c,
40                                     Apttus_Config2__HasAttributes__c,
41                                     Apttus_Config2__HasOptions__c
42                                     FROM Product2
43                                     WHERE Name IN :optionProductNames])
44     {
45         optionProducts.add(new
    Apttus_Config2.CPQStruct.ProductLineItemDO(optionProductS0,5,1, null,null,'
    Comment from API'));
46     }
47     Apttus_Config2.CPQStruct.ProductBundleDO bundleDO = new
    Apttus_Config2.CPQStruct.ProductBundleDO(BundleProduct, optionProducts);
48     List<Apttus_Config2__LineItem__c> lineItems =
    Apttus_Config2.CPQWebService.createBundleLineItems(cart,bundleDO);
49     return lineItems;
50 }

```

## Creating and Updating Related Line Items

You can use this global method to create related line items records of an asset and associate them with a given service product. You can also delete the related line item. You can create and delete related line items for multiple service products in a single call, however, the number of line items must adhere to the Salesforce Governor Limits.

API	Signature	
updateRelatedLineItems	static Apttus_Config2.CPQStruct.UpdateRelatedLineItemResponseDO updateRelatedLineItems(Apttus_Config2.CPQStruct.UpdateRelatedLineItemRequestDO requestDO)	
<b>Parameters</b>		
Name	Type	Description
requestDO	Apttus_Config2.CPQStruct.UpdateRelatedLineItemRequestDO	Request related line item details data object that is invoked by the method.
<b>Request Data Object - Apttus_Config2.CPQStruct.UpdateRelatedLineItemResponseDO</b>		
Name	Type	Description
CartId	ID	The ID of the product configuration.
RelatedLineItemColls	Map<Id, List<Apttus_Config2.RelatedLineItemColl>>	Map of service line item id to the list of data objects containing relevant details of the asset associated with the given service line item.
<b>Data Object - Apttus_Config2.RelatedLineItemColl</b>		
Name	Type	Description
Action	String	Define one of the following strings to specify whether associate or dissociate the asset from the service line item: <ul style="list-style-type: none"> <li>• <i>add</i></li> <li>• <i>remove</i></li> </ul>
AssetLineItemId	Id	The Id of the asset line item

Data Object - Apttus_Config2.RelatedLineItemColl		
Name	Type	Description
RelatedLineItemSO	Apttus_Config2.RelatedLineItemColl	The Subject with the updated values to be saved in the database.

Response Data Object - Apttus_Config2.CPQStruct.UpdateRelatedLineItemResponseDO		
Field	Type	Description
Errors	List <Strings>	List of errors encountered while creating or deleting the related line item
isSuccess	Boolean	Indicates whether the related line items were created or deleted successfully.

## Code Sample

The code sample below helps you create and delete a related line item.

```
public void addRemoveRelatedLineItems(String lineItemId, String assetId1, String
assetId2, String assetId3, String cartId)
{
    // Use LineAction.Add to create new RLI to associate an Asset with Service Line
    Item
    Apttus_Config2.CPQStruct.RelatedLineItemColl relatedLineItemD01 = new
Apttus_Config2.CPQStruct.RelatedLineItemColl();
    relatedLineItemD01.AssetLineItemId = assetId1;
    relatedLineItemD01.Action = 'Add';

    Apttus_Config2.CPQStruct.RelatedLineItemColl relatedLineItemD02 = new
Apttus_Config2.CPQStruct.RelatedLineItemColl();
```

```

relatedLineItemD02.AssetId = assetId2;
relatedLineItemD02.Action = 'Add';

// Use LineAction.Remove to delete existing RLI for dissociating an Asset from
Service Line
Apttus_Config2.CPQStruct.RelatedLineItemColl relatedLineItemD03 = new
Apttus_Config2.CPQStruct.RelatedLineItemColl();
relatedLineItemD03.AssetId = assetId3;
relatedLineItemD03.Action = 'Delete';

// Prepare the request structure to be passed to the API
Apttus_Config2.CPQStruct.UpdateRelatedLineItemRequestD0 requestD0 = new
Apttus_Config2.CPQStruct.UpdateRelatedLineItemRequestD0();
requestD0.RelatedLineItemColls.put(lineItemId, new
List<Apttus_Config2.CPQStruct.RelatedLineItemColl>{relatedLineItemD01,
relatedLineItemD02, relatedLineItemD03});
requestD0.CartId = cartId;

// Invoke the API by passing the request formed above
Apttus_Config2.CPQStruct.UpdateRelatedLineItemResponseD0 responseD0 =
Apttus_Config2.CPQWebService.updateRelatedLineItems(requestD0);

// Check the response for errors if any
if (!responseD0.IsSuccess) {
    System.debug('Errors thrown :');
    if (responseD0.Errors != null && responseD0.Errors.size() > 0) {
        for (String strError : responseD0.Errors) {
            System.debug('\nError - ' + strError);
        }
    }
}
}
}

```

## Saving Related Line Items

You can use this global method to save modified related line items. You can modify the following fields in the related line items:

- **Weightage Type**
- **Weightage Percent**
- **Weightage Amount**
- Any custom fields

API	Signature	
saveRelatedLineItems	<pre>static Apttus_Config2.CPQStruct.SaveRelatedLineItemResponseDO saveRelatedLineItems(Apttus_Config2.CPQStruct.SaveRelatedLineItemRequestDO requestDO)</pre>	
Parameters		
Name	Type	Description
requestDO	Apttus_Config2.CPQStruct.SaveRelatedLineItemRequestDO	The related line item request data object.
Request Data Object - Apttus_Config2.CPQStruct.SaveRelatedLineItemRequestDO		
Name	Type	Description
CartId	ID	The ID of the product configuration where you want to save the modified related line item.
RelatedLineItem	List<Apttus_Config2.RelatedLineItemColl>	List of related line items to be saved in the database.
Data Object - Apttus_Config2.RelatedLineItemColl		
Name	Type	Description
Action	String	Define one of the following strings to specify whether associate or dissociate the asset from the service line item: <ul style="list-style-type: none"> <li>• <i>add</i></li> <li>• <i>remove</i></li> </ul>
AssetLineItemId	Id	The Id of the asset line item

Data Object - Apttus_Config2.RelatedLineItemColl		
Name	Type	Description
RelatedLineItemSO	Apttus_Config2__RelatedLineItem__c	The Subject with the updated values of related line items to be saved in the database.

Response Data Object - Apttus_Config2.CPQStruct.SaveRelatedLineItemResponseDO		
Field	Type	Description
Errors	List <Strings>	List of errors encountered while saving the related line item
isSuccess	Boolean	Indicates whether the related line items were saved successfully.

### Code Sample

The code sample below helps you save related line items.

```

public void saveRelatedLineItems(String cartId, String lineItemId) {

    // Prepare the request structure to be passed to the API
    Apttus_Config2.CPQStruct.SaveRelatedLineItemRequestDO requestDO = new
    Apttus_Config2.CPQStruct.SaveRelatedLineItemRequestDO();
    requestDO.CartId = cartId;

    List<Apttus_Config2__RelatedLineItem__c> relatedLineItemSOs = [SELECT
    Apttus_Config2__LineItemId__c, Apttus_Config2__WeightageType__c,
    Apttus_Config2__WeightageAmount__c FROM Apttus_Config2__RelatedLineItem__c WHERE
    Apttus_Config2__LineItemId__c =: lineItemId];
    
```

```

        // below is simple example logic to set the Weightage Amount and setting
Weightage type to Amount
        // Customers can have their use cases implemented here
        for (Apttus_Config2__RelatedLineItem__c relatedLineItemSO :
relatedLineItemSOs) {
            relatedLineItemSO.Apttus_Config2__WeightageType__c = 'Amount';
            relatedLineItemSO.Apttus_Config2__WeightageAmount__c = 35;
            Apttus_Config2.CPQStruct.RelatedLineItemColl relatedLineItemDO = new
Apttus_Config2.CPQStruct.RelatedLineItemColl();
            relatedLineItemDO.RelatedLineItemSO = relatedLineItemSO;

            requestDO.RelatedLineItems.add(relatedLineItemDO);
        }

        // Invoke the API by passing the request formed above
        Apttus_Config2.CPQStruct.SaveRelatedLineItemResponseDO responseDO =
Apttus_Config2.CPQWebService.saveRelatedLineItems(requestDO);

        // Check the response for errors if any
        if (!responseDO.IsSuccess) {
            System.debug('Errors thrown :');
            if (responseDO.Errors != null && responseDO.Errors.size() > 0) {
                for (String strError : responseDO.Errors) {
                    System.debug('\nError - ' + strError);
                }
            }
        }
    }
}

```

## Splitting Related Line Items

You can use this global method to clone a service line item for each related line item associated with that service line item.

API	Signature
<b>splitRelatedLineItems</b>	<i>static Apttus_Config2.CPQStruct.SplitRelatedLineItemResponseDO splitRelatedLineItems(Apttus_Config2.CPQStruct.SplitRelatedLineItemRequestDO requestDO)</i>

Parameters		
Name	Type	Description
requestDO	Apttus_Config2.CPQStruct.SplitRelatedLineItemRequestDO	The request data object of related line item details invoked by the method.
Request Data Object - Apttus_Config2.CPQStruct.SplitRelatedLineItemRequestDO		
Name	Type	Description
CartId	ID	The ID of the product configuration record.
Flow	String	The flow you want to use to validate the split criteria fields.
LineNumber	Integer	The <b>Line Number</b> of the service line item you want to split.
SplitCriteriaFields	List<String>	List of some or all fields defined in <b>Split Criteria Fields</b> in Config System Properties to be used as criteria to split the service line item.
Response Data Object - Apttus_Config2.CPQStruct.SplitRelatedLineItemResponseDO		
Field	Type	Description
Errors	List <Strings>	List of errors encountered while splitting the service line item
isSuccess	Boolean	Indicates whether the splitting of service line item was successful.

### Code Sample

The code sample below helps you split a service line item.

```
public void splitServiceCart(String cartId, String splitCriteriaFieldName)
```



```

{

    // Prepare the request structure to be passed to the API
    Apttus_Config2.CPQStruct.SplitRelatedLineItemRequestDO request = new
Apttus_Config2.CPQStruct.SplitRelatedLineItemRequestDO();
    request.CartId = cartId;

    // Service Line number to split
    request.LineNumber = 1;

    // this can take 00TB CPQ fields
    request.SplitCriteriaFields.add(splitCriteriaFieldName);

    // flow to validate the split criteria fields
    request.Flow = 'QuotingCartGrid';

    // Invoke the API by passing the request formed above
    Apttus_Config2.CPQStruct.SplitRelatedLineItemResponseDO responseDO =
Apttus_Config2.CPQWebService.splitRelatedLineItems(request);

    // Check the response for errors if any
    if (!responseDO.IsSuccess) {
        System.debug('Errors thrown :');
        if (responseDO.Errors != null && responseDO.Errors.size() > 0) {
            for (String strError : responseDO.Errors) {
                System.debug('\nError - ' + strError);
            }
        }
    }
}
}

```

## CPQ Admin Web Service

The CPQ Admin web service APIs account for the standard actions to configuring products.

You can invoke APIs in CPQ Admin Web Service from the following command:

```
Apttus_Config2.CPQAdminWebService.<Name of the Function>
where the name of the function is API Name and it's parameters.
```

You can use the CPQ Admin web service APIs to complete the following tasks:

- [Creating New Products](#)
- [Creating New Categories](#)
- [Associating Products to Category](#)
- [Building Hierarchy of Products](#)
- [Checking if Product Exists in Hierarchy](#)
- [Removing Products from Hierarchy](#)
- [Retrieving the List Option Group](#)
- [Retrieving Options in an Option Group](#)
- [Retrieving of Product Hierarchy](#)
- [Retrieves List of Parent Product](#)
- [Retrieves List Product IDs](#)
- [Retrieves List of Category IDs](#)
- [Creating New Attribute Group](#)
- [Adding and Removing Attributes from an Attribute Group](#)
- [Associating Attribute Group and Products](#)
- [Dissociating Attribute Group and Products](#)

## Creating New Products

You can use this API to create products.

API	Signature
createProducts	<i>webservice static</i> <i>Apttus_Config2.CPQAdminStruct.ProductResponseDO</i> <i>createProducts(Apttus_Config2.CPQAdminStruct.ProductRequestDO productRequestDO)</i>

Request Parameter		
Name	Type	Description
productRequestDO	Apttus_Config2.CPQAdminStruct.ProductRequestDO	The create products request data object.

Request Data Object - Apttus_Config2.CPQAdminStruct.ProductRequestDO		
Name	Type	Description
ProductDOs	List<Apttus_Config2.CPQAdminStruct.ProductDO>	The list of product data objects.
Data Object - Apttus_Config2.CPQAdminStruct.ProductDO		
Name	Type	Description
Name	String	The name of the product
ConfigType	String	Type of configuration of the product. You can use the following values: <ul style="list-style-type: none"> <li>• Standalone</li> <li>• Bundle</li> <li>• Option</li> </ul>
IsActive	Boolean	Indicates whether the product should be active or inactive.
OptionalFields	List<Apttus_Config2.CPQAdminStruct.MapDO>	The list of optional field values. You can use this parameter to associate additional field values with the product.
CategoryId	Id	The Id of the category to which the products must be associated to.
Response Parameter - Apttus_Config2.CPQAdminStruct.ProductResponseDO		
Field	Type	Description
ProductStructureJSON	List<Apttus_Config2.CPQAdminStruct.MapDO>	The list of key-value pairs. The key refers to the product name that was passed in the request and the value refers to the Id of the product.

Response Data Object - Apttus_Config2.CPQAdminStruct.MapDO		
Field	Type	Description
String	Key	The name of the product.
String	Value	The value refers to the Id of the product.

## Code Sample

The sample code below enables you to create products

```

1  /**
2   * The below code demonstrates how to create one standalone and one bundle
   products for the given category Id
3   */
4
5  Public void createProducts()
6  {
7
8      Apttus_Config2.CPQAdminStruct.ProductRequestDO request = new
Apttus_Config2.CPQAdminStruct.ProductRequestDO();
9      Apttus_Config2.CPQAdminStruct.ProductDO productDO = new
Apttus_Config2.CPQAdminStruct.ProductDO();
10     List<Apttus_Config2.CPQAdminStruct.MapDO> listOfMapDO = new
List<Apttus_Config2.CPQAdminStruct.MapDO>();
11     Apttus_Config2.CPQAdminStruct.MapDO m = new
Apttus_Config2.CPQAdminStruct.MapDO();
12
13     // ADD 1ST PRODUCT
14     // required fields
15     productDO.Name = 'Standalone Product Name';
16     productDO.ConfigType = 'Standalone';
17     productDO.IsActive = true;
18
19     // additional fields (Please use namespace for the fields outside your
ORG)
20     // 1st additional field
21     m.Key = 'Description';

```

```
22     m.Value = 'Standalone product description';
23     listOfMapD0.add(m);
24
25     // 2nd additional field
26     m = new Apttus_Config2.CPQAdminStruct.MapD0();
27     m.Key = ' Apttus_Config2__HasOptions__c';
28     m.Value = 'false';
29     listOfMapD0.add(m);
30
31     // 3rd additional field
32     m = new Apttus_Config2.CPQAdminStruct.MapD0();
33     m.Key = ' Apttus_Config2__EffectiveDate__c';
34     m.Value = '2016-01-13 02:28:00';
35     listOfMapD0.add(m);
36     // add additional fields to the optionalFields list
37     productD0.OptionalFields.addAll(listOfMapD0);
38
39     // ADD 2ND PRODUCT
40     request.ProductD0s.add(productD0);
41     productD0 = new Apttus_Config2.CPQAdminStruct.ProductD0();
42
43     // required fields
44     productD0.Name = 'Bundle Product Name';
45     productD0.ConfigType = 'Bundle';
46     productD0.IsActive = true;
47
48     // additional fields
49     m = new Apttus_Config2.CPQAdminStruct.MapD0();
50
51     // 1st additional field
52     m.Key = 'Description';
53     m.Value = 'Bundle product description';
54     listOfMapD0.add(m);
55
56     // 2nd additional field
57     m = new Apttus_Config2.CPQAdminStruct.MapD0();
58     m.Key = 'Apttus_Config2__HasOptions__c';
59     m.Value = 'true';
60     listOfMapD0.add(m);
61
62     // 3rd additional field
63     m = new Apttus_Config2.CPQAdminStruct.MapD0();
64     m.Key = 'Apttus_Config2__EffectiveDate__c';
65     m.Value = '2021-01-01 15:24:00';
66     listOfMapD0.add(m);
```

```

67
68     // add optional fields to the optionalFields list
69     productD0.OptionalFields.addAll(listOfMapD0);
70     productD0.categoryID = 'a0i6C000001tnac'; // Id of any existing
category
71
72     // add product to request
73     request.ProductD0s.add(productD0);
74     Apttus_Config2.CPQAdminStruct.ProductResponseD0 response =
Apttus_Config2.CPQAdminWebService.createProducts(request);
75 }
    
```

## Creating New Categories

You can use this API to create new categories with the minimum required parameters like name, type, and label.

API	Signature
<code>createCategories()</code>	<i>webService static Apttus_Config2.CPQAdminStruct.CategoryResponseDO createCategories(Apttus_Config2.CPQAdminStruct.CategoryR equestDO categoryRequestDO)</i>

Request Parameter		
Name	Type	Description
categoryRequestDO	Apttus_Config2.CPQAdminStruct.C ategoryRequestDO	The category request data object.

Request Data Object - Apttus_Config2.CPQAdminStruct.CategoryRequestDO		
Name	Type	Description
CategoryDOs	List<Apttus_Config2.CPQAdminStruct.CategoryDO>	The list of category data objects.
Response Parameter - Apttus_Config2.CPQAdminStruct.CategoryResponseDO		
Field	Type	Description
CategoryDOs	List<Apttus_Config2.CPQAdminStruct.CategoryDO>	The list of category data objects.
CategoryResponseMap	List<Apttus_Config2.CPQAdminStruct.MapDO>	The list of key-value pairs. The key refers to the category name that was passed in the request and the value refers to the Id of the category.
Data Object - Apttus_Config2.CPQAdminStruct.CategoryDO		
Name	Type	Description
Label	String	The label of the category
Name	String	The name of the category
OptionalFields	List<Apttus_Config2.CPQAdminStruct.MapDO>	The list of optional field values. You can use this parameter to associate additional field values with the category.
Type	String	Type of the category. You can use the following values: <ul style="list-style-type: none"> <li>• Offering</li> <li>• Option Group</li> <li>• Both</li> </ul>

Response Data Object - Apttus_Config2.CPQAdminStruct.MapDO		
Field	Type	Description
String	Key	The name of the category.
String	Value	The value refers to the Id of the category.

## Code Sample

The below sample enables you to create categories. You can also add optional fields, add the namespace to the field names.

```

1  /**
2   * The below code create categories of type Option Group and Offering
3   */
4
5  Public void createCategories()
6  {
7      Apttus_Config2.CPQAdminStruct.CategoryResponseDO categsResponse = null;
8      Apttus_Config2.CPQAdminStruct.CategoryRequestDO categsRequest = new
Apttus_Config2.CPQAdminStruct.CategoryRequestDO();
9      Apttus_Config2.CPQAdminStruct.CategoryDO categDO = null;
10     List<Apttus_Config2.CPQAdminStruct.MapDO> mapDOList = null;
11     Apttus_Config2.CPQAdminStruct.MapDO mapDO = null;
12
13     // CREATE 3 CATEGORIES:
14     // CREATING 1ST CATEGORY
15     // necessary fields
16     categDO = new Apttus_Config2.CPQAdminStruct.CategoryDO();
17     categDO.Name = 'CategoryOptionGroup1';
18     categDO.Label = 'CategoryOptionGroup1';
19     categDO.Type = 'Option Group';
20
21     mapDOList = new List<Apttus_Config2.CPQAdminStruct.MapDO>();
22
23     // additional fields
24     // 1st
25     mapDO = new Apttus_Config2.CPQAdminStruct.MapDO();

```



```

26     mapDO.Key = 'Apttus_Config2__GuidePage__c';
27     mapDO.Value = 'a page';
28     mapDOList.add(mapDO);
29
30     // 2nd
31     mapDO = new Apttus_Config2.CPQAdminStruct.MapDO();
32     mapDO.Key = 'Apttus_Config2__Active__c';
33     mapDO.Value = 'true';
34     mapDOList.add(mapDO);
35
36     // add categDO to categsRequest
37     categDO.OptionalFields = mapDOList;
38     categsRequest.CategoryDOs.add(categDO);
39
40
41     // CREATING 2ND CATEGORY
42     // necessary fields
43     categDO = new Apttus_Config2.CPQAdminStruct.CategoryDO();
44     categDO.Name = 'CategoryOffering1';
45     categDO.Label = 'CategoryOffering1';
46     categDO.Type = 'Offering';
47
48     // add categDO to categsRequest
49     categsRequest.CategoryDOs.add(categDO);
50
51
52     // CREATING 3RD CATEGORY
53     // necessary fields
54     categDO = new Apttus_Config2.CPQAdminStruct.CategoryDO();
55     categDO.Name = 'CategoryOptionGroup2';
56     categDO.Label = 'CategoryOptionGroup2';
57     categDO.Type = 'Option Group';
58
59     // add categDO to categsRequest
60     categsRequest.CategoryDOs.add(categDO);
61
62     // create categories
63     categsResponse =
Apttus_Config2.CPQAdminWebService.createCategories(categsRequest);
64 }

```

## Associating Products to Category

You can use this API to associate a product to a category.

API		Signature
associateProductToCategory		<i>webservice static Boolean associateProductToCategory(Apttus_Config2.CPQAdminStruct.ProductRequestDO productsRequestDO)</i>
Request Parameter		
Name	Type	Description
productsRequestDO	Apttus_Config2.CPQAdminStruct.ProductRequestDO	The category request data object.
Data Object - Apttus_Config2.CPQAdminStruct.ProductDO		
Name	Type	Description
ProductId	Id	The ID of the product you want to associate with the category
CategoryId	Id	The Id of the category to which the products must be associated to.

Response Parameter		
Name	Type	Description
IsSuccessful	Boolean	Indicates whether the association of product to the category was successful or not.

## Code Sample

The below sample code enables you to associate the product to a specific category.

```

1  /**
2   * The below code associate the products with related category. Invoke
   below method by passing map of productid and categoryId as parameter.
3   */
4
5  Public Boolean associateProductToCategory(Map<Id, Id>
   mapProductIdCategoryId)
6  {
7     Apttus_Config2.CPQAdminStruct.ProductRequestDO prodsAssocRequest = new
   Apttus_Config2.CPQAdminStruct.ProductRequestDO();
8     CPQAdminStruct.ProductDO prodForAssociation = null;
9     For (Id productId : mapProductIdCategoryId.keySet()) {
10        prodForAssociation.productID = prodIDs[i];
11        prodForAssociation.categoryID = categIDs[i];
12        prodsAssocRequest.ProductDOs.add(prodForAssociation);
13    }
14
15    return
16
   Apttus_Config2.CPQAdminWebService.associateProductToCategory(prodsAssocReq
   uest);
16 }

```

## Building Hierarchy of Products

You can use this API to build a hierarchy of products based on a list of parent and child pairs.

API	Signature
buildHierarchy	<i>webService static</i> <i>Apttus_Config2.CPQAdminStruct.HierarchyResponseDO</i> <i>buildHierarchy(Apttus_Config2.CPQAdminStruct.HierarchyRequ</i> <i>estDO hierarchyRequestDO)</i>

Request Parameter		
Name	Type	Description
hierarchyRequestDO	Apttus_Config2.CPQAdminStruct.HierarchyRequestDO	The hierarchy request data object.

Request Data Object - Apttus_Config2.CPQAdminStruct.HierarchyRequestDO		
Name	Type	Description
HierarchyDOs	List<Apttus_Config2.CPQAdminStruct.HierarchyDO>	the list of hierarchy data object
isCascadeGroupChanges	Boolean	Indicates whether to reflect the changes made to this instance of the option group to all of its sibling option groups across products.
OptionGroupId	Id	The Id of the option group.
OptionGroupIds	List <Id>	The list of option group Ids to be associated with the bundle product.
ProductId	Id	The Id of the product to be the parent in the hierarchy.
ProductIds	List <Id>	The list of child product Ids.
Data Object - Apttus_Config2.CPQAdminStruct.HierarchyDO		
Name	Type	Description
ChildOptionGroupId	Id	The Id of the child option child group.
ChildProductId	Id	The Id of the child product.

Data Object - Apttus_Config2.CPQAdminStruct.HierarchyDO		
Name	Type	Description
ParentOptionGroupId	Id	The Id of the parent option group.
ParentProductId	Id	The Id of the parent product.
Response Parameter - Apttus_Config2.CPQAdminStruct.HierarchyResponseDO		
Field	Type	Description
ProductOptionGroupDO	Apttus_Config2.CPQAdminStruct2.ProductOptionGroupDO	The parent option group data object.
Success	Boolean	Indicates whether building hierarchy was successful or not.
Data Object - Apttus_Config2.CPQAdminStruct2.ProductOptionGroupDO		
Field	Type	Description
ChildOptionGroupDOs	List<Apttus_Config2.CPQAdminStruct2.ProductOptionGroupDO>	The list of child option group data object.
ChildProductOptionGroupSOs	List<Apttus_Config2__ProductOptionGroup__c>	The list of child product option groups.
ComponentSOs	List<Apttus_Config2__ProductOptionComponent__c>	The list of product option components.

Data Object - Apttus_Config2.CPQAdminStruct2.ProductOptionGroupDO		
Field	Type	Description
ProductOptionComponentDOs	List<Apttus_Config2.CPQAdminStruct2.ProductOptionComponentDO>	The list of product option components data objects.
ProductOptionGroupSO	Apttus_Config2__ProductOptionGroup__c	The sObject of the product option group.

Data Object - Apttus_Config2.CPQAdminStruct2.ProductOptionComponentDO		
Field	Type	Description
DefaultQuantityExpressionSO	Apttus_Config2__FieldExpression__c	The sObject of default quantity expression.
MaxQuantityExpressionSO	Apttus_Config2__FieldExpression__c	The sObject of max quantity expression
MinQuantityExpressionSO	Apttus_Config2__FieldExpression__c	The sObject of the product option group.
ProductOptionComponentSO	Apttus_Config2__ProductOptionComponent__c	The sObject of the product option component.

**Code Sample**

The below sample code demonstrates how to build a hierarchy on parent and child products.

```

1  /**
2   * The below method accepts parent productname and list of option child
   products and categoryhierarchy name as input and return true or false as
   status of the build.
3  */
4  public Boolean buildHierarchy(String parentProductName, List<String>
   optionProducts, String categoryHierarchyName)
5  {
6     Apttus_Config2.CPQAdminStruct.HierarchyDO hierDO = new
   Apttus_Config2.CPQAdminStruct.HierarchyDO();
7     Apttus_Config2.CPQAdminStruct.HierarchyRequestDO hierRequest = new
   Apttus_Config2.CPQAdminStruct.HierarchyRequestDO();
8     Apttus_Config2.CPQAdminStruct.HierarchyResponseDO hierResponse;
9     List< Apttus_Config2__ProductOptionComponent__c> poc;
10
11    // STEP 1 - Get bundle product and options
12    Product2 bundleProduct = [SELECT Id, Name
13                               FROM Product2
14                               WHERE Name = :parentProductName
15                               LIMIT 1];
16    Product2 options = [SELECT Id, Name
17                          FROM Product2
18                          WHERE Name IN : optionProducts];
19
20    // STEP 2 - Get an option group to create a bundle with the 3 options
   from above
21    Apttus_Config2__ ClassificationHierarchy__c optionGroup = [SELECT Id,
   Name
22                                                                FROM
   Apttus_Config2__ ClassificationHierarchy__c
23                                                                WHERE Name
   = :categoryHierarchyName
24                                                                LIMIT 1];
25
26    // STEP 3 - Create Bundle by associating Option Group and Product
27    hierRequest.ProductId = bundleProduct.Id;
28    hierRequest.OptionGroupIds.add(optionGroup.Id);
29
30    // send request and save response

```

```

31     hierResponse =
Apttus_Config2.CPQAdminWebService.buildHierarchy(hierRequest);
32
33     // STEP 4 - Add options to the newly created bundle
34     hierRequest = new Apttus_Config2.CPQAdminStruct.HierarchyRequestDO();
35     hierRequest.ProductId = bundleProduct.Id;
36     hierRequest.OptionGroupId = pog[0].Id;
37     for (Product2 option : options) {
38         hierRequest.ProductIds.add(option.Id);
39     }
40     hierResponse =
Apttus_Config2.CPQAdminWebService.buildHierarchy(hierRequest);
41
42     return hierResponse.Success;
43 }

```

## Checking if Product Exists in Hierarchy

You can use this API to check if a product exist in a category.

API	Signature
productExistsInHierarchy	<i>webService static Apttus_Config2.CPQAdminStruct.HierarchyResponseDO productExistsInHierarchy(Apttus_Config2.CPQAdminStruct. HierarchyRequestDO hierarchyRequestDO)</i>

Request Parameter		
Name	Type	Description
hierarchyRequestDO	Apttus_Config2.CPQAdminStruct.HierarchyRequestDO	The category request data object.



Request Parameter - Apttus_Config2.CPQAdminStruct.HierarchyRequestDO		
Name	Type	Description
HierarchyDOs	List<Apttus_Config2.CPQAdminStruct.HierarchyDO>	The list of hierarchy data object

Data Object- Apttus_Config2.CPQAdminStruct.HierarchyDO		
Name	Type	Description
ChildProductId	Id	The Id of child product.
ParentProductId	Id	The Id of the parent product.

Response Data Object- Apttus_Config2.CPQAdminStruct.HierarchyResponseDO		
Name	Type	Description
Issuccess	Boolean	Indicates whether the product exists in the hierarchy or not.

### Code Sample

The below sample code finds whether the child product is available in the hierarchy of its parent product.

```

1  public Boolean productExistsInHierarchy(String parentProductName, String
2  childProductName)
3  {
4      Apttus_Config2.CPQAdminStruct.HierarchyDO hierDO = new
5      Apttus_Config2.CPQAdminStruct.HierarchyDO();
6      Apttus_Config2.CPQAdminStruct.HierarchyRequestDO hierRequest = null;
7      Apttus_Config2.CPQAdminStruct.HierarchyResponseDO hierResponse = null;
8
9      // get child, parent products, and option group

```

```

8      Product2 parentProduct = [SELECT Id FROM Product2 WHERE Name
= :parentProductName LIMIT 1];
9      Product2 childProduct = [SELECT Id FROM Product2 WHERE Name
= :childProductName LIMIT 1];
10
11     hierRequest = new Apttus_Config2.CPQAdminStruct.HierarchyRequestDO();
12     hierDO = new Apttus_Config2.CPQAdminStruct.HierarchyDO();
13     hierDO.ParentProductId = parentProduct.Id;
14     hierDO.ChildProductId = childProduct.Id;
15     hierRequest.HierarchyDOs.add(hierDO);
16     hierResponse =
Apttus_Config2.CPQAdminWebService.productExistsInHierarchy(hierRequest);
17     return hierResponse.Issuccess;
18 }

```

## Removing Products from Hierarchy

You can use this API to remove a product from a category.

API	Signature
removeProductFromHierarchy	<i>webservice static Apttus_Config2.CPQAdminStruct.HierarchyResponseDO removeProductFromHierarchy(Apttus_Config2.CPQAdminS truct.HierarchyRequestDO hierarchyRequestDO)</i>

Request Parameter		
Name	Type	Description
hierarchyRequestDO	Apttus_Config2.CPQAdminStruct.Hierarch yRequestDO	The category request data object.

Request Parameter - Apttus_Config2.CPQAdminStruct.HierarchyRequestDO		
Name	Type	Description
HierarchyDOs	List<Apttus_Config2.CPQAdminStruct.Hierarchy DO>	The list of hierarchy data object

Data Object- Apttus_Config2.CPQAdminStruct.HierarchyDO		
Name	Type	Description
ChildProductId	Id	The Id of child product.
ParentProductId	Id	The Id of the parent product.

Response Data Object- Apttus_Config2.CPQAdminStruct.HierarchyResponseDO		
Name	Type	Description
Issuccess	Boolean	Indicates whether the removal of the product from the hierarchy.

### Code Sample

The below sample code enables you to remove the products from their relevant category.

1	<code>/**</code>
2	<code> * The below code removes the products from category. Invoke below method</code>
3	<code> by passing parent and child product names.</code>
4	<code> * Remove a child product from a hierarchy of a parent product.</code>
5	<code> * The input request contains the Child Product Id and the Parent Product</code>
6	<code> Id.</code>
7	<code> * In case a list of products is passed, this will be an all-or-none</code>
8	<code> scenario stored in</code>
9	<code> * the single Boolean return value.</code>
10	<code>*/</code>

```

8
9  public Boolean removeProductFromHierarchy(String parentProductName, String
childProductName)
10 {
11     Apttus_Config2.CPQAdminStruct.HierarchyDO hierDO = new
Apttus_Config2.CPQAdminStruct.HierarchyDO();
12     Apttus_Config2.CPQAdminStruct.HierarchyRequestDO hierRequest = new
Apttus_Config2.CPQAdminStruct.HierarchyRequestDO();
13     Apttus_Config2.CPQAdminStruct.HierarchyResponseDO removeResponse =
null;
14
15     // STEP 1 - Retrieve data
16     // get child, parent products, and option group
17     Product2 parentProduct = [SELECT Id
18                               FROM Product2
19                               WHERE Name = :parentProductName
20                               LIMIT 1];
21     Product2 childProduct = [SELECT Id
22                               FROM Product2
23                               WHERE Name = :childProductName
24                               LIMIT 1];
25
26     // STEP 2 - Remove child from hierarchy
27     // build request object
28     hierDO = new Apttus_Config2.CPQAdminStruct.HierarchyDO();
29     hierDO.ParentProductId = parentProduct.Id;
30     hierDO.ChildProductId = childProduct.Id;
31     hierDO.ParentOptionGroupId = null;
32     hierDO.ChildOptionGroupId = null;
33     // add hierarchy DO to request
34     hierRequest.HierarchyDOs.add(hierDO);
35
36     // send request for removal and save response
37     removeResponse =
Apttus_Config2.CPQAdminWebService.removeProductFromHierarchy(hierRequest);
38     return removeResponse.Issuccess;
39 }

```

## Retrieving the List Option Group

You can use this API to retrieve the list Option Group associated with a product.

API	Signature
getChildOptionGroups	<i>webservice static Apttus_Config2.CPQAdminStruct.HierarchyResponseDO getChildOptionGroups(Apttus_Config2.CPQAdminStruct.HierarchyRequestDO hierarchyRequestDO)</i>

Request Parameter		
Name	Type	Description
hierarchyRequestDO	Apttus_Config2.CPQAdminStruct.HierarchyRequestDO	The option group request data object.
Request Parameter - Apttus_Config2.CPQAdminStruct.HierarchyRequestDO		
Name	Type	Description
ProductID	Ids	The ID of the product for which you want to retrieve option groups.

Response Data Object- Apttus_Config2.CPQAdminStruct.HierarchyResponseDO		
Name	Type	Description
OptionGroupId	List<Id>	The Id option associated with the product.

### Code Sample

The below sample code returns the list of option group Ids that are associated with a given product name.

```

1  public Apttus_Config2.CPQAdminStruct.HierarchyResponseDO
   getChildOptions(String productName, String optionGroupName)
2  {
3      Apttus_Config2.CPQAdminStruct.HierarchyDO hierDO = new
   Apttus_Config2.CPQAdminStruct.HierarchyDO();
4      Apttus_Config2.CPQAdminStruct.HierarchyRequestDO hierRequest = new
   Apttus_Config2.CPQAdminStruct.HierarchyRequestDO();
5      Apttus_Config2.CPQAdminStruct.HierarchyResponseDO hierResponse = null;
6
7      // get product and option group
8      Product2 prod = [SELECT Id FROM Product2 WHERE Name = :productName
   LIMIT 1];
9      Apttus_Config2__ClassificationHierarchy__c optionGroup = [SELECT Id
   FROM Apttus_Config2__ClassificationHierarchy__c WHERE Name
   = :optionGroupName LIMIT 1];
10
11     hierRequest.ProductId = prod.Id;
12     hierRequest.OptionGroupId = optionGroup.Id;
13
14     hierResponse =
   Apttus_Config2.CPQAdminWebService.getChildOptions(hierRequest);
15
16     return hierResponse;
17 }

```

## Retrieving Options in an Option Group

You can use this API to retrieve the list of options associated with a given option group.

API	Signature
getChildOptions	<pre> <i>webService static</i> Apttus_Config2.CPQAdminStruct.HierarchyResponseDO getChildOptions(Apttus_Config2.CPQAdminStruct.HierarchyRe questDO hierarchyRequestDO) </pre>

Request Parameter		
Name	Type	Description
hierarchyRequestDO	Apttus_Config2.CPQAdminStruct.HierarchyRequestDO	The category request data object.
Request Parameter - Apttus_Config2.CPQAdminStruct.HierarchyRequestDO		
Name	Type	Description
HierarchyDOs	List<Apttus_Config2.CPQAdminStruct.HierarchyDO>	The list of hierarchy data object

Data Object- Apttus_Config2.CPQAdminStruct.HierarchyDO		
Name	Type	Description
ChildProductId	Id	The Id of child product.
ParentProductId	Id	The Id of the parent product.
Response Data Object- Apttus_Config2.CPQAdminStruct.HierarchyResponseDO		
Name	Type	Description
ProductStructureJSONs	List<Apttus_Config2.CPQAdminStruct.MapDO>	The list of map, where the key is the option group Id and value is the JSON string of the product
Response Data Object - Apttus_Config2.CPQAdminStruct.MapDO		
Field	Type	Description
String	Key	The option group ID
String	Value	The value refers to the JSON String of the product.

## Code Sample

The below sample code returns the list of options for the given product and the option group.

```
1  public Apttus_Config2.CPQAdminStruct.HierarchyResponseDO
   getChildOptions(String productName, String optionGroupName)
2  {
3      Apttus_Config2.CPQAdminStruct.HierarchyDO hierDO = new
   Apttus_Config2.CPQAdminStruct.HierarchyDO();
4      Apttus_Config2.CPQAdminStruct.HierarchyRequestDO hierRequest = new
   Apttus_Config2.CPQAdminStruct.HierarchyRequestDO();
5      Apttus_Config2.CPQAdminStruct.HierarchyResponseDO hierResponse = null;
6
7      // get product and option group
8      Product2 prod = [SELECT Id FROM Product2 WHERE Name = :productName
   LIMIT 1];
9      Apttus_Config2__ClassificationHierarchy__c optionGroup = [SELECT Id
   FROM Apttus_Config2__ClassificationHierarchy__c WHERE Name
   = :optionGroupName LIMIT 1];
10
11     hierRequest.ProductId = prod.Id;
12     hierRequest.OptionGroupId = optionGroup.Id;
13
14     hierResponse =
   Apttus_Config2.CPQAdminWebService.getChildOptions(hierRequest);
15
16     return hierResponse;
17 }
```

## Retrieving of Product Hierarchy

You can use this API to retrieve the product hierarchy for a given product in JSON array.



API		Signature
getProductStructure		<i>webservice static Apttus_Config2.CPQAdminStruct.HierarchyResponseDO getProductStructure(Apttus_Config2.CPQAdminStruct.HierarchyRequestDO hierarchyRequestDO)</i>
Request Parameter		
Name	Type	Description
hierarchyRequestDO	Apttus_Config2.CPQAdminStruct.HierarchyRequestDO	The hierarchy request data object.

Request Data Object - Apttus_Config2.CPQAdminStruct.HierarchyRequestDO		
Name	Type	Description
ProductId	Id	The Id of the product for which you want to retrieve hierarchy.
Response Parameter - Apttus_Config2.CPQAdminStruct.HierarchyResponseDO		
Field	Type	Description
ProductStructureJSONs	List<Apttus_Config2.CPQAdminStruct.MapDO>	The product hierarchy structure in JSON format.
Success	Boolean	Indicates whether building hierarchy was successful or not.
Response Data Object - Apttus_Config2.CPQAdminStruct.MapDO		
Field	Type	Description
String	Key	The Id of the product.

Response Data Object - Apttus_Config2.CPQAdminStruct.MapDO		
Field	Type	Description
String	Value	The value refers to the JSON string of the product tree.

### Code Sample

The below sample code returns the product hierarchy for the given product in JSON array.

```

1  //The below method returns the entire product hierarchy for the given
2  //product in JSON Array
3  public List<Apttus_Config2.CPQAdminStruct.MapDO>
4  getProductStructure(String productName)
5  {
6      Product2 bundleProduct = [SELECT Id, Name
7                                FROM Product2
8                                WHERE Name = :productName
9                                LIMIT 1];
10
11     Apttus_Config2.CPQAdminStruct.HierarchyRequestDO request = new
12     Apttus_Config2.CPQAdminStruct.HierarchyRequestDO();
13     request.ProductIds.add(bundleProduct.Id);
14     Apttus_Config2.CPQAdminStruct.HierarchyResponseDO response =
15     Apttus_Config2.CPQAdminWebService.getProductStructure(request);
16
17     return(response != null && response.ProductStructureJSONs.size() > 0 ?
18     response.ProductStructureJSONs : null);
19 }

```

## Retrieves List of Parent Product

You can use this API to retrieve the list of product Ids that are parents to the given product.

API	Signature
getParentProducts	<i>webservice static List getParentProducts(String productId)</i>

Request Parameter		
Name	Type	Description
productId	String	The id product for which you want to retrieve the parent product.

Response Parameter		
Name	Type	Description
parentProductIDs	List	The list of parent products

## Code Sample

The below sample code retrieves the parent products of a given product.

```

1  public List<Id> getParentProducts(String productName) {
2      // get child and parent products and option group
3      Product2 childProduct = [SELECT Id FROM Product2 WHERE Name
= :productName LIMIT 1];
4
5      // retrieve product IDs
6      List<ID> parentProductIds =
CPQAdminWebService.getParentProducts(childProduct.Id);
7
8      return parentProductIds;
9  }

```

## Retrieves List Product IDs

You can use this API to retrieve the list of product IDs for the given product name.

API		Signature
getProductIds		<i>WebService static List getProductIds(String productName)</i>
Request Parameter		
Name	Type	Description
productName	String	The name product for which you want to retrieve the product IDs.
Response Parameter		
Name	Type	Description
productId	List<String>	The list of product IDs.

### Code Sample

The below sample code retrieves the product IDs of a given product name.

```

1  public List<Id> getParentProducts(String productName) {
2
3      // retrieve product IDs
4      List<ID> productIds =
5      Apttus_Config2.CPQAdminWebService.getProductIds(productName);
6      return productIds;
7  }

```

## Retrieves List of Category IDs

You can use this API to retrieve the list of category IDs that are associated with the given product.

API	Signature
getCategoryIds	<i>webservice static List getCategoryIds(String productId)</i>

Request Parameter		
Name	Type	Description
productId	String	The Id product for which you want to retrieve the category.

Response Parameter		
Name	Type	Description
categoryIds	List	The list of category Ids

### Code Sample

The below sample code retrieves the category associated with the given product.

```

1  public void getCategoryIDs(String productName) {
2      List<ID> categoryIds;
3
4      // retrieve the productS0
5      Product2 productS0 = [SELECT Id, Name
6                          FROM Product2
7                          WHERE Name = :productName
8                          LIMIT 1];
9
10     // retrieve associated category ids

```

```

11     categoryIds =
12     Apttus_Config2.CPQAdminWebService.getCategoryIDs(productS0.Id);
13     retrun ategoryIds;
14 }
    
```

## Creating New Attribute Group

You can use this API to create attribute group for a set of attributes.

API	Signature
createAttributeGroup	<i>webservice static</i> <i>Apttus_Config2.CPQAdminStruct.ProductAttributeResponseD</i> <i>O</i> <i>createAttributeGroup(Apttus_Config2.CPQAdminStruct.Prod</i> <i>uctAttributeRequestDO prodAttrRequestDO)</i>

Request Parameter		
Name	Type	Description
prodAttrRequestDO	Apttus_Config2.CPQAdminStruct.Pr oductAttributeRequestDO	The product attribute request data object.

Request Data Object - Apttus_Config2.CPQAdminStruct.ProductAttributeRequestDO		
Name	Type	Description
ProductAttributeDO s	List<Apttus_Config2.CPQAdmi nStruct.ProductAttributeDO>	The list of product attribute data objects.

Request Data Object - Apttus_Config2.CPQAdminStruct.ProductAttributeRequestDO		
Name	Type	Description
ProductAttributeGroupMemberSOs	List<Apttus_Config2__ProductAttributeGroupMember__C>	The list of attributes for the product attribute group.
Data Object - Apttus_Config2.CPQAdminStruct.ProductAttributeDO		
Name	Type	Description
AttributeGroupName	String	The name of the attribute group you need to associate attributes to.
ProductAttributesToAdd	List <String>	The API name of the attributes you want to add to the attribute group.
ProductAttributesToRemove	List <String>	The API name of the attributes you want to remove from the attribute group.
ProductId	Id	The ID of the product.
Response Parameter - Apttus_Config2.CPQAdminStruct.ProductAttributeResponseDO		
Field	Type	Description
ProductAttributeGroupResponseMap	List<Apttus_Config2.CPQAdminStruct.MapDO>	The list of key-value pairs. The key refers to the attribute group name that you provided and the value contains the record ID.
Success	Boolean	Indicates whether the attribute group was created successfully or not.
Response Data Object - Apttus_Config2.CPQAdminStruct.MapDO		
Field	Type	Description
String	Key	The name of the attribute group.

Response Data Object - Apttus_Config2.CPQAdminStruct.MapDO		
Field	Type	Description
String	Value	The value refers record of the newly created attribute group.

## Code Sample

The sample code below enables you to create an attribute group for the attribute group name and a list of API names of the attributes.

```

1  public Map<Apttus_Config2__ProductAttributeGroup__c>
   createAttributeGroups(String attributeName, List<String>
   attributeList)
2  {
3      Apttus_Config2.CPQAdminStruct.ProductAttributeDO requestAttributeDO =
   new Apttus_Config2.CPQAdminStruct.ProductAttributeDO();
4
5      requestAttributeDO.AttributeGroupName = attributeName;
6      requestAttributeDO.ProductAttributesToAdd = attributeList;
7
8      Apttus_Config2.CPQAdminStruct.ProductAttributeRequestDO requestDO =
   new Apttus_Config2.CPQAdminStruct.ProductAttributeRequestDO();
9      requestDO.ProductAttributeDOs = new
   List<Apttus_Config2.CPQAdminStruct.ProductAttributeDO>
   {requestAttributeDO };
10
11     Apttus_Config2.CPQAdminStruct.ProductAttributeResponseDO response =
   Apttus_Config2.CPQAdminWebService.createAttributeGroup(requestDO);
12
13     List<Apttus_Config2.CPQAdminStruct.MapDO> attributeGroupDetails =
   response.ProductAttributeGroupResponseMap;
14
15     List<Id> attributGroupIds = new List<Id> ();
16     for (Apttus_Config2.CPQAdminStruct.MapDO attributeGroup :
   attributeGroupDetails)
17     {
18         attributGroupIds.add(attributeGroup.Value);

```



```

19     }
20
21     return [Select Id from Apttus_Config2__ProductAttributeGroup__c Where
    Id IN :attributGroupIds];
22 }

```

## Adding and Removing Attributes from an Attribute Group

You can use this API to add or remove attributes from attributes groups.

API	Signature
updateAttributeGroup	<i>webservice static</i> <i>Apttus_Config2.CPQAdminStruct.ProductAttributeResponseDO</i> <i>O</i> <i>updateAttributeGroup(Apttus_Config2.CPQAdminStruct.ProductAttributeRequestDO prodAttrRequestDO)</i>

Request Parameter		
Name	Type	Description
prodAttrRequestDO	Apttus_Config2.CPQAdminStruct.ProductAttributeRequestDO	The product attributes request data object.

Request Data Object - Apttus_Config2.CPQAdminStruct.ProductAttributeRequestDO		
Name	Type	Description
ProductAttributeDOs	List<Apttus_Config2.CPQAdminStruct.ProductAttributeDO>	The list of product attribute data objects.

Request Data Object - Apttus_Config2.CPQAdminStruct.ProductAttributeRequestDO		
Name	Type	Description
ProductAttributeGroupMemberSOs	List<Apttus_Config2__ProductAttributeGroupMember__C>	The list of attributes to be added to or removed from the attribute group.
Data Object - Apttus_Config2.CPQAdminStruct.ProductAttributeDO		
Name	Type	Description
AttributeGroupId	Id	The ID of the attribute group.
AttributeGroupName	String	The name of the attribute group you need to associate attributes to.
ProductAttributesToAdd	List <String>	The API name of the attributes you want to add to the attribute group.
ProductAttributesToRemove	List <String>	The API name of the attributes you want to remove from the attribute group.
ProductId	Id	The ID of the product.
Response Parameter - Apttus_Config2.CPQAdminStruct.ProductAttributeResponseDO		
Field	Type	Description
ProductAttributeGroupResponseMap	List<Apttus_Config2.CPQAdminStruct.MapDO>	The list of key-value pairs. The key refers to the attribute group name that you provided and the value contains the record ID.
Success	Boolean	Indicates whether the attributes were added or removed successfully.

Response Data Object - Apttus_Config2.CPQAdminStruct.MapDO		
Field	Type	Description
String	Key	The name of the attribute group.
String	Value	The value refers record of the attribute group updated.

### Code Sample

The sample code below enables you to add attributes in an attribute group

```

1  public Boolean addAttributesToAttributeGroups(Id attributeGroupId,
2  List<String> attributeList)
3  {
4      Apttus_Config2.CPQAdminStruct.ProductAttributeDO requestAttributeDO =
5      new Apttus_Config2.CPQAdminStruct.ProductAttributeDO();
6
7      requestAttributeDO.AttributeGroupId = attributeGroupId;
8      requestAttributeDO.ProductAttributesToAdd = attributeList;
9
10     Apttus_Config2.CPQAdminStruct.ProductAttributeRequestDO requestDO =
11     new Apttus_Config2.CPQAdminStruct.ProductAttributeRequestDO();
12     requestDO.ProductAttributeDOs = new
13     List<Apttus_Config2.CPQAdminStruct.ProductAttributeDO>
14     {requestAttributeDO };
15
16     Apttus_Config2.CPQAdminStruct.ProductAttributeResponseDO response =
17     Apttus_Config2.CPQAdminWebService.updateAttributeGroup(requestDO);
18
19     return response.Success;
20 }

```

The sample code below enables you to remove attributes from an attribute group

```

1  public Map<Apttus_Config2__ProductAttributeGroup__c>
   removeAttributesFromAttributeGroups(Id attributeGroupId, List<String>
   attributeList)
2  {
3      Apttus_Config2.CPQAdminStruct.ProductAttributeDO requestAttributeDO =
   new Apttus_Config2.CPQAdminStruct.ProductAttributeDO();
4
5      requestAttributeDO.AttributeGroupId = attributeGroupId;
6      requestAttributeDO.ProductAttributesToRemove = attributeList;
7
8      Apttus_Config2.CPQAdminStruct.ProductAttributeRequestDO requestDO =
   new Apttus_Config2.CPQAdminStruct.ProductAttributeRequestDO();
9      requestDO.ProductAttributeDOs = new
   List<Apttus_Config2.CPQAdminStruct.ProductAttributeDO>
10     {requestAttributeDO};
11
12     Apttus_Config2.CPQAdminStruct.ProductAttributeResponseDO response =
   Apttus_Config2.CPQAdminWebService.updateAttributeGroup(requestDO);
13
14     return response.Success;
15 }

```

## Associating Attribute Group and Products

You can use this API to associate attribute group and products.

API	Signature
associateAttributeGroup	<i>webservice static</i> <i>Apttus_Config2.CPQAdminStruct.ProductAttributeResponse</i> <i>DO</i> <i>associateAttributeGroup(Apttus_Config2.CPQAdminStruct.</i> <i>ProductAttributeRequestDO prodAttrRequestDO)</i>

Request Parameter		
Name	Type	Description
prodAttrRequestDO	Apttus_Config2.CPQAdminStruct.ProductAttributeRequestDO	The product attribute request data object.

Request Data Object - Apttus_Config2.CPQAdminStruct.ProductAttributeRequestDO		
Name	Type	Description
ProductAttributeDOs	List<Apttus_Config2.CPQAdminStruct.ProductAttributeDO>	The list of product attribute data objects.
ProductAttributeGroupMemberSOs	List<Apttus_Config2__ProductAttributeGroupMember__C>	The list of attributes for the product attribute group.

Data Object - Apttus_Config2.CPQAdminStruct.ProductAttributeDO		
Name	Type	Description
AttributeGroupName	String	The name of the attribute group you need to associate with the product.
ProductAttributesToAdd	List <String>	The API name of the attributes you want to add to the attribute group.
ProductAttributesToRemove	List <String>	The API name of the attributes you want to remove from the attribute group.
ProductId	Id	The ID of the product.

Response Parameter - Apttus_Config2.CPQAdminStruct.ProductAttributeResponseDO		
Field	Type	Description
ProductAttributeGroupResponseMap	List<Apttus_Config2.CPQAdminStruct.MapDO>	The list of key-value pairs. The key refers to the attribute group name that you provided and the value contains the record ID.
Success	Boolean	Indicates whether the attribute group was associated successfully or not.

Response Data Object - Apttus_Config2.CPQAdminStruct.MapDO		
Field	Type	Description
String	Key	The name of the attribute group.
String	Value	The value refers record of the associated attribute group.

### Code Sample

The sample code below enables you associates products and an attribute group.

```

1  public List<Apttus_Config2__ProductAttributeGroupMember__c>
   associateAttributeGroups(Id attributeGroupId, Id productId)
2  {
3      Apttus_Config2.CPQAdminStruct.ProductAttributeDO requestAttributeDO =
   new Apttus_Config2.CPQAdminStruct.ProductAttributeDO();
4
5      requestAttributeDO.AttributeGroupId = attributeGroupId;
6      requestAttributeDO.ProductId = productId;
7
8      Apttus_Config2.CPQAdminStruct.ProductAttributeRequestDO requestDO =
   new Apttus_Config2.CPQAdminStruct.ProductAttributeRequestDO();
9

```

```

10     requestDO.ProductAttributeDOs = new
        List<Apttus_Config2.CPQAdminStruct.ProductAttributeDO>
        {requestAttributeDO};
11
12     Apttus_Config2.CPQAdminStruct.ProductAttributeResponseDO response =
        Apttus_Config2.CPQAdminWebService.associateAttributeGroup(requestDO);
13     return response.ProductAttributeGroupMembersDOs;
14 }

```

## Dissociating Attribute Group and Products

You can use this API to remove the association attribute group and products.

API	Signature
removeAttributeGroup	<i>webService static</i> <i>Apttus_Config2.CPQAdminStruct.ProductAttributeResponseDO</i> <i>removeAttributeGroup(Apttus_Config2.CPQAdminStruct.ProductAttributeRequestDO prodAttrRequestDO)</i>

Request Parameter		
Name	Type	Description
prodAttrRequestDO	Apttus_Config2.CPQAdminStruct.ProductAttributeRequestDO	The product attribute request data object.

Request Data Object - Apttus_Config2.CPQAdminStruct.ProductAttributeRequestDO		
Name	Type	Description
ProductAttributeDOs	List<Apttus_Config2.CPQAdminStruct.ProductAttributeDO>	The list of product attribute data objects.

Request Data Object - Apttus_Config2.CPQAdminStruct.ProductAttributeRequestDO		
Name	Type	Description
ProductAttributeGroupMemberSOs	List<Apttus_Config2__ProductAttributeGroupMember__C>	The list of attributes for the product attribute group.
Data Object - Apttus_Config2.CPQAdminStruct.ProductAttributeDO		
Name	Type	Description
AttributeGroupName	String	The name of the attribute group you need to dissociate with the product.
ProductAttributesToAdd	List <String>	The API name of the attributes you want to add to the attribute group.
ProductAttributesToRemove	List <String>	The API name of the attributes you want to remove from the attribute group.
ProductId	Id	The ID of the product.
Response Parameter - Apttus_Config2.CPQAdminStruct.ProductAttributeResponseDO		
Field	Type	Description
ProductAttributeGroupResponseMap	List<Apttus_Config2.CPQAdminStruct.MapDO>	The list of key-value pairs. The key refers to the attribute group name that you provided and the value contains the record ID.
Success	Boolean	Indicates whether the attribute group was dissociated successfully or not.



Response Data Object - Apttus_Config2.CPQAdminStruct.MapDO		
Field	Type	Description
String	Key	The name of the attribute group.
String	Value	The value refers record of the dissociated attribute group.

### Code Sample

The sample code below enables you dissociates products and an attribute group.

```

1  public List<Apttus_Config2__ProductAttributeGroupMember__c>
   disassociateAttributeGroups(Id attributeGroupId, Id productId)
2  {
3      Apttus_Config2.CPQAdminStruct.ProductAttributeDO requestAttributeDO =
   new Apttus_Config2.CPQAdminStruct.ProductAttributeDO();
4
5      requestAttributeDO.AttributeGroupId = attributeGroupId;
6      requestAttributeDO.ProductId = productId;
7
8      Apttus_Config2.CPQAdminStruct.ProductAttributeRequestDO requestDO =
   new Apttus_Config2.CPQAdminStruct.ProductAttributeRequestDO();
9
10     requestDO.ProductAttributeDOs = new
   List<Apttus_Config2.CPQAdminStruct.ProductAttributeDO>{requestAttributeDO}
   ;
11
12     Apttus_Config2.CPQAdminStruct.ProductAttributeResponseDO response =
   Apttus_Config2.CPQAdminWebService.removeAttributeGroup(requestDO);
13     return response.ProductAttributeGroupMemberSOs;
14 }

```

## Batch Job Service

The Batch Job Service APIs account for the standard actions to configure, price, and quote.

You can invoke APIs in Batch Job Service using the following command:

```
Apttus_Config2.BatchJobService.<Name of the Function>
where the name of the function is API Name and it's parameters.
```

You can use the Batch Job Service APIs to complete the following tasks:

- [Splitting a Smart Cart](#)
- [Updating Price for the Smart Cart](#)
- [Finalizing Configuration in Enterprise and Smart Cart](#)
- [Retrieving Apex Batch Job Information](#)

## Splitting a Smart Cart

This global method splits the line items in the cart into groups in a Smart Cart flow.

API	Signature
splitCart	<i>static Id splitCart(Id cartId)</i>

Request Parameter			
Name	Type	Required?	Description
cartID	ID	Yes	The Id of the cart to be split.

Response Parameter		
Field	Type	Description
jobId	ID	The Id of the job executed to split the cart.

### Code Sample

This sample enables you to split the line items in the cart into groups in a Smart Cart flow. When the method is executed, the task pending flag is set to true on the cart passed as the parameter. After the line items are split into groups, the flag is set to false.

```

1 //Api to split large cart (pass in either Main Cart Id or Config Cart Id).
  This would set the task pending flag on the cart passed as the parameter
  (either main or config). When the split is completed, the flag is cleared.
  Note that it will call the pricing once split is done.
2
3 ID jobId = Apttus_Config2.BatchJobService.splitCart(mainOrConfigCartId);

```

## Updating Price for the Smart Cart

This global method updates the prices of the line items after the cart is split.

API		Signature	
updatePriceForCart		<i>static Id updatePriceForCart(Id cartId)</i>	
Request Parameter			
Name	Type	Required?	Description
cartID	ID	Yes	The Id of the cart for which you want to update prices of the line item

Response Parameter		
Field	Type	Description
jobId	ID	The Id of the batch job executed to update the pricing of the cart

### Code Sample

This sample enables you to update the price of the line items after the cart is split. When the method is executed, the task pending flag is set to true on the cart passed as the parameter. After pricing is calculated for the line items, the flag is set to false.

```

1 //Api to price large cart (pass in either Main Cart Id or Config Cart Id),
  //if there is a change in any pricing parameter post split.
2 This would set the task pending flag on the cart passed as the parameter
  (either main or config). When all carts are priced, the flag is cleared.
3
4 ID jobId =
  Apttus_Config2.BatchJobService.updatePriceForCart(mainOrConfigCartId);

```

## Finalizing Configuration in Enterprise and Smart Cart

This global method finalizes a cart and executes post synchronization tasks in Enterprise and Smart Cart flow.

API		Signature	
finalizeCart		<i>static Id finalizeCart(Id cartId)</i>	
Request Parameter			
Name	Type	Required?	Description
cartId	ID	Yes	The Id of the cart for which you want to finalize
Response Parameter			
Field	Type	Description	
jobId	ID	The Id of the batch job executed to finalize the cart	

## Code Sample

This sample enables you to finalize a cart in Smart Cart flow after the cart is split & updating price is complete. When the method is executed, the task pending flag is set to true on the cart passed as the parameter. After finalization is completed, the status of product configurations is updated to *Finalized*.

```

1  /**
2  *Finalizes and executes post synchronization tasks for the given cart
3  *@param cartId the cart id
4  *@return the batch job id
5  */
6  global static ID finalizeCart (ID cartId)
7  {
8      ID jobId =
9      Apttus_Config2.BatchJobService.finalizeCart(mainOrConfigCartId);
10 }
```

## Retrieving Apex Batch Job Information

This global method retrieved the information of the asynchronous apex job running in the backend for a business object.

You can retrieve current information of the following apex jobs:

- Create and activate Agreement
- Clone proposal and agreement
- Finalize proposal and agreement
- Accept proposal
- Create and accept Order
- Asynchronous cart pricing

API	Signature
<code>getBatchJobInfoForContext</code>	<code>static Apttus_Config2.CPQStruct.JobInfo getBatchJobInfoForContext(Id ctxObjectId)</code>

Request Parameter			
Name	Type	Required?	Description
ctxObjectId	ID	Yes	The Id of the business object like proposal or agreement.
Response Data Object - Apttus_Config2.CPQStruct.JobInfo			
Field	Type	Description	
ChildJobs	List	The child apex jobs that are associated the original apex jobs.	
CompletedDate	Datetime	The date and time when the original or child of the apex job was completed.	
ExtendedStatus	String	The messages of errors encountered during apex job runtime.	
JobId	Id	The apex job Id	
JobName	String	The functional and apex job name in the following format: <Function name>:<apex job name>	
JobType	String	The type of the apex job. For example, queueable and batchable.	
ParentJobId	Id	The Id of the given apex job's parent	
PercentComplete	Integer	The percentage of completion of the apex job at the time when global method was invoked.	

Response Data Object - Apttus_Config2.CPQStruct.JobInfo		
Field	Type	Description
Status	String	The status of the parent apex job. The following are the possible values: <ul style="list-style-type: none"> <li>• Processing</li> <li>• Completed</li> <li>• Failed</li> <li>• Aborted</li> </ul>
SubmittedDate	Datetime	The date and time when the apex job was submitted.

### Code Sample

The above global method can be used in API polling in a specific interval to retrieve the current status of the apex jobs running in the backend. For example, you are building a custom Lightning Web Component to show the live progress bar with the percentage of completion and status of the job in the proposal object. The Below apex controller calls the global method and returns the percentage of completion and the status to the UI controller to handle.

```

1  public static final String STATUS_FAILED = 'Failed';
2  public static final String STATUS_API_CALLOUT_ERROR = 'APIError';
3      public static final String GENERIC_JOB_NAME = 'asynchronous';
4          public static Boolean isLightning = null;
5          public static Boolean isDeployedInPackage = null;
6  public ProposalBannerController(ApexPages.StandardController
stdController) {
7      }
8      /**
9       * Gets the async job info for the proposal
10     * @param proposalID id of the quote/proposal record
11     * @return map of status, percentcomplete and jobId to the LWC.
12     */
13     @AuraEnabled
14     public static Map<String,Object> getBatchJobInfo(String proposalID){
15         Map<String,Object> responseDO = new Map<String,Object>();
16         String functionalJobName = GENERIC_JOB_NAME;
17         String asyncClassName = '';

```

```

18     try {
19         Apttus_Config2.CPQStruct.JobInfo parentJobInfo =
Apttus_CPQAPI.BatchUpdateService.getBatchJobInfoForContext (proposalID);
20         responseD0.put('parentJobStatus', parentJobInfo.Status);
21         responseD0.put('parentPercentComplete',
parentJobInfo.PercentComplete);
22         responseD0.put('parentJobId', parentJobInfo.JobId);
23
24         if (parentJobInfo.JobName != null) {
25             functionalJobName = parentJobInfo.JobName.split(':')[0];
26             asyncClassName = parentJobInfo.JobName.split(':')[1] + ':';
27         }
28         responseD0.put('jobName', functionalJobName);
29
30         if(parentJobInfo.status == STATUS_FAILED){
31             responseD0.put('ExtendedStatus', asyncClassName +
parentJobInfo.ExtendedStatus);
32
33         }
34     } catch (Exception e) {
35         // We have to ignore this error in LWC
36         responseD0.put('ExtendedStatus', e.getMessage());
37         responseD0.put('parentJobStatus', STATUS_API_CALLOUT_ERROR);
38         responseD0.put('jobName', GENERIC_JOB_NAME);
39     }
40
41     return responseD0;
42 }
43 }

```

## Constraint Web Service 2

The Constraint web service APIs account for the actions to constraint rules.

You can invoke APIs in Constraint Web Service 2 from the following command:

```
Apttus_Config2.ConstraintWebService2.<Name of the Function>
where the name of the function is API Name and it's parameters.
```

You can use the constraint web service APIs to complete the following tasks:



- [Associating Constraint Rule to Products Added to the Cart](#)
- [Applying Constraint Rule to All Products Added to the Cart](#)
- [Applying Constraint Rule to Bundles Added to the Cart](#)
- [Retrieving Products Auto-included to the Cart](#)

## Associating Constraint Rule to Products Added to the Cart

You can use this API to execute constraint rules on a list of products.

API	Signature
associateConstraintRulesForProducts	<i>WebService static Boolean associateConstraintRulesForProducts(Id cartId, List productIds)</i>

Parameters		
Name	Type	Description
cartId	Id	The Id of the cart you added products to.
productIds	List<Id>	The list of Ids of the products you added to the cart.

Response Parameter		
Field	Type	Description
IsSuccess	Boolean	Indicates whether the association was successful.

### Code Sample

The sample below enables you to executes a constraint rule on a list of line items in the given cart.

```

1  public Boolean associateConstraintRules(String proposalID)
2  {
3
4      Apttus_Config2__ProductConfiguration__c cart = [SELECT
Apttus_Config2__PriceListId__c, Id
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
FROM
Apttus_Config2__ProductConfiguration__c
WHERE
Apttus_QPConfig__ProposalId__c =: proposalID
LIMIT 1];

//Fetch All products from Product Configuration
List<Apttus_Config2__LineItem__c> liSOList = [SELECT
Apttus_Config2__ProductID__c
FROM
Apttus_Config2__LineItem__c
WHERE
Apttus_Config2__ConfigurationId__c = :cart.Id];
List<Id> productIds = new List<Id> ();
for (Apttus_Config2__LineItem__c liSO: liSOList)
{
    productIds.add(liSO.Apttus_Config2__ProductID__c);
}
Boolean isSuccess =
Apttus_Config2.ConstraintWebService2.associateConstraintRulesForProducts(c
art.Id, productIds);
return isSuccess;
}

```

## Applying Constraint Rule to All Products Added to the Cart

You use this API to applying constraint rules to all products you added to the cart.

API	Signature
applyConstraintRulesOnAddAll	<i>webservice static Apttus_Config2.ConstraintWebService2.ConstraintResult 2 applyConstraintRulesOnAddAll(Id cartId)</i>

Parameters		
Name	Type	Description
cartId	Id	The Id of the cart you added products to.

Response Parameter - Apttus_Config2.ConstraintWebService2.ConstraintResult2		
Field	Type	Description
IsSuccess	Boolean	Indicates whether the association was successful.

### Code Sample

The sample below enabled you to associate a constraint rule to all the line items in the given cart.

1	<b>public</b> Boolean associateConstraintRulesOnCart(String proposalID)
2	{
3	
4	Apttus_Config2__ProductConfiguration__c cart = [SELECT
5	Apttus_Config2__PriceListId__c, Id
6	FROM
	Apttus_Config2__ProductConfiguration__c
	WHERE
	Apttus_QPConfig__ProposalId__c =: proposalID

```

7      Apttus_Config2.ConstraintWebService2.ConstraintResult2 response =
      Apttus_Config2.ConstraintWebService2.applyConstraintRulesOnAddAll(cart.Id)
      ;
8      return response.isSuccess;
9  }
    
```

## Applying Constraint Rule to Bundles Added to the Cart

You use this API to apply constraint rules to the bundles you added to the cart.

API	Signature
applyConstraintRulesOnAddBundle	<i>webService static Apttus_Config2.ConstraintWebService2.ConstraintResult2 applyConstraintRulesOnAddBundle(Id cartId, Decimal lineNumber)</i>

Parameters		
Name	Type	Description
cartId	Id	The Id of the cart you added products to.
lineNumber	Decimal	The line number of the bundle added to the cart

Response Parameter - Apttus_Config2.ConstraintWebService2.ConstraintResult2		
Field	Type	Description
IsSuccess	Boolean	Indicates whether the CPQ successfully applied the constraint rule

## Code Sample

The sample below enables you to apply a constraint rule to the bundle line item in the given cart.

```

1  public Boolean associateConstraintRulesOnAddBundle(String proposalID,
2  String productId)
3  {
4      Apttus_Config2__ProductConfiguration__c cart = [SELECT
5  Apttus_Config2__PriceListId__c, Id
6  FROM
7  Apttus_Config2__ProductConfiguration__c
8  WHERE
9  Apttus_QPConfig__ProposalId__c =: proposalID ]
10
11     //Fetch All products from Product Configuration
12     Apttus_Config2__LineItem__c lineItemSO = [SELECT
13  Apttus_Config2__ProductID__c, Apttus_Config2__LineNumber__c
14  FROM
15  Apttus_Config2__LineItem__c
16  WHERE
17  Apttus_Config2__ConfigurationId__c = :cart.Id
18  AND
19  Apttus_Config2__ProductID__c =: productId limit 1];
20
21     Apttus_Config2.ConstraintWebService2.ConstraintResult2 response =
22     Apttus_Config2.ConstraintWebService2.applyConstraintRulesOnAddBundle(cart.
23     Id, lineItemSO.Apttus_Config2__LineNumber__c);
24     return response.isSuccess;
25 }

```

## Retrieving Products Auto-included to the Cart

You use this API to retrieve IDs of the products added to the cart by the auto-inclusion constraint rule.

API	Signature
getAutoIncludedProductIdsForCart	<i>WebService static List&lt;&gt;</i> <i>getAutoIncludedProductIdsForCart(Id cartId)</i>

Parameters		
Name	Type	Description
cartId	Id	The Id of the cart you added products to.

Response Parameter		
Field	Type	Description
response	List<list<ID>>	The IDs of products auto-included to the cart

### Code Sample

The sample below retrieves the ID of the products auto-included to cart by the auto-inclusion type constraint rules. The list of ID list is returned after the execution of this API. The ID list contains the ID of the condition products and the list of the ID of associated action products.

```

1  public Map<Id, Set<Id>> getAutoIncludedProductsForCart(String proposalID)
2  {
3
4      Apttus_Config2__ProductConfiguration__c cart = [SELECT
5      Apttus_Config2__PriceListId__c, Id
6
7
8
9
10     FROM
11     Apttus_Config2__ProductConfiguration__c
12
13     WHERE
14     Apttus_QPConfig__ProposalId__c =: proposalID
15     ]
16
17     Map<Id, Set<Id>> result = new Map<Id, Set<Id>>();
18     result.put(cart.Id, new Set<Id>());
19     return result;
20 }

```

```

7
8     List<list<id>> response = Apttus_Config2.ConstraintWebService2.
getAutoIncludedProductIdsForCart (cart.Id);
9
10    Map<Id, Set<Id>> autoIncludedIdsByTriggeringProductId = new Map<Id,
Set<Id>> ();
11    for (list<id> productIds : response)
12    {
13
14        for (integer index=1; index <productIds.size() ; index ++)
15        {
16            if
17            (autoIncludedIdsByTriggeringProductId.containsKey(productIds[0]))
18            {
19                autoIncludedIdsByTriggeringProductId.get(productIds[0]).ad
d(productIds[index]);
20            }
21            else
22            {
23                autoIncludedIdsByTriggeringProductId.put(productIds[0],
new set<Id> {productIds[index]});
24            }
25        }
26    return autoIncludedIdsByTriggeringProductId;
27 }

```

## Quote Collaboration Service

The Quote Collaboration Service global methods account for the standard action to Quote Collaboration feature.

Global methods are not like APIs. They can only be invoked in an Apex Code.

You can invoke the method in the Quote Collaboration service using the following command:

```
QuoteCollaborationService.<Name of the Function>
where the name of the function is API Name and it's parameters.
```

In this Section:

- [Adding Products to a Collaboration Request](#)

## Adding Products to a Collaboration Request

This method adds products to an existing collaboration request. This global method is not an API. It can only be invoked in an Apex Code.

Method	Signature		
<b>addProductsToCollaboration</b>	<i>static Apttus_Config2.CollabStruct.AddProductCRResponseDO                      addProductsToCollaboration(Apttus_Config2.CollabStruct.AddProductCRRestDO requestDO)</i>		
Parameters			
Name	Type	Required ?	Description
request	Apttus_Config2.CollabStruct.AddProductCRRestDO	Yes	The request data object.
Request Data Object - Apttus_Config2.CollabStruct.AddProductCRRestDO			
Field	Type	Description	
collaborationRequestId	ID	The id of the collaboration request to which you want add products.	
lineItemIds	List	The list of line item IDs you want to add to the collaboration request	
Response Data Object - Apttus_Config2.CollabStruct.AddProductCRResponseDO			
Field	Type	Description	
errorMessages	List	The list of errors occurred while adding products to the collaboration request.	
hasErrors	Boolean	Indicates that the errors occurred while adding products to the collaboration request.	



Response Data Object - Apttus_Config2.CollabStruct.AddProductCRResponseDO		
Field	Type	Description
isSuccess	Boolean	Indicated that the products are successfully added to the collaboration request.

### Code Sample

The following sample enables you to add products to an existing collaboration request with a Collaboration ID. Provide a list of line item IDs of the products you want to add to the collaboration request. If the products are successfully added to the collaboration request, the API returns true, otherwise, the API returns the list of errors occurred while adding the products.

```

1  /**
2   * The below method demonstrates how to add a product in an existing
   collaboration request
3   */
4   CollabStruct.AddProductCRRequestDO request = new
   CollabStruct.AddProductCRRequestDO();
5   request.collaborationRequestId = 'a3B1S0000005V3wUAE';
6   request.lineItemIds = new List<Id> {'a0a1S0000005YwWj'};
7   CollabStruct.AddProductCRResponseDO response =
   QuoteCollaborationService.addProductsToCollaboration(request);
8   if (!response.isSuccess)
9   {
10      system.debug(response.errorMessages);
11  }
```

## Favorite Configuration Global Service

The Favorite Configuration Global Service API account for the standard actions on the configuration saved as a favorite.

You can invoke APIs in Favorite Configuration Global Service using the following command:

```
Apttus_Config2.FavoriteConfigurationGlobalService.<Name of the Function>
where the name of the function is API Name and it's parameters.
```

You can use the APIs to complete the following tasks:

- [Creating Favorite Configuration](#)

## Creating Favorite Configuration

This API creates a favorite configuration from the Catalog page referenced by cartID.

API	Signature
saveFavoriteConfiguration	<i>static Apttus_Config2.FavoriteConfigurationStruct.SaveFavoriteResponseDO saveFavoriteConfiguration(Apttus_Config2.FavoriteConfigurationStruct.SaveFavoriteRequestDO requestDO)</i>

Parameters		
Name	Type	Description
request	Apttus_Config2.FavoriteConfigurationStruct.SaveFavoriteRequestDO	The request data object.

Request Data Object - Apttus_Config2.FavoriteConfigurationStruct.SaveFavoriteRequestDO		
Field	Type	Description
cartID	ID	The identifier of the cart.
favoriteConfigurationSO	Apttus_Config2__FavoriteConfiguration__c	Reference to the Favorite Configuration Object

Request Data Object - Apttus_Config2.FavoriteConfigurationStruct.SaveFavoriteRequestDO		
Field	Type	Description
lineNumbers	List <Integer>	The list of line numbers to be added to the favorite configuration.

Response Data Object - Apttus_Config2.FavoriteConfigurationStruct.SaveFavoriteResponseDO		
Field	Type	Description
favoriteConfigurationSO	Apttus_Config2__FavoriteConfiguration__c	Reference to the Favorite Configuration Object
processedLineNumbers	List<Integer>	List of line numbers added as a favorite configuration.

Data Object - Apttus_Config2__FavoriteConfiguration_c		
Field	Type	Description
Name	String	Enter a suitable name for your favorite configuration. This name is displayed as a record under your Favorites category.
PriceListId	String	The pricelist id associated with a favorite configuration.

Data Object - Apttus_Config2__FavoriteConfiguration_c		
Field	Type	Description
Scope	String	Choose an option for the visibility of your configuration record. The available values are, <i>Public</i> and <i>Private</i> . If you set this to <i>Private</i> , this record is visible to only the owner of the record. If you set this to <i>Public</i> , this record is accessible to all the users.

### Code Sample

The sample below enables you to create a favorite configuration for a cart with a cartId. You can invoke this API in use cases when you want to show a Favorite configuration page based on the cartID and pricelist.

```

public class QAddToFavorites implements Queueable {
    public Apttus_Config2.FavoriteConfigurationStruct.SaveFavoriteRequestDO
requestDO;

    public
QAddToFavorites(Apttus_Config2.FavoriteConfigurationStruct.SaveFavoriteRequestDO
requestDO)
    {
        this.requestDO = requestDO;
    }

    public void execute(QueueableContext context)
    {
        Apttus_Config2.FavoriteConfigurationStruct.SaveFavoriteResponseDO responseDO
= new Apttus_Config2.FavoriteConfigurationStruct.SaveFavoriteResponseDO();
        responseDO =
Apttus_Config2.FavoriteConfigurationGlobalService.saveFavoriteConfiguration(requestDO
);
        List<Integer> processedLineNumbers = responseDO.processedLineNumbers;
        List<Integer> unProcessedLineNumbers = new List<Integer>();

        for (Integer lineNumber : requestDO.lineNumbers)
        {

```

```

        if (!processedLineNumbers.contains(lineNumber))
        {
            unProcessedLineNumbers.add(lineNumber);
        }
    }

    if (!unProcessedLineNumbers.isEmpty())
    {
        requestDO.favoriteConfigurationS0.id =
responseDO.favoriteConfigurationS0.id ;
        requestDO.lineNumbers = unProcessedLineNumbers;
        system.debug('Request DO is ' + requestDO);
        System.enqueueJob(new QAddToFavorites(requestDO));
    }
}
}

//Sample code to execute from the anonymous window.

Apttus_Config2.FavoriteConfigurationStruct.SaveFavoriteRequestDO requestDO = new
Apttus_Config2.FavoriteConfigurationStruct.SaveFavoriteRequestDO();
requestDO.favoriteConfigurationS0.Name = 'QueueableAPI-1 Fav BundleandStd'; // Name
of favorite configuration
requestDO.favoriteConfigurationS0.Apttus_Config2__Active__c = true;
requestDO.favoriteConfigurationS0.Apttus_Config2__Scope__c = 'Public';
requestDO.favoriteConfigurationS0.Apttus_Config2__PriceListId__c = 'a4QP000000wNYC';
// priceList ID
requestDO.favoriteConfigurationS0.Apttus_Config2__Description__c = 'Sample
description';
requestDO.cartId = ID.valueOf('a4jP000000M23t'); // product configuration or cartID
requestDO.lineNumbers = new List<Integer>{1,2,3,4,5,6,7,8,9,10};
System.enqueueJob(new QAddToFavorites(requestDO));

```

## Quote/Proposal Config Web Service

The Quote/Proposal Config web service API account for the standard actions to configure, price, and quote.

You can invoke APIs in Quote/Proposal Web Service from the following command:

```

Apttus_QPConfig.QPConfigWebService.<Name of the Function>
where the name of the function is API Name and it parameters.

```

You can use the Quote/Proposal Config web service API to complete the following tasks:

- [Accepting a Quote](#)
- [Synchronizing Cart](#)
- [Copying Product Configuration to the Proposal](#)
- [Synchronizing Quote/Proposal with Opportunity](#)

## Accepting a Quote

This API accepts a quote or proposal.

API		Signature	
acceptQuote		<i>webService static Boolean acceptQuote(Id quoteOrProposalId)</i>	
Request Parameter			
Name	Type	Required?	Description
QuoteId	ID	Yes	The id of the quote/proposal you want to accept.
Response Parameter			
Field	Type	Description	
isSuccess	Boolean	Indicates whether the quote is accepted or not. The ID of the newly created cart object	

### Code Sample

The following sample enables you to accept a quote with a Quote ID. Provide a Quote ID of the Quote or Proposal you want to accept. If the quote is successfully accepted, the API returns true, otherwise, the API returns false.

```

1  /**
2  * Accepts the given quote/proposal

```

```

3      * @param quoteOrProposalId the id of the quote/proposal subject to
      accept
4      * @return <code>true</code> if the operation was successful,
      <code>false</code> otherwise
5
6      */
7
8      Webservice static Boolean acceptQuote(ID quoteOrProposalId) {
9          Id quoteOrProposalId = 'a0Y3C000000u2xv';
10         Boolean isSuccess =
      Apttus_QPConfig.QPConfigWebService.acceptQuote(quoteOrProposalId);
11     }

```

## Integration Details

Use the following information in your integrations with CPQ Web Services API. Refer to [Integrating Conga with External Systems](#) for information on how to get started.

## Response/Request XML

### Example Request

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:qpc="http://soap.sforce.com/schemas/class/Apttus_QPConfig/
QPConfigWebService">
  <soapenv:Header>
    <qpc:SessionHeader>
      <qpc:sessionId>00DZ000000NAEIA!
ASAAQNJKTI9h4.FgZutgi4zWkwHRIvIJFNU3_Qh4XPyi0Dbj2Q3XzSjmRb76.sgyY.e0utBVPu5wFakw8WmXH
CIs0jzI375z</qpc:sessionId>
    </qpc:SessionHeader>
  </soapenv:Header>
  <soapenv:Body>
    <qpc:acceptQuote>
      <qpc:quoteOrProposalId>a0eZ0000005pF4h</qpc:quoteOrProposalId>
    </qpc:acceptQuote>
  </soapenv:Body>
</soapenv:Envelope>

```

```

Example Response

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns="http://soap.sforce.com/schemas/class/Apttus_QPConfig/QPConfigWebService">
  <soapenv:Body>
    <acceptQuoteResponse>
      <result>true</result>
    </acceptQuoteResponse>
  </soapenv:Body>
</soapenv:Envelope>
    
```

## Synchronizing Cart

This API synchronizes the given product configuration

API		Signature	
syncConfiguration		<i>webservice static Boolean syncConfiguration(Id configId)</i>	
Request Parameter			
Name	Type	Required?	Description
configId	ID	Yes	The id of the product configuration to synchronize.
Response Parameter			
Field	Type	Description	
isSuccess	Boolean	Indicates whether the synchronization is successful or not.	

### Code Sample

The sample enables you to synchronize the products (line items) added to the cart with the quote or proposal used to generate the cart.



```

1  /**
2   * Below method demonstrate to finalize the cart by passing the
   configuration request Id and execute.
3  */
4
5  public void syncConfigurationAndSync (Id configId)
6  {
7     // finalize the cart
8     Apttus_QPConfig.QPConfigWebService. syncConfiguration(configId);
9  }

```

## Copying Product Configuration to the Proposal

You can use this API to copy product configuration from a saved or finalized proposal to another proposal.

API		Signature	
copyProductConfiguration		<i>webService static Boolean copyProductConfiguration(Id oldProposalId, Id newProposalId)</i>	
Request Parameter			
Name	Type	Required?	Description
newProposalId	ID	Yes	The ID of the proposal from which you want to copy the product configuration.
oldProposalId	ID	Yes	The ID of the proposal where you want to copy the product configuration.
Response Parameter			
Field	Type	Description	
isSuccess	Boolean	Indicates whether the product configuration was copied or not.	

## Code Sample

The sample enables you to copy product configuration from one proposal to another.

```


1  /**
2  * The below method demonstrates how to copy product configuration from one
3  * proposal to another one.
4  */
5  public static Boolean copyProductConfiguration(Id oldProposalID, Id
6  newProposalId )
7  {
8  Boolean isSuccess = false;
9  isSuccess =
10 Apttus_qpconfig.QPConfigWebService.copyProductConfiguration(oldProposalID,
11 newProposalId); return isSuccess ;
12 }


```

## Synchronizing Quote/Proposal with Opportunity

This API synchronizes the quote/proposal line item with a given opportunity by creating opportunity line items, summary items and other related data.

- **Synchronize only line items:** If any one of below first two parameters is *true* or the last setting is *false*, the API copies the proposal line items and creates new opportunity line items. It also copies related line items when the opportunity-related item collection lookup is defined.
  - *inclOptions* value passed to the API
  - **Sync Bundle Using Line Items** in Proposal System Properties
  - Enable **Proposal Summary** in Proposal System Properties
- **Synchronize line items and summary items:** If the above condition is false, this API creates both opportunity line items and summary items. It also copies related line items when the opportunity-related item collection lookup is defined.

 If the external users do not have the required license to the conga packages, they can not use this API.

 The operation will be aborted when

- The user has no permission to access the API
- There is no opportunity or price book associated with the quote or proposal

API		Signature	
syncWithOpportunity		<i>webservice static Boolean syncWithOpportunity(Id quoteOrProposalId, Boolean inclOptions)</i>	
Request Parameter			
Name	Type	Required?	Description
QuoteID	ID	Yes	The ID of the quote/proposal you want to synchronize.
inclOptions	Boolean	Yes	Options inclusion indicator.
Response Parameter			
Field	Type	Description	
isSuccess	Boolean	Indicates whether the synchronization is successful or not.	

## Code Sample

The sample enables you to synchronize quote/proposal line item with a given opportunity.

```

1  /**
2   * The below code demonstrates how to call synchWithOpportnity API form
   Apex code and synchronize only line items
3   */
4   public Boolean syncWithOpportunity (ID quoteProposalID, Boolean
   inclOptions) {
5   Boolean synchOperationFlag = false;
6   synchOperationFlag =
   Apttus_QPConfig.QPConfigWebService.syncWithOpportunity (quoteProposalID,
   inclOptions);
7   return synchOperationFlag;
8   }

```

# Asset Service

The Asset Service method accounts for the standard action to Asset-based Operation.

You can invoke methods in Asset Service using the following command:

```
Apttus_Config2.AssetService.<Name of the Function>
where the name of the function is API Name and it's parameters.
```

Global methods are not APIs. They can only be invoked in an Apex Code.

In this Section:

- [Fetching Asset Line Items](#)
- [Getting a count of Asset Line Items](#)
- [Swapping Assets](#)
- [Getting a list of Products to be swapped with Assets](#)
- [Terminating Assets](#)
- [Renewing Assets](#)
- [Incrementing Assets](#)
- [Changing Assets](#)
- [Merging Duplicate Assets](#)
- [Suspending Assets](#)
- [Resuming Assets](#)

## Fetching Asset Line Items

This global method fetches all Asset Line Items for various Accounts.

API	Signature
<b>getAssetLineItems</b>	<i>static Apttus_Config2.CPQStruct.QueryAssetsResponseDO getAssetLineItems(Apttus_Config2.CPQStruct.QueryAssetsRequestDO request)</i>

Parameters		
Name	Type	Description
request	Apttus_Config2.CPQStruct.QueryAssetsRequestDO	Request object invoked by the method
Request Data Object - Apttus_Config2.CPQStruct.QueryAssetsRequestDO		
Field	Type	Description
AccountIds	List <Id>	List of Account Ids
CustomFilter	String	SOQL condition expression used as a custom filter. For example, user could request only assets after a given billing end date.  This is optional.
Descending	Boolean	This indicates whether the assets are sorted in descending order
FieldNames	List	This can be used to include fields you have added to the Asset Line Item object.
Nrecord	Integer	This can be used to limit the number of records returned
Offset	Integer	This can be used to include an offset
SortFields	List	This can be used sort the result based on a list of fields.
Response Data Object - Apttus_Config2.CPQStruct.QueryAssetsResponseDO		
Field	Type	Description
AssetCount	Integer	The number of assets returned
AssetLineItems	List	List of Asset Line Items.

## Code Sample

The code sample below helps you get Asset Line Items for a list of Accounts. Use this global method to fetch all asset line items associated with an account.

```
// create list of account ids
List<ID> listAccount = new List<ID>();
listAccount.add(accountId);

// create and populate request object
Apttus_Config2.CPQStruct.QueryAssetsRequestDO request = new
  Apttus_Config2.CPQStruct.QueryAssetsRequestDO();
request.AccountIds = listAccount;

// retrieve all fields in Asset Line Item sObject
request.FieldNames = null;

// sort by billing end date (optinal)
request.SortFields = new List<String>{'Apttus_Config2__BillingEndDate__c'};
request.Descending = false; // sort ascending
request.CustomFilter = 'Apttus_Config2__BillingEndDate__c > 2011-04-30'; // optional

// do not offset and do not limit number of records returned
request.Offset = null;
request.Nrecord = null;

// call getAssetLineItems API
Apttus_Config2.CPQStruct.QueryAssetsResponseDO response =
  Apttus_Config2.AssetService.getAssetLineItems(request);

List<Apttus_Config2__AssetLineItem__c> listItems = response.AssetLineItems;
```

## Getting a count of Asset Line Items

This global method gets the count of Asset Line Item in one or more accounts.

API	Signature
countAssetLineItems	<i>static Apttus_Config2.CPQStruct.QueryAssetsResponseDO countAssetLineItems(Apttus_Config2.CPQStruct.QueryAssetsRequestDO request)</i>

Parameters		
Name	Type	Description
request	Apttus_Config2.CPQStruct.QueryAssetsRequestDO	Request object invoked by the method
Request Data Object - Apttus_Config2.CPQStruct.QueryAssetsRequestDO		
Field	Type	Description
AccountIds	List <Id>	List of Account Ids
CustomFilter	String	SOQL condition expression used as a custom filter. For example, user could request only assets after a given billing end date.  This is optional.
Descending	Boolean	This indicates whether the assets are sorted in descending order
FieldNames	List	This can be used to include fields you have added to the Asset Line Item object.
Nrecord	Integer	This can be used to limit the number of records returned
Offset	Integer	This can be used to include an offset
SortFields	List	This can be used sort the result based on a list of fields.
Response Data Object - Apttus_Config2.CPQStruct.QueryAssetsResponseDO		
Field	Type	Description
AssetCount	Integer	The number of assets returned
AssetLineItems	List	List of Asset Line Items.

## Code Sample

The code sample below helps you get a count of Asset Line Items for a particular account. Use this global method to fetch all asset line items associated with an account.

```
// create list of account ids
List<ID> listAccount = new List<ID>();
listAccount.add(accountId);

// create and populate request object
Apttus_Config2.CPQStruct.QueryAssetsRequestDO request = new
  Apttus_Config2.CPQStruct.QueryAssetsRequestDO();
request.AccountIds = listAccount;
request.CustomFilter = 'Apttus_Config2__BillingEndDate__c > 2019-04-30'; // optional

// call countAssetLineItems API
Apttus_Config2.CPQStruct.QueryAssetsResponseDO response =
  Apttus_Config2.AssetService.countAssetLineItems(request);

ApexPages.addMessage(new ApexPages.Message(ApexPages.severity.info, 'Asset Line Item
Count: ' + response.AssetCount));
```

## Swapping Assets

You can invoke this API for swapping Assets.

API	Signature	
<b>swapAssets</b>	<i>static Apttus_Config2.CPQStruct.SwapAssetsResponseDO swapAssets(Apttus_Config2.CPQStruct.SwapAssetsRequestDO request)</i>	
Parameters		
Name	Type	Description
request	Apttus_Config2.CPQStruct.SwapAssetsRequestDO	Request object invoked by the method



Request Data Object - Apttus_Config2.CPQStruct.SwapAssetsRequestDO		
Name	Type	Description
NewStartDate	Date	The new Asset Start Date on Swapping
ProductIds	List<ID>	Id of the product with which the Asset is being swapped.
AssetIds	List<ID>	The Id of the Asset which is swapped.
CartIds	Id	The Id of Cart for which an Asset is being swapped

Response Data Object - Apttus_Config2.CPQStruct.SwapAssetsResponseDO		
Field	Type	Description
LineItemMap	Map<ID, Apttus_Config2__LineItem__c>	Map of line items of the assets that are being swapped.

### Code Sample

The code sample below helps you swap an Asset with another product.

```
// create list of asset ids
List<ID> listAssetId = new List<ID>();
for (AssetLineItemWrapperClass record : wrapperAssetLineItemList)
{
    if (record.selected)
    {
        listAssetId.add(record.assetId);
    }
}

// create list of product ids
List<ID> listProductId = new List<ID>();
for (ProductWrapperClass product : wrapperProductList)
{
    if (product.selected)
    {
```

```

        listProductId.add(product.productId);
    }
}

// create and populate request object
if (objAssetLineItem.Apttus_Config2__StartDate__c != null)
{
    Apttus_Config2.CPQStruct.SwapAssetsRequestDO request = new
Apttus_Config2.CPQStruct.SwapAssetsRequestDO();
    request.AssetIds = listAssetId;
    request.ProductIds = listProductId;
    request.NewStartDate = objAssetLineItem.Apttus_Config2__StartDate__c;
    request.CartId = cartId;
}

// call swapAssets API
Apttus_Config2.CPQStruct.SwapAssetsResponseDO response =
Apttus_Config2.AssetService.swapAssets(request);

ApexPages.addMessage(new ApexPages.Message(ApexPages.severity.info,'swapAssets: ' +
response));

```

## Getting a list of Products to be swapped with Assets

The global method fetches a list of products that are to be swapped.

 Ensure that the Replacement Rule is active for Swapping Assets.

API	Signature
<b>getSwappedProducts</b>	<i>static Apttus_Config2.CPQStruct.RecommendationResponseDO getSwappedProducts(Apttus_Config2.CPQStruct.RecommendationRequestDO request)</i>

Parameters		
Name	Type	Description
request	Apttus_Config2.CPQStruct.RecommendationRequestDO	Request object invoked by the method
Request Data Object - Apttus_Config2.CPQStruct.RecommendationRequestDO()		
Name	Type	Description
CartId	Id	Id of Cart for which Swap action is performed
ProductIds	List<ID>	Id of the product with which the Asset is being swapped.
Response Data Object - Apttus_Config2.CPQStruct.RecommendationResponseDO		
Field	Type	Description
ProductIds	List<ID>	Ids of products that can be swapped with assets.

## Code Sample

The code sample below helps you fetch the list of Product Ids that can be swapped with Assets.

```
// create list of product ids
List<ID> listProductId = new List<ID>();
for (ProductWrapperClass product : wrapperProductList)
{
    if (product.selected)
    {
        listProductId.add(product.productId);
    }
}

// create and populate request object
```

```

Apttus_Config2.CPQStruct.RecommendationRequestDO request = new
    Apttus_Config2.CPQStruct.RecommendationRequestDO();
request.cartId = cartId;
request.ProductIds = listProductId;

// call getSwappedProducts API
Apttus_Config2.CPQStruct.RecommendationResponseDO response =
Apttus_Config2.AssetService.getSwappedProducts(request);

ApexPages.addMessage(new ApexPages.Message(ApexPages.severity.info, 'getSwappedProduct
s: ' + response));
    
```

## Terminating Assets

You can invoke this global method for asset cancellation.

API	Signature	
cancelAssets	<i>static Apttus_Config2.CPQStruct.CancelAssetsResponseDO cancelAssets(Apttus_Config2.CPQStruct.CancelAssetsRequestDO request)</i>	
Parameters		
Name	Type	Description
request	Apttus_Config2.CPQStruct.CancelAssetsRequestDO	Request object invoked by the method
Request Data Object - Apttus_Config2.CPQStruct.CancelAssetsRequestDO		
Name	Type	Description
CancelDate	Date	The termination date of the Asset
AssetIds	List<ID>	List of asset ids to terminate.
CartId	ID	The id of Cart for which an asset is being canceled.

Response Data Object - Apttus_Config2.CPQStruct.CancelAssetsResponseDO		
Field	Type	Description
LineItemMap	Map <ID, Apttus_Config2__LineItem__c>	Line Items that are to be terminated.

## Code Sample

The code sample below helps you terminate Assets based on the Termination Date.

```
// create list of asset ids
List<ID> listAssetId = new List<ID>();
for (AssetLineItemWrapperClass record : wrapperAssetLineItemList)
{
    if (record.selected)
    {
        listAssetId.add(record.assetId);
    }
}

// create and populate request object
if (objAssetLineItem.Apttus_Config2__CancelledDate__c != null)
    Apttus_Config2.CPQStruct.CancelAssetsRequestDO request = new
    Apttus_Config2.CPQStruct.CancelAssetsRequestDO();
    request.CancelDate = objAssetLineItem.Apttus_Config2__CancelledDate__c.date();
    request.AssetIds = listAssetId;
    request.CartId = cartId;
}

// call cancelAssets API
Apttus_Config2.CPQStruct.CancelAssetsResponseDO response =
Apttus_Config2.AssetService.cancelAssets(request);

ApexPages.addMessage(new ApexPages.Message(ApexPages.severity.info, 'cancelAssets: ' +
response));
```

## Renewing Assets

You can invoke this global method to renew Assets.

API	Signature	
<b>renewAssets</b>	<i>static Apttus_Config2.CPQStruct.RenewAssetsResponseDO renewAssets(Apttus_Config2.CPQStruct.RenewAssetsRequestDO request)</i>	
Parameters		
Name	Type	Description
request	Apttus_Config2.CPQStruct.RenewAssetsRequestDO()	Request object invoked by the method
Request Data Object - Apttus_Config2.CPQStruct.RenewAssetsRequestDO		
Name	Type	Description
FarthestAssetEndDate	Boolean	Setting this value to true renews assets using the farthest end date.
RenewEndDate	Date	The Asset End Date for Renewal
RenewTerm	Integer	The Renewal Term for an asset
AssetIds	List<ID>	List of asset ids
CartId	ID	The id of Cart for which an asset is being renewed
Response Data Object- Apttus_Config2.CPQStruct.RenewAssetsResponseDO		
Field	Type	Description
LineItemMap	Map<ID, Apttus_Config2__LineItem__c>	Returns all line items with all their field values.

### Code Sample

The code sample below helps you renew Assets based on the specified Renewal Date or Renewal Term.

```

// create list of asset ids
List<ID> listAssetId = new List<ID>();
for (AssetLineItemWrapperClass record : wrapperAssetLineItemList)
{
    if (record.selected)
    {
        listAssetId.add(record.assetId);
    }
}

// renew assets using an asset line item record with valid Renewal Date and Term
if (objAssetLineItem.Apttus_Config2__RenewalDate__c != null ||
objAssetLineItem.Apttus_Config2__RenewalTerm__c != null)
{

    // create and populate request object
    Apttus_Config2.CPQStruct.RenewAssetsRequestDO request = new
Apttus_Config2.CPQStruct.RenewAssetsRequestDO();
    request.CartId = cartId;
    request.AssetIds = listAssetId;
    request.RenewEndDate = objAssetLineItem.Apttus_Config2__RenewalDate__c;
    request.RenewTerm = objAssetLineItem.Apttus_Config2__RenewalTerm__c;
    request.FarthestAssetEndDate = false;

    // call renewAssets API
    Apttus_Config2.CPQStruct.RenewAssetsResponseDO response =
Apttus_Config2.AssetService.renewAssets(request);

    ApexPages.addMessage(new ApexPages.Message(ApexPages.severity.info, 'renewAssets:
' + response));
}

```

## Incrementing Assets

You can invoke **incrementAssets**, a global method to increment assets.

Method	Signature
<b>incrementAssets</b>	<i>static Apttus_Config2.CPQStruct.IncrementAssetsResponseDO incrementAssets(Apttus_Config2.CPQStruct.IncrementAssetRequestDO request)</i>

Parameters		
Name	Type	Description
request	Apttus_Config2.CPQStruct.IncrementAssetRequestDO	Request object invoked by the method
Request Data Object - Apttus_Config2.CPQStruct.IncrementAssetRequestDO		
Field	Type	Description
IncrementAssetDOs	List <Apttus_Config2.CPQStructIncrementAssetDO>	List of increment assets data objects.
CartId	ID	The ID of the cart that consists of assets to be incremented.
Data Object - Apttus_Config2.CPQStruct.IncrementAssetDO		
Field	Type	Description
AssetId	Id	The ID of the Asset you want to apply the increment.
LineAction	String	This defines the update made to the asset. There are two types. <ul style="list-style-type: none"> <li>• <i>Increment</i>: creates a new asset</li> <li>• <i>Increment or Merge</i>: updates the existing asset.</li> </ul>
NewStartDate	Date	New start date for the updated or new asset.
NewEndDate	Date	New end date for the updated or new asset.
Quantity	Integer	The amount of quantity to be incremented.



Response Data Object- Apttus_Config2.CPQStruct.IncrementAssetsResponseDO		
Field	Type	Description
LineItemMap	Map<ID, Apttus_Config2__LineItem__c>	Returns all line items with all their field values.

## Code Sample

The following sample enables you to increment the quantity of standalone products. You can also provide new start and end dates for the products. If you define **LineAction** as *Increment*, new assets are created and, if you define **LineAction** as *Increment and Merge*, existing assets are updated.

```
Apttus_Config2.CPQStruct.IncrementAssetRequestDO request = new
    Apttus_Config2.CPQStruct.IncrementAssetRequestDO();

for (AssetLineItemWrapperClass record : wrapperAssetLineItemList)
{
    if (record.selected)
    {
        // create and populate request object
        Apttus_Config2.CPQStruct.IncrementAssetDO assetDO = new
            Apttus_Config2.CPQStruct.IncrementAssetDO();
        assetDO.AssetId = record.assetId;
        assetDO.NewStartDate =
            record.objAssetLineItem.Apttus_Config2__StartDate__c;
        assetDO.NewEndDate = record.objAssetLineItem.Apttus_Config2__EndDate__c;
        assetDO.Quantity = record.objAssetLineItem.Apttus_Config2__Quantity__c;
        assetDO.LineAction = 'Increment'; // Or 'Increment and Merge'
        request.IncrementAssetDOs.add(assetDO);
    }
}
request.CartId = cartId;

// call incrementAssets API
Apttus_Config2.CPQStruct.IncrementAssetsResponseDO response =
    Apttus_Config2.AssetService.incrementAssets(request);
ApexPages.addMessage(new ApexPages.Message(ApexPages.severity.info, incrementAssets:
    ' + response));
```

## Changing Assets

You can invoke this global method for changing Assets.

API	Signature	
changeAssets	<i>static Apttus_Config2.CPQStruct.ChangeAssetsResponseDO changeAssets(Apttus_Config2.CPQStruct.ChangeAssetsRequestDO request)</i>	
Parameters		
Name	Type	Description
request	Apttus_Config2.CPQStruct.ChangeAssetsRequestDO	Request object invoked by the method
Request Data Object - Apttus_Config2.CPQStruct.ChangeAssetsRequestDO		
Name	Type	Description
AssetId	List<ID>	List of Asset IDs to change.
CartId	ID	The id of Cart for which an asset is being changed.
Response Data Object - Apttus_Config2.CPQStruct.ChangeAssetsResponseDO		
Field	Type	Description
Map <ID, LineItem__c>	LineItemMap	Returns all line items with all their field values.

### Code Sample

The code sample below helps you make changes to an Asset by invoking this global method.

```
// create list of asset ids
```

```

List<ID> listAssetId = new List<ID>();
for (AssetLineItemWrapperClass record : wrapperAssetLineItemList)
{
    if (record.selected)
    {
        listAssetId.add(record.assetId);
    }
}

// create and populate request object
Apttus_Config2.CPQStruct.ChangeAssetsRequestDO request = new
    Apttus_Config2.CPQStruct.ChangeAssetsRequestDO();
request.AssetId = listAssetId;
request.CartId = cartId;

// call changeAssets API
Apttus_Config2.CPQStruct.ChangeAssetsResponseDO response =
    Apttus_Config2.AssetStatusService.changeAssets(request);

ApexPages.addMessage(new ApexPages.Message(ApexPages.severity.info, 'changeAssets: '
    + response));

```

## Merging Duplicate Assets

This global method enables you to merge duplicate assets automatically.

Method	Signature
mergeAssets	<i>static Apttus_Config2.CPQStruct.MergeAssetsResponseDO mergeAssets(Apttus_Config2.CPQStruct.MergeAssetsRequestDO request)</i>

Parameters		
Name	Type	Description
Apttus_Config2.CPQStruct.MergeAssetsResponseDO	request	Request object invoked by the method

Request Data Object - Apttus_Config2.CPQStruct.MergeAssetsResponseDO		
Field	Type	Description
AssetIds	List	The list of the ID of assets you want to merge.
CartId	ID	The ID of the cart that consists of assets to be incremented.
MergeAssetItemDO	Apttus_Config2.CPQStruct.MergeAssetDO	The merge asset data objects.
Data Object - Apttus_Config2.CPQStruct.MergeAssetDO		
Field	Type	Description
EndDate	Date	The asset end date for merge.
ProductId	Id	The ID of the assets to be merged.
Quantity	Integer	The quantity of the asset.
StartDate	Date	The asset start date for merge.
Response Parameter - Apttus_Config2.CPQStruct.MergeAssetsResponseDO		
Field	Type	Description
Errors	List	Returns the list of errors if any.
LineItemMap	Map<ID, Apttus_Config2__LineItem__c>	Returns all line items with all their field values.

### Code Sample

The following sample enables you to merge a list of assets in a cart.

```

Apttus_Config2.CPQStruct.MergeAssetsRequestDO request = new
  Apttus_Config2.CPQStruct.MergeAssetsRequestDO();
for (AssetLineItemWrapperClass record : wrapperAssetLineItemList)
{
  if (record.selected)
  {
    request.AssetIds.add(record.assetId);
  }
}

request.CartId = 'a1I3J000000BKJeUA0';
request.MergeAssetItemDO.startdate = Date.newInstance (2017, 06, 01);

Apttus_Config2.CPQStruct.MergeAssetsResponseDO response =
  Apttus_Config2.AssetService.mergeAssets(request);
ApexPages.addMessage(new ApexPages.Message(ApexPages.severity.info, mergeAssets: ' +
  response));

```

## Suspending Assets

You can invoke this global method to suspend assets. You can process a maximum of 200 line items in each Suspend API call.

API	Signature	
<b>suspendAssets</b>	<i>static Apttus_Config2.CPQStruct.SuspendAssetResponseDO suspendAssets(Apttus_Config2.CPQStruct.SuspendAssetRequestDO request)</i>	
Parameters		
Name	Type	Description
request	Apttus_Config2.CPQStruct.SuspendAssetRequestDO	Request object invoked by the method.

Request Data Object - Apttus_Config2.CPQStruct.SuspendAssetRequestDO			
Field	Type	Description	Is Required
assetIds	List<Id>	List of asset IDs to be suspended. All the asset IDs must be valid and all the assets must be activated.	Yes
CartId	Id	Product configuration ID.	Yes
CustomData	Map<String, String>	Custom field values to be updated. For example, you can pass the reason for suspension as custom data. You can pass more than one custom field.	No
NewEndDate	Date	<p>Date until which assets must remain suspended. This field is optional in the input. CPQ copies this date to the End Date field in the cart line item.</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p><b>i</b> The end date passed for asset suspension must be less the end date of the asset. If you provide both start date and end date, CPQ considers the end date you provide in this field as the end date for asset suspension. If you do not provide both end date and term, CPQ considers the asset end date as the end date for asset suspension.</p> </div>	No
NewStartDate	Date	Date from which assets must be suspended. This field is mandatory in the input.	Yes

Request Data Object - Apttus_Config2.CPQStruct.SuspendAssetRequestDO			
Field	Type	Description	Is Required
Term	Integer	<p>Term for which assets must be suspended. This field is optional in the input. CPQ copies this date to the Start Date field in the cart line item.</p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p><b>i</b> If you do not provide the end date but provide the term, CPQ calculates the end date for asset suspension based on the asset suspension start date and term (according to the selling frequency). If you do not provide both end date and term, CPQ considers the asset end date as the end date for asset suspension.</p> </div>	No
Response Data Object - Apttus_Config2.CPQStruct.SuspendAssetResponseDO			
Field	Type	Description	
Errors	List<String>	Contains errors messages if any operations failed.	
IsSuccess	Boolean	Indicates whether the suspend asset operation is successful.	
LineItemMap	Map<ID,Apttus_Config2__LineItem__c>	Indicates the line items to be suspended.	

### Code Sample

The following code sample helps you suspend assets.

```
List<Apttus_Config2__AssetLineItem__c> assetLinesToSuspend = [SELECT Id,
Apttus_Config2__StartDate__c
```

```

                                                                    FROM
Apttus_Config2__AssetLineItem__c

                                                                    WHERE
Apttus_Config2__AccountId__c = '001630000142XpT'

                                                                    AND
Apttus_Config2__ProductId__r.Name = 'AutoABO_ORN_Standalone04'];

// asset ids to be suspended
List<Id> assetIds = new List<Id>();
for (Apttus_Config2__AssetLineItem__c assetLineS0 : assetLinesToSuspend)
{
    assetIds.add(assetLineS0.Id);
}

Apttus_Config2.CPQStruct.SuspendAssetRequestDO request = new
    Apttus_Config2.CPQStruct.SuspendAssetRequestDO();
request.NewStartDate = assetLinesToSuspend[0].Apttus_Config2__StartDate__c.addDays(90)
;
request.NewEndDate = assetLinesToSuspend[0].Apttus_Config2__StartDate__c.addDays(180)
;
request.CartId='a5663000002DlFN';
request.assetIds = assetIds;

//input custom field values
Map<String, String> CustomData = new Map<String, String>();
CustomData.put('Apttus_Config2__Comments__c', 'Suspending for 3 months due to late
payment');
request.CustomData = CustomData;

Apttus_Config2.CPQStruct.SuspendAssetResponseDO response =
Apttus_Config2.AssetService.suspendAssets(request);
system.debug(response.Errors);
system.debug(response.IsSuccess);

```

## Resuming Assets

You can invoke this global method to resume suspended assets.



API		Signature	
resumeAssets		<pre>static Apttus_Config2.CPQStruct.ResumeAssetResponseDO resumeAssets(Apttus_Config2.CPQStruct.ResumeAssetRequest DO request)</pre>	
Parameters			
Name	Type	Description	
Request	Apttus_Config2.CPQStruct.ResumeAssetRequestDO	Request object invoked by the method	
Request Data Object - Apttus_Config2.CPQStruct.ResumeAssetRequestDO			
Field	Type	Description	Is Required
assetIds	List<Id>	List of asset IDs to be resumed.	Yes
CartId	Id	Product configuration ID.	Yes
CustomData	Map<String, String>	Custom field values to be updated. For example, you can pass the reason for resumption as custom data. You can pass more than one custom field.	No
NewStartDate	Date	New start date for the resumed assets.	No
		<div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p><span style="font-size: 1.2em;">i</span> This parameter will be effective in future when Conga CPQ supports the resumption of assets from the date specified in this parameter.</p> </div>	

Response Data Object - Apttus_Config2.CPQStruct. ResumeAssetResponseDO		
Field	Type	Description
Errors	List<String>	Contains errors messages if any operations failed.
IsSuccess	Boolean	Indicates whether the resume asset operation is successful.
LineItemMap	Map <ID, Apttus_Config2__LineItem__c>	Indicates the line items to be resumed.

## Code Sample

The following code sample helps you suspend assets.

```
List<Apttus_Config2__AssetLineItem__c> assetLinesToResume = [SELECT Id,
Apttus_Config2__StartDate__c
FROM
Apttus_Config2__AssetLineItem__c
WHERE
Apttus_Config2__AccountId__c = '001630000142XpT'
AND
Apttus_Config2__ProductId__r.Name = 'AutoABO_ORN_Standalone04'];

// asset ids to be suspended
List<Id> assetIds = new List<Id>();
for (Apttus_Config2__AssetLineItem__c assetLineSO : assetLinesToResume)
{
    assetIds.add(assetLineSO.Id);
}

CPQStruct.ResumeAssetRequestDO request = new CPQStruct.ResumeAssetRequestDO();
request.CartId = 'a5663000002DQKs';
request.assetIds = assetIds;
request.NewStartDate = Date.today();
CPQStruct.ResumeAssetResponseDO response = AssetService.resumeAssets(request);
system.debug( response );
```

# Remote CPQ Admin Controller

The Remote CPQ Admin Controller methods account for the standard operation in CPQ Admin.

You can invoke methods in using the following command:

```
Apttus_Config2.RemoteCPQAdminController.<Name of the Function>
where the name of the function is API Name and it parameters.
```

Global methods are not APIs. They can only be invoked in an Apex Code.

In this Section:

- [Searching Custom Fields in Objects](#)

## Searching Custom Fields in Objects

You can invoke this global method to retrieve the list of objects in which a given custom field is used.

API	Signature	
findCriteriaFieldInCustomSettings	static Apttus_Config2.CPQAdminStruct2.CriteriaFieldSearchResponse findCriteriaFieldInCustomSettings(Apttus_Config2.CPQAdminStruct2.CriteriaFieldSearchRequest request)	
<b>Parameters</b>		
Name	Type	Description
request	Apttus_Config2.CPQAdminStruct2.CriteriaFieldSearchRequest	Request object invoked by the method.
<b>Resquest Data Object - Apttus_Config2.CPQAdminStruct2.CriteriaFieldSearchResponse</b>		
Field	Type	Description
FieldName	String	The name field that you want to search.

Response Data Object - Apttus_Config2.CPQAdminStruct2.CriteriaFieldSearchRequest		
Name	Type	Description
CustomSettingNames	List	The list of custom settings in which the requested field is present.

### Code Sample

The following code sample allows you to fetch the list of objects in which the request custom field is used.

```
Apttus_Config2.CPQAdminStruct2.CriteriaFieldSearchRequest request = new
Apttus_Config2.CPQAdminStruct2.CriteriaFieldSearchRequest();
request.FieldName = 'Quantity__c'; // namespace prefixed field name to search
Apttus_Config2.CPQAdminStruct2.CriteriaFieldSearchResponse response =
Apttus_Config2.RemoteCPQAdminController.findCriteriaFieldInCustomSettings(request);
System.debug('*** Matching Custom Setting Names: ' +
JSON.serialize(response.CustomSettingNames));
/* Matching Custom Setting Names: ["Config Asset Pricing Criteria Fields","Config
LineItem Criteria Fields","Config Expression Criteria Fields","Config Custom Display
Columns","Config Data Cache","Config Field Set"]*/
```

## Asset Web Services

This section describes the following ABO web services:

- [Change Assets](#)
- [Renew Assets](#)
- [Get a List of Products to be Swapped with Assets](#)
- [Swap Assets](#)
- [Terminate Assets](#)
- [Get a Count of Assets](#)
- [Get a List of Assets](#)

# Change Assets

You can invoke this API to change the value of assets.

API	Signature	
changeAssets	<i>WebService static Apttus_CPQApi.CPQAsset.ChangeAssetsResponseDO changeAssets(Apttus_CPQApi.CPQAsset.ChangeAssetsRequestDO request)</i>	
Parameters		
Name	Type	Description
request	Apttus_CPQApi.CPQAsset.ChangeAssetsRequestDO	Request object passed by the invoker method.
Request Data Object - Apttus_CPQApi.CPQAsset.ChangeAssetsRequestDO		
Name	Type	Description
AssetIds	List<ID>	List of asset IDs to be amended.
CartId	ID	ID of the cart for which an asset is being amended.
Response Data Object - Apttus_CPQApi.CPQAsset.ChangeAssetsResponseDO		
Field	Type	Description
LineItemMap	List<LineItemEntryDO>	Line items that are amended.
Errors	List<String>	List of error messages if any error occurred.
LineItemEntryDO		
Name	Type	Description
LineItemId	ID	Lineitem ID

LineItemEntryDO		
Name	Type	Description
LineItemSO	Apttus_Config2__LineItem__c	LineItem SObject

## Code Sample

The following code sample helps you change the value of assets.

```
Public Apttus_CPQApi.CPQAsset.ChangeAssetsResponseDO changeAssets()
{
    // create and populate request object
    Apttus_CPQApi.CPQAsset.ChangeAssetsRequestDO request = new
    Apttus_CPQApi.CPQAsset.ChangeAssetsRequestDO();

    request.CartId = 'a1I6C000000k6lmUAA';
    // list of primary asset ids to be amended
    request.AssetIds = new List<ID> {'a0e6C000001uATS', 'a0e6C000001aULM'};

    // call changeAssets API
    Apttus_CPQApi.CPQAsset.ChangeAssetsResponseDO response =
    Apttus_CPQApi.CPQAssetWebService.changeAssets(request);
    return response;
}
```

## Renew Assets

You can invoke this API to renew assets.

API	Signature
<b>renewAssets</b>	<i>WebService static Apttus_CPQApi.CPQAsset.RenewAssetsResponseDO renewAssets(Apttus_CPQApi.CPQAsset.RenewAssetsRequestDO request)</i>

<b>Parameters</b>		
<b>Name</b>	<b>Type</b>	<b>Description</b>
request	Apttus_CPQApi.CPQAsset.RenewAssetsRequestDO	Request object passed by the invoker method.
<b>Request Data Object - Apttus_CPQApi.CPQAsset.RenewAssetsRequestDO</b>		
<b>Name</b>	<b>Type</b>	<b>Description</b>
RenewEndDate	Date	Asset end date for renewal.
FarthestAssetEndDate	Boolean	Setting this value to true renews assets using the farthest end date.
RenewTerm	Integer	Renewal term for an asset.
AssetIds	List<ID>	List of asset IDs to be renewed.
CartId	ID	ID of the cart for which an asset is being renewed.
<b>Response Data Object - Apttus_CPQApi.CPQAsset.RenewAssetsResponseDO</b>		
<b>Field</b>	<b>Type</b>	<b>Description</b>
LineItemMap	List<LineItemEntryDO>	Line items that are renewed.
Errors	List<String>	List of error messages if any error occurred.
<b>LineItemEntryDO</b>		
<b>Name</b>	<b>Type</b>	<b>Description</b>
LineItemId	ID	Lineitem ID
LineItemSO	Apttus_Config2__LineItem__c	Lineitem SObject

### Code Sample

The following code sample helps you renew assets based on the renewal date, farthest asset end date, or renewal term.

```

Public Apttus_CPQApi.CPQAsset.RenewAssetsResponseDO renewAssets()
{
    // create and populate request object
    Apttus_CPQApi.CPQAsset.RenewAssetsRequestDO request = new
Apttus_CPQApi.CPQAsset.RenewAssetsRequestDO();
    request.CartId = 'a1I6C000000k6lmUAA';
    // list of primary asset ids to be renewed
    request.AssetIds = new List<ID> {'a0e6C000001uATS'};
    request.renewenddate = Date.newInstance(YYYY, MM, DD);

    // call renewAssets API
    Apttus_CPQApi.CPQAsset.RenewAssetsResponseDO response =
Apttus_CPQApi.CPQAssetWebService.renewAssets(request);
    return response;
}
    
```

## Get a List of Products to be Swapped with Assets

You can invoke this API to fetch a list of products to be swapped with assets. Relevant replacement rules should be active for getting the list of products that can be swapped with assets.

API	Signature	
getSwappedProducts	<i>WebService static Apttus_CPQApi.CPQAsset.RecommendationResponseDO getSwappedProducts(Apttus_CPQApi.CPQAsset.RecommendationRequestDO request)</i>	
Parameters		
Name	Type	Description
request	Apttus_CPQApi.CPQAsset.RecommendationRequestDO	Request object passed by the invoker method.



Request Data Object - Apttus_CPQApi.CPQAsset.RecommendationRequestDO		
Name	Type	Description
ProductIds	List<ID>	List of product IDs with which an asset is being swapped.
CartId	ID	ID of the cart for which the swap action is performed.

Response Data Object - Apttus_CPQApi.CPQAsset.RecommendationResponseDO		
Field	Type	Description
ProductIds	List<ID>	List of product IDs that can be swapped with assets.
Errors	List<String>	List of error messages if any error occurred.

### Code Sample

The following code sample helps you fetch the list of product IDs that can be swapped with assets.

```

Public Apttus_CPQApi.CPQAsset.RecommendationResponseDO getSwappedProducts()
{
    // create and populate request object
    Apttus_CPQApi.CPQAsset.RecommendationRequestDO request = new
Apttus_CPQApi.CPQAsset.RecommendationRequestDO();
    request.ProductIds = new List<ID> {'01t3C000000l8qi', '01t2f000000dV1d'};
    request.CartId = 'a1I6C000000k6lmUAA';
    // call getSwappedProducts API
    Apttus_CPQApi.CPQAsset.RecommendationResponseDO response =
Apttus_CPQApi.CPQAssetWebService.getSwappedProducts(request);
    return response;
}

```

# Swap Assets

You can invoke this API to swap assets.

API	Signature	
<b>swapAssets</b>	<i>webService static Apttus_CPQApi.CPQAsset.SwapAssetsResponseDO swapAssets(Apttus_CPQApi.CPQAsset.SwapAssetsRequestDO request)</i>	
Parameters		
Name	Type	Description
request	Apttus_CPQApi.CPQAsset.SwapAssetsRequestDO	Request object passed by the invoker method.
Request Data Object - Apttus_CPQApi.CPQAsset.SwapAssetsRequestDO		
Name	Type	Description
NewStartDate	Date	The new Asset Start Date on Swapping
ProductIds	List<ID>	Id of the product with which an Asset is being swapped
AssetIds	List<ID>	ID of an asset that is swapped.
CartId	ID	ID of the cart for which an asset is being swapped.
Response Data Object - Apttus_CPQApi.CPQAsset.SwapAssetsResponseDO		
Field	Type	Description
LineItemMap	List<LineItemEntryDO>	Line items that are cancelled and swapped.
Errors	List<String>	List of error messages if any error occurred.

LineItemEntryDO		
Name	Type	Description
LineItemId	ID	Lineitem ID
LineItemSO	Apttus_Config2__LineItem__c	Lineitem SObject

## Code Sample

The following code sample helps you swap assets.

```

Public Apttus_CPQApi.CPQAsset.SwapAssetsResponseDO swapAssets()
{
    // create and populate request object
    Apttus_CPQApi.CPQAsset.SwapAssetsRequestDO request = new
Apttus_CPQApi.CPQAsset.SwapAssetsRequestDO();
    request.CartId = 'sa1I6C000000k6lmUAA';
    request.NewStartDate = Date.newInstance(YYYY, MM, DD);
    request.ProductIds = new List<ID>{'01t3C000002oEVL'};
    // only one asset can be swapped per API call
    request.AssetIds = new List<ID> {'a0e6C000001uATS'};

    // call swapAssets API
    Apttus_CPQApi.CPQAsset.SwapAssetsResponseDO response =
Apttus_CPQApi.CPQAssetWebService.swapAssets(request);
    return response;
}

```

## Terminate Assets

You can invoke this API to terminate or cancel assets.

API	Signature
cancelAssets	<i>WebService static Apttus_CPQApi.CPQAsset.CancelAssetsResponseDO cancelAssets(Apttus_CPQApi.CPQAsset.CancelAssetsRequestDO request)</i>

Parameters		
Name	Type	Description
request	Apttus_CPQApi.CPQAsset.CancelAssetsRequestDO	Request object passed by the invoker method.
Request Data Object - Apttus_CPQApi.CPQAsset.CancelAssetsRequestDO		
Name	Type	Description
CancelDate	Date	The termination date of an asset.
AssetIds	List<ID>	List of asset IDs to be terminated.
CartId	ID	ID of the cart for which an asset is being terminated.
Response Data Object - Apttus_CPQApi.CPQAsset.CancelAssetsResponseDO		
Field	Type	Description
LineItemMap	List<LineItemEntryDO>	Line items that are terminated.
Errors	List<String>	List of error messages if any error occurred.
LineItemEntryDO		
Name	Type	Description
LineItemId	ID	Lineitem ID
LineItemSO	Apttus_Config2__LineItem__c	Lineitem SObject

### Code Sample

The following code sample helps you terminate assets.

```

Public Apttus_CPQApi.CPQAsset.CancelAssetsResponseDO cancelAssets()
{
    // create and populate request object
    Apttus_CPQApi.CPQAsset.CancelAssetsRequestDO request = new
Apttus_CPQApi.CPQAsset.CancelAssetsRequestDO();

    request.CartId = 'a1I6C000000ju66';
    // list of primary asset ids
    request.AssetIds = new List<ID> {'a0e6C000001tWxA', 'a0e6C000001wVWB'};
    request.CancelDate = Date.newInstance(YYYY, MM, DD);

    // call cancelAssets API
    Apttus_CPQApi.CPQAsset.CancelAssetsResponseDO response =
Apttus_CPQApi.CPQAssetWebService.cancelAssets(request);
    return response;
}

```

## Get a Count of Assets

This API returns the count of asset line items of various accounts.

API	Signature	
countAssetLineItems	<i>WebService static Apttus_CPQApi.CPQAsset.QueryAssetsResponseDO countAssetLineItems (Apttus_CPQApi.CPQAsset.QueryAssetsRequestDO request)</i>	
Parameters		
Name	Type	Description
request	Apttus_CPQApi.CPQAsset.QueryAssetsRequestDO	Request object passed by the invoker method.
Request Data Object - Apttus_CPQApi.CPQAsset.QueryAssetsRequestDO		
Name	Type	Description
AccountIds	List<ID>	List of account IDs.

Request Data Object - Apttus_CPQApi.CPQAsset.QueryAssetsRequestDO		
Name	Type	Description
CustomFilter	String	SOQL condition expression used as a custom filter. For example, you can request only assets after a specified billing end date. This is optional.
Descending	Boolean	N/A
FieldNames	List<String>	N/A
SortFields	List<String>	N/A
Offset	Integer	N/A
Nrecord	Integer	N/A
Response Data Object - Apttus_CPQApi.CPQAsset.QueryAssetsResponseDO		
Field	Type	Description
AssetCount	Integer	Number of assets returned.
AssetLineItems	List<Apttus_Config2__AssetLineItem__c>	N/A
Errors	List<String>	List of error messages if any error occurred.

### Code Sample

The following code sample helps you fetch the count of asset line items of various accounts.

```

Public Apttus_CPQApi.CPQAsset.QueryAssetsResponseDO countAssetLineItems()
{

```

```

Apttus_CPQApi.CPQAsset.QueryAssetsRequestDO request = new
Apttus_CPQApi.CPQAsset.QueryAssetsRequestDO();
request.AccountIds = new List<ID>{'0012f00000BQd03'};
request.CustomFilter = 'Apttus_Config2__PriceType__c = \'Recurring\''; //
optional
// call countAssetLineItems API
Apttus_CPQApi.CPQAsset.QueryAssetsResponseDO response =
Apttus_CPQApi.CPQAssetWebService.countAssetLineItems(request);
return response;
}

```

## Get a List of Assets

This API returns the list of asset line items of various accounts.

API	Signature	
getAssetLineItems	<i>WebService static Apttus_CPQApi.CPQAsset.QueryAssetsResponseDO getAssetLineItems(Apttus_CPQApi.CPQAsset.QueryAssetsRequestDO request)</i>	
Parameters		
Name	Type	Description
request	Apttus_CPQApi.CPQAsset.QueryAssetsRequestDO	Request object passed by the invoker method.
Request Data Object - Apttus_CPQApi.CPQAsset.QueryAssetsRequestDO		
Name	Type	Description
AccountIds	List<ID>	List of account IDs.
FieldNames	List<String>	Optional. Use this to include fields you have added to an Asset Line Item object.

<b>Request Data Object - Apttus_CPQApi.CPQAsset.QueryAssetsRequestDO</b>		
<b>Name</b>	<b>Type</b>	<b>Description</b>
CustomFilter	String	Optional. SOQL condition expression used as a custom filter. For example, you can request only assets after a specified billing end date.
Descending	Boolean	Optional. This indicates whether assets are sorted in descending order.
SortFields	List<String>	Optional. This can be used sort the result based on a list of fields.
Offset	Integer	Optional. This can be used to include an offset.
Nrecord	Integer	Optional. This can be used to limit the number of records returned.
<b>Response Data Object - Apttus_CPQApi.CPQAsset.QueryAssetsResponseDO</b>		
<b>Field</b>	<b>Type</b>	<b>Description</b>
AssetCount	Integer	N/A
AssetLineItems	List<Apttus_Config2__AssetLineItem__c>	List of asset line items.
Errors	List<String>	List of error messages if any error occurred.

**Code Sample**



The following code sample helps you get a list of asset line items of various accounts.

```
Public Apttus_CPQApi.CPQAsset.QueryAssetsResponseDO getAssetLineItems ()
{
    Apttus_CPQApi.CPQAsset.QueryAssetsRequestDO request = new
    Apttus_CPQApi.CPQAsset.QueryAssetsRequestDO();
    request.AccountIds = new List<ID>{'0012f00000BQd03'};
    request.CustomFilter = 'Apttus_Config2__PriceType__c = \'Recurring\''; //
optional
    // retrieve all fields in Asset Line Item sObject
    request.FieldNames = null;
    // sort by billing end date (optional)
    request.SortFields = new List{'Apttus_Config2__BillingEndDate__c'};

    // call countAssetLineItems API
    Apttus_CPQApi.CPQAsset.QueryAssetsResponseDO response =
    Apttus_CPQApi.CPQAssetWebService.getAssetLineItems (request);
    return response;
}
```

## Merge Web Service


The Merge web service APIs account for the standard actions on quotes.

You can invoke APIs in Web Service from the following command:

```
Apttus_Proposal.MergeWebService.<Name of the Function>
where the name of the function is API Name and it's parameters.
```

You can use the Merge web service APIs to complete the following tasks:

- [Creating Proposal Document](#)
- [Creating Proposal Document with Draft Indication](#)
- [Generating Documents for Proposal with Large Number of Line Items](#)
- [Generating Documents Asynchronously](#)

 The Salesforce API version used in CPQ SOAP APIs is upgraded to 50.0.

## Creating Proposal Document

You can use this API to create proposal documents for the given format, template name, document type, and protection level. As a callback action, the API creates a task for further action.

API		Signature	
<b>generateDoc</b>		<i>webService static Id generateDoc(Id templateId, Id proposalId, String pLevel, String docFormat, String sessionId, String serverUrl)</i>	
Request Parameter			
Name	Type	Required?	Description
templateId	ID	Yes	The ID of the template you want to use for the proposal.
proposalId	ID	Yes	The ID of the proposal for which you want to generate the document.
pLevel	String	Yes	The protection level you want to apply on the document
docFormat	String	Yes	The format in which you want to generate the document. The valid values for this parameter are listed below: <ul style="list-style-type: none"> <li>• PDF</li> <li>• DOC</li> <li>• DOCX</li> <li>• RTF</li> </ul>
sessionId	String	Yes	The session ID you want to use for the callback.
serverUrl	String	Yes	The server URL you want to use for the callback.

Response Parameter		
Field	Type	Description
documentID	ID	The ID of the generated document.

## Code Sample

The sample code below enables you to create a proposal document by providing a valid proposal ID, template name, protection level, and format. After the execution, it returns the proposal document ID.

```

1  /**
2   * The below method demonstrates how to create proposal document by
3   * passing the proposal ID(auto generated ID as record name),
4   * template name, protection level and document format.
5   * Possible protection levels are:
6   * - Full access,
7   * - Insert comments and tracked changes only,
8   * - Insert comments only,
9   * - Fill in form fields only and
10  * - Read only
11  */
12  public Id createDocumentForQuote(String ProposalID, String templateName,
13  String protectionLevel, String docFormat)
14  {
15      Id documentId;
16      Id proposalSOId = [SELECT ID FROM Apttus_Proposal__Proposal__c WHERE
17  Name = :ProposalID LIMIT 1].Id;
18      Id templateSOId = [SELECT Id, Name
19  FROM Apttus__APTS_Template__c
20  WHERE Name = :templateName AND
21  Apttus__IsActive__c = TRUE LIMIT 1].Id;
22      if (proposalSOId != null && templateSOId != null)
23      {
24          String serverUrl;
25          Apttus_Proposal__ProposalSystemProperties__c prop =
26  Apttus_Proposal__ProposalSystemProperties__c.getInstance('System
27  Properties');
28          if (prop.Apttus_Proposal__EnableFile__c )
29          {
30              serverUrl = URL.getSalesforceBaseUrl().toExternalForm()

```

```

26         + '/services/Soap/u/50.0/' + UserInfo.getOrganizationId();
27
28     }
29     else
30     {
31         serverUrl = URL.getSalesforceBaseUrl().toExternalForm()
32         + '/services/Soap/u/50.0/' + UserInfo.getOrganizationId();
33     }
34
35     documentId =
Apttus_Proposal.MergeWebService.generateDoc(templateS0Id,
36
proposalS0Id,
37
protectionLevel,
38
docFormat,
39
userInfo.getSessionId(),
40
serverUrl);
41     }
42     return documentId;
43 }

```

## Creating Proposal Document with Draft Indication

You can use this API to create proposal documents for the given format, template name, document type, protection level, and an indicator of whether the document is a draft or final document. As a callback action, the API creates a task for further action.

API	Signature
<b>generateDoc2</b>	<i>webService static Id generateDoc2(Id templateId, Id proposalId, String pLevel, String docFormat, Boolean isDraft, String sessionId, String serverUrl)</i>

Request Parameter			
Name	Type	Required?	Description
templateId	ID	Yes	The ID of the template you want to use for the proposal.
proposalId	ID	Yes	The ID of the proposal for which you want to generate the document.
pLevel	String	Yes	The protection level you want to apply on the document
docFormat	String	Yes	The format in which you want to generate the document. The valid values for this parameter are listed below: <ul style="list-style-type: none"> <li>• PDF</li> <li>• DOC</li> <li>• DOCX</li> <li>• RTF</li> </ul>
isDraft	Boolean	Yes	Indicate whether the generated document is a draft or a final document.
sessionId	String	Yes	The session ID you want to use for the callback.
serverUrl	String	Yes	The server URL you want to use for the callback.
Response Parameter			
Field	Type	Description	
documentID	ID	The ID of the generated document.	

## Code Sample

The sample code below enables you to create a proposal document by providing a valid proposal ID, template name, protection level, format, draft state. After the execution, it returns the proposal document ID.

```

1  /**
2   * The below method demonstrates how to create draft or final proposal
3   * document by passing the proposal ID(auto generated ID as record name),
4   * template name, protection level, document format and isDraft indicator.
5   * Possible protection levels are:
6   * - Full access,
7   * - Insert comments and tracked changes only,
8   * - Insert comments only,
9   * - Fill in form fields only and
10  * - Read only
11  */
12  public Id createDocumentForQuote(String ProposalID, String templateName,
13  String protectionLevel, String docFormat, Boolean isDraft){
14  Id documentId;
15  Id proposalSOId = [SELECT ID FROM Apttus_Proposal__Proposal__c WHERE
16  Name = :ProposalID LIMIT 1].Id;
17  Id templateSOId = [SELECT Id, Name FROM Apttus__APTS_Template__cWHERE
18  Name = :templateName AND Apttus__IsActive__c = TRUE LIMIT 1].Id;
19  if (proposalSOId != null && templateSOId != null)
20  {
21  String serverUrl;
22  Apttus_Proposal__ProposalSystemProperties__c prop =
23  Apttus_Proposal__ProposalSystemProperties__c.getInstance('System
24  Properties');
25
26  if (prop.Apttus_Proposal__EnableFile__c )
27  {
28  serverUrl = URL.getSalesforceBaseUrl().toExternalForm()
29  + '/services/Soap/u/50.0/' +
30  UserInfo.getOrganizationId();
31
32  }
33  else
34  {
35  serverUrl = URL.getSalesforceBaseUrl().toExternalForm()
36  + '/services/Soap/u/50.0/' +
37  UserInfo.getOrganizationId();
38
39  }
40  }

```

```

31
32     documentId =
Apttus_Proposal.MergeWebService.generateDoc2(templateS0Id,
33
34     proposalS0Id,
35
36     protectionLevel,
37
38     docFormat,
39
40     isDraft,
41     userInfo.getSessionId(),
42
43     serverUrl);
44     }
45     return documentId;
46 }

```

## Generating Documents for Proposal with Large Number of Line Items

You can use this API to generate documents for proposals with a large number of line items, asynchronously for the given parameters.

API	Signature		
<code>submitGenerateLargeDoc</code>	<i>webService static Id submitGenerateLargeDoc(Id templateId, Id proposalId, String pLevel, String docFormat, Boolean isDraft, String sessionId, String serverUrl)</i>		
Request Parameter			
Name	Type	Required?	Description
templateId	ID	Yes	The ID of the template you want to use for the proposal.
proposalId	ID	Yes	The ID of the proposal for which you want to generate the document.

Request Parameter			
Name	Type	Required?	Description
pLevel	String	Yes	The protection level you want to apply on the document
docFormat	String	Yes	The format in which you want to generate the document. The valid values for this parameter are listed below: <ul style="list-style-type: none"> <li>• PDF</li> <li>• DOC</li> <li>• DOCX</li> <li>• RTF</li> </ul>
isDraft	Boolean	Yes	Indicate whether the generated document is a draft or a final document.
sessionId	String	Yes	The session ID you want to use for the callback.
serverUrl	String	Yes	The server URL you want to use for the callback.
Response Parameter			
Field	Type	Description	
asynccallRequestId	ID	The ID of the asynchronous merge call SObject.	

### Code Sample

The sample code below enables you to create a proposal document asynchronously for a proposal with a large number of line items, by providing a valid proposal ID, template name, protection level, format, draft state. After the execution, it returns the ID of the asynchronous merge call SObject.



```

1  /**
2   * The below method demonstrates how to create draft or final proposal
3   * document asynchronously by passing the proposal ID(auto generated ID as
4   * record name),
5   * template name, protection level, document format and isDraft indicator.
6   * Possible protection levels are:
7   * - Full access,
8   * - Insert comments and tracked changes only,
9   * - Insert comments only,
10  * - Fill in form fields only and
11  * - Read only
12  */
13  public Id submitGenerateLargeDoc (String ProposalID, String templateName,
14  String protectionLevel, String docFormat, Boolean isDraft){
15      Id asyncCallRequestId;
16      Id proposalSOId = [SELECT ID FROM Apttus_Proposal__Proposal__c WHERE
17  Name = :ProposalID LIMIT 1].Id;
18      Id templateSOId = [SELECT Id, Name
19  FROM Apttus__APTS_Template__c
20  WHERE Name = :templateName AND
21  Apttus__IsActive__c = TRUE
22  LIMIT 1].Id;
23      if (proposalSOId != null && templateSOId != null)
24      {
25          String serverUrl;
26          Apttus_Proposal__ProposalSystemProperties__c prop =
27  Apttus_Proposal__ProposalSystemProperties__c.getInstance('System
28  Properties');
29
30          if (prop.Apttus_Proposal__EnableFile__c )
31          {
32              serverUrl = URL.getSalesforceBaseUrl().toExternalForm()
33              + '/services/Soap/u/50.0/' +
34  UserInfo.getOrganizationId();
35
36          }
37          else
38          {
39              serverUrl = URL.getSalesforceBaseUrl().toExternalForm()
40              + '/services/Soap/u/50.0/' +
41  UserInfo.getOrganizationId();
42          }
43      }
44  }
45  }

```

```

36         asyncCallRequestId =
Apttus_Proposal.MergeWebService.submitGenerateLargeDoc(templateS0Id,
proposalS0Id, protectionLevel, docFormat, isDraft,
userInfo.getSessionId(), serverUrl);
37     }
38     return asyncCallRequestId;
39 }
    
```

## Generating Documents Asynchronously

You can use this API to generate documents asynchronously for the given parameters.

API		Signature	
submitGenerateDoc		<i>webService static Id submitGenerateDoc(Id templateId, Id proposalId, String pLevel, String docFormat, Boolean isDraft, String sessionId, String serverUrl)</i>	
Request Parameter			
Name	Type	Required?	Description
templateId	ID	Yes	The ID of the template you want to use for the proposal.
proposalId	ID	Yes	The ID of the proposal for which you want to generate the document.
pLevel	String	Yes	The protection level you want to apply on the document
docFormat	String	Yes	The format in which you want to generate the document. The valid values for this parameter are listed below: <ul style="list-style-type: none"> <li>• PDF</li> <li>• DOC</li> <li>• DOCX</li> <li>• RTF</li> </ul>

Request Parameter			
Name	Type	Required?	Description
isDraft	Boolean	Yes	Indicate whether the generated document is a draft or a final document.
sessionId	String	Yes	The session ID you want to use for the callback.
serverUrl	String	Yes	The server URL you want to use for the callback.
Response Parameter			
Field	Type	Description	
asynccallRequestId	ID	The ID of the asynchronous merge call SObject.	

## Code Sample

The sample code below enables you to create a proposal document asynchronously by providing a valid proposal ID, template name, protection level, format, draft state. After the execution, it returns the ID of the asynchronous merge call SObject.

```

1  /**
2   * The below method demonstrates how to create draft or final proposal
3   * document asynchronously by passing the proposal ID(auto generated ID as
4   * record name),
5   * template name, protection level, document format and isDraft indicator.
6   * Possible protection levels are:
7   * - Full access,
8   * - Insert comments and tracked changes only,
9   * - Insert comments only,
10  * - Fill in form fields only and
11  * - Read only
12  */
13  public Id generateDocAsync (String ProposalID, String templateName, String
14  protectionLevel, String docFormat, Boolean isDraft){

```

```

12     Id asyncCallRequestId;
13     Id proposalSOId = [SELECT ID FROM Apttus_Proposal__Proposal__c WHERE
Name = :ProposalID LIMIT 1].Id;
14     Id templateSOId = [SELECT Id, Name
15                             FROM Apttus__APTS_Template__c
16                             WHERE Name = :templateName AND
17                             Apttus__IsActive__c = TRUE
18                             LIMIT 1].Id;
19     if (proposalSOId != null && templateSOId != null)
20     {
21         String serverUrl;
22         Apttus_Proposal__ProposalSystemProperties__c prop =
Apttus_Proposal__ProposalSystemProperties__c.getInstance('System
Properties');
23
24         if (prop.Apttus_Proposal__EnableFile__c )
25         {
26             serverUrl = URL.getSalesforceBaseUrl().toExternalForm()
27             + '/services/Soap/u/50.0/' + UserInfo.getOrganizationId();
28
29         }
30         else
31         {
32             serverUrl = URL.getSalesforceBaseUrl().toExternalForm()
33             + '/services/Soap/u/50.0/' +
UserInfo.getOrganizationId();
34         }
35
36         asyncCallRequestId =
Apttus_Proposal.MergeWebService.submitGenerateDoc(templateSOId,
37
proposalSOId,
38
protectionLevel,
39
docFormat,
40
isDraft,
41
userInfo.getSessionId(),
42
serverUrl);
43     }

```

```
44     return asyncCallRequestId;  
45 }
```

# Async APIs Using Batch Apex

Async APIs are used to execute certain operations using Batch Apex Jobs internally. These Batch jobs are usually used for carts with a large number of line items. You can use standard Salesforce APIs like Database.executeBatch, to trigger the Apex Jobs. IDs of these apex Jobs can be used to track the status of the job. You are notified through email about the completion of the job, as there is no synchronous response from the API.

The following Batch Apex job is described in this section:

- [Repricing the Cart](#)

## Repricing the Cart

You can use this Batch Apex Job to price the Line Items in a given Cart asynchronously. This Batch Apex creates one batch for each primary line item in the cart.

API		Signature of the Constructor	
RepriceCartJob		<i>global RepriceCartJob(ID cartId, String cartTitle);</i>	
Request Parameter			
Name	Type	Required?	Description
cartId	ID	Yes	The ID of cart you want to reprice.
cartTitle	String	Yes	The title of the cart.

### Code Sample

The below code sample returns job ID and reprices the line item one by one based on the batch size. The cart used in the sample code has products that added both manually and through an API. After the job is completed, CPQ sends you the email confirming the completion of the repricing action. If the Apex class encounters any run-time errors, the repricing is aborted.

```

1 public Id rePriceCartJob(Id cartId, String cartTitle) {
2     // create the reprice cart batch job

```

```
3 ID jobId = Database.executeBatch(new Apttus_Config2.RepriceCartJob(cartId,  
4     return jobId;  
5 }
```

## Scenarios

- [Working with Products in the Shopping Cart](#)

After you set up the user roles and use either sites or communities to build the basic framework for your cart page, create a visual force page and a sample CSS to define the look and feel for your cart. You can then proceed with using the CPQ APIs to create a working catalog page.

## Working with Products in the Shopping Cart

After you set up the user roles and use either sites or communities to build the basic framework for your cart page, create a visual force page and a sample CSS to define the look and feel for your cart. You can then proceed with using the CPQ APIs to create a working catalog page.

- [Searching Products](#)

- [Configuring and Adding Products](#)

In the product catalog, you can see the list of products that a vendor offers. Using this list, you can add a product to your cart, configure the product attributes, such as color and memory. If the product is a bundled product, you can select the options with which you want to bundle your product. This section lists the scenarios using which the customer can view the configuration options for a product.

- [Displaying Recommendations](#)

In the product catalog, you can see the list of products that a vendor offers. When the customer selects a product, you can display the list of recommended products on the same page. This section lists the scenarios using which the recommended products appear. To configure recommendations for a product use the following APIs:

- [Updating Price](#)

- [Viewing Price Break-up Details](#)

- [Adding Price Ramps to a Cart](#)

After you add a line item to a cart, this API enables you to add primary and secondary ramp line items for the line item. Once the ramp line items are created, you can also update the ramp line item details or delete ramp line item details using standard SOQL queries.

- **Viewing Configured Order in the Shopping Cart**

After the customer has configured and added all the products to the cart, display the



detailed shopping cart items. This section lists all the scenarios using which the customer can view the finalized shopping cart.

- [Updating Taxes and Shipping for an Order](#)
- [Creating and Updating Quote and Quote Line Items](#)
- [Creating and Updating Order and Order Line Items](#)
- [Asset-Based Ordering](#)

The end-user or a renewals manager can view and modify products that the organization owns and should be able to filter the assets to renew orders. Display and modify the assets that belong to a company based on the following criteria:

- [Finalizing the Cart](#)
- [Updating Quote Terms to Modify Dates](#)

## Searching Products

A customer browsing a product catalog can navigate for a product and select the required product manually or can search for a product. You can enable searches using the search text field in the cart page and enable the user to search for a desired product by category, sub-category, or options available with the product. This section lists all the possible scenarios for which a customer can search or view a product listing.

## To browse products

1. Create a cart using [Creating a Cart from a Quote](#).
2. Browse products by categories and subcategories using the search text entered by the user by utilizing [Retrieving Products and List Prices For a Price List Category and Search Text](#) and [Retrieving Products and List Prices For a Price List and Search Text](#). Using this API, the customer can search for a product directly or select a category and enter search text to refine the search further.
3. Browse for categories and option groups using [Retrieving Categories for a Price List](#) and [Retrieving Option Groups, Options, and List Prices for a Price List Product](#).
4. Browse for products using [Retrieving Products and List Prices for a Price List](#) and [Retrieving Products and List Prices for a Price List and Category](#).
5. Compare the displayed products using [Comparing Products](#).

## Configuring and Adding Products

In the product catalog, you can see the list of products that a vendor offers. Using this list, you can add a product to your cart, configure the product attributes, such as color and memory. If the product is a bundled product, you can select the options with which you

want to bundle your product. This section lists the scenarios using which the customer can view the configuration options for a product.

When you configure and add products to your cart, you can do the following:

- Show Product / Bundle Details
- Create Cart (Create Quote first and then Cart for the Quote)
- Configure Bundles
- Add Single Product to Cart
- Add Multiple Products to Cart
- Show Recommendations for Selected Product
- Show Validation and other messages
- Auto-Include Products
- Delete single product from Cart
- Delete Multiple products from Cart
- Apply any constraint rules related to Cart
- Remove any constraint rules for a deleted product.

## To add products and apply constraint rules in the cart

1. Create a cart using [createCart](#).
2. Add products and options to the cart using [addBundle](#) , [addMultiProducts](#), or [Adding Options to a Bundle](#).
3. Associate constraint rules for the added products, by querying the right set of rules, using [associateConstraintRules](#).
4. Apply constraint rules to the products in the cart using [applyConstraintRules](#).
5. View the results of applying the constraint rules using [getConstraintRuleResult](#). If the results return *NeedMoreProcessing*, then [applyConstraintRules](#) must be run again.

## To delete products and apply constraint rules for deleted line items

1. Delete products from the cart using [removeBundle](#) or [Removing Multiple Bundles from a Cart](#)
2. Remove constraint rules for a deleted product using [Applying Constraint Rules to Deleted Products](#)
3. View the results of removing the constraint rules using [getConstraintRuleResult](#).

## Displaying Recommendations

In the product catalog, you can see the list of products that a vendor offers. When the customer selects a product, you can display the list of recommended products on the same page. This section lists the scenarios using which the recommended products appear. To configure recommendations for a product use the following APIs:

### Scenario 1

After a customer searches a product, the customer selects a product by clicking the product name or image in the result view. The recommendations for that product along with product details should appear.

### Scenario 2

After a customer searches a product, the customer adds the product to the cart using the Add to Cart button. On the Cart Details page, the recommendations for that product along with product details should appear along with the cart line items.

For each recommended product, display the product image, name, description, and price.

## Updating Price

If you want to update price for a cart after a specific interval use the [Updating Price For A Cart API](#).

You can use the Update Price API for your cart in the following scenarios:

- When the quantity of a product is incremented or decremented.
- When an additional product is added to your cart.
- If any additional charges apply for the product added.
- If any Adjustment whether Markup or Discounts are applied to the cart.

## Viewing Price Break-up Details

If the customer wants to view the breakup of how the tiered pricing rules are applied for a given quantity / term of the line item use the [Price Breakup for a Cart or Specific Line Item](#).

The customer might want to see the breakup of how the tiered pricing rules are applied for a given quantity / term of the line item. For example:

There is a tiered pricing structure for a given quantity as follows:

- 0-20 - 5% Discount
- 21-30 - 10% Discount
- > 30 - 15% Discount

If the line item has 25 quantities then show the pricing breakup as

- Product 1 - List Price: \$10 Quantity: 20 Discount: 5% Net Price: \$190
- Product 1 - List Price: \$10 Quantity: 5 Discount: 10% Net Price: \$180

Tiered pricing is applicable in the following scenarios:

- Tiered Pricing for a Single Larger Tiers
  - Tiered Pricing defined in Price List
  - Tiered Pricing negotiated in the quote
  - Tiered pricing negotiated in pricing agreement
- Tiered Pricing for a Single Order leading to higher tier e.g. single order of quantity 25 in above case
- Tiered Pricing for new and add-on order leading to higher tier e.g. initial order of 15 and then add-on (increment) asset based order of 10 quantity
- Tiered for different price types
  - Usage Tiers
  - One time Charge Tiers
  - Recurring subscription tiers
- Ramps

The breakup details for each line item should display a cart id, line id, and price break up details, such as quantity tier, list price, base price, extended price, adjustment, and net price.

## Adding Price Ramps to a Cart

After you add a line item to a cart, this API enables you to add primary and secondary ramp line items for the line item. Once the ramp line items are created, you can also update the ramp line item details or delete ramp line item details using standard SOQL queries.

When you use CPQ out of the box, invoke the ramp using the red icon to the left of the primary line item.

		Standard Price	\$100.00	\$100.00	1/13/2015	2/12/2015	<input type="text" value="1"/>	\$100.00		\$100.00	--None--	<input type="text" value=""/>	\$100	\$100.00	0.0000%	0.0000%
---	---	----------------	----------	----------	-----------	-----------	--------------------------------	----------	--	----------	----------	-------------------------------	-------	----------	---------	---------

Once you click the ramp icon, the ramp dialog appears:

Ramp Level	List Price	Start Date	End Date	Quantity	Adjustment Type	Adjustment Amount	Status
+	1	\$100.00	1/13/2015	2/12/2015	1	--None--	✓

The ramp dialog allows you to add, edit dates and quantity, make adjustments, save the changes, and cancel the changes.

Once the customer adds a ramp to a cart line item, they can do the following:

- Edit the start date, end date, quantity, adjustment type and adjustment amount based on the custom setting.
- The start date of a ramp line item defaults to the end date+1 of the previous line item.
- The end date of a ramp line item defaults to a date such that the difference between the start date and end date is the same as that of the previous line item.
- The user can add more ramp line items after or in between the ramp line items.
- The user can remove the new line before saving by clicking on the icon in the right most column.

Use the addMultiProducts API to add products to the cart.

addMultiProducts

Parameters		
Name	Type	Description
request	CPQ.AddMultiProduct RequestDO	The request data object.

Request Data Object - CPQ.AddMultiProductRequestDO		
Field	Type	Description
CartId	ID	The Id of the cart.
SelectedProducts	List <CPQ.SelectedProductDO>	The list of selected product data objects.

Response Data Object - CPQ.AddMultiProductResponseDO		
Field	Type	Description
LineNumbers	List<Decimal>	The list of line numbers added to the cart.

Data Object - CPQ.SelectedProductDO		
Field	Type	Description
ProductId	ID	Id of the product bundle.
Quantity	Decimal	The bundle quantity.
SellingTerm	Decimal	The bundle selling term.
StartDate	Date	The start date. You should ensure you use the correct date format.
EndDate	Date	The end date.
Comments	String	Comments associated with the record.
CustomFields	List<String> CustomFields	List of custom fields created for your product.

Data Object - CPQ.SelectedProductDO		
Field	Type	Description
CustomData	Apttus_Config2_ _LinItem__c CustomData	This can be used to include the list of custom fields you have added to the product.

## Code Sample

The sample below enables you to add ramp line items after you have:

- Added Products to the cart using the AddMultiProducts APIs,
- Updated the Price for the added products using the updatePriceforCart API.
- Selected the products for which you want to add a ramp for.

Using the sample below you fetch the list of selected products for which you want to add a ramp. You also fetch the parameters for each of the selected products. For all the ramps you create, set PriceGroup as Price Ramp and PricingStatus as Pending. For a primary line item, set IsPrimaryLine\_\_c = true, IsPrimaryRampLine\_\_c = true, and PrimaryLineNumber\_\_c = 1.

```

1  public void createRampLineItems()
2      {
3          List<String> rampLineItems = new List<String>();
4
5          if(lstWrapItems.size() > 0)
6              {
7                  // Create a list of selected products for which you want to
create a ramp for
8                  for(LineItemWrapperClass objLineItemWrapperClass :
lstWrapItems)
9                      {
10                     if(objLineItemWrapperClass.Selected)
11                         {
12                             rampLineItems.add(objLineItemWrapperClass.Name);
13                         }
14                     }
15
16                     // Sort Ramp Line Items by name
17                     rampLineItems.sort();

```

```

18
19     Integer rampLineItemIndex = 1;
20
21     for(String rampLineItemName : rampLineItems)
22     {
23         // Get Line Item parameters for the selected products
24         Apttus_Config2__LineItem__c lineItem = [SELECT
Apttus_Config2__ItemSequence__c, Apttus_Config2__PricingStatus__c,
Apttus_Config2__PriceGroup__c,
25             Apttus_Config2__IsPrimaryRampLine__c,
Apttus_Config2__IsPrimaryLine__c, Apttus_Config2__LineNumber__c,
Apttus_Config2__PrimaryLineNumber__c from Apttus_Config2__LineItem__c
WHERE
26             Name=:rampLineItemName];
27
28         //Set the parameters for each of the line items
29         lineItem.Apttus_Config2__PriceGroup__c = 'Price Ramp';
30         lineItem.Apttus_Config2__PricingStatus__c = 'Pending';
31         lineItem.Apttus_Config2__LineNumber__c = 1;
32         lineItem.Apttus_Config2__PrimaryLineNumber__c = 1;
33         lineItem.Apttus_Config2__ItemSequence__c =
rampLineItemIndex;
34         //For a primary line item set the following
35         if(rampLineItemIndex == 1)
36         {
37             lineItem.Apttus_Config2__IsPrimaryLine__c = true;
38             lineItem.Apttus_Config2__IsPrimaryRampLine__c = true;
39         }
40
41         //For all secondary line items set the following
parameters
42         else
43         {
44             lineItem.Apttus_Config2__IsPrimaryLine__c = false;
45             lineItem.Apttus_Config2__IsPrimaryRampLine__c = false;
46         }
47
48         // Update Line Items
49         update lineItem;
50
51         rampLineItemIndex++;
52     }
53 }

```



```

54         else
55         {
56             ApexPages.addMessage(new
ApexPages.Message(ApexPages.severity.info, 'No line items available.));
57         }
58     }

```

## Updating Taxes and Shipping for an Order

After the user has added products to the cart, you can apply taxes to each of the cart line items and shipping charges for the entire order.

- Tax and shipping can be calculated for the Quotes (Estimated tax / Shipping amount), but will be more generally used with E-Commerce orders with Pay Now scenario as the actual tax and shipping amount.

After the user has added the products to the cart, the sequence of actions is as follows:

- Customer specifies the Ship-to location.
- Once the ship to account or location is specified at the header (line item level in case of multiple ship-to locations)level, invoke the APIs to calculate Shipping as well as tax.
- Once the shipping and tax charges are calculated, the total price, shipping, and tax amount is displayed for a customer to review.
- Customer will then specify Payment option and Place the order.

### Tax Calculation Workflow:

- Create Tax callback class.
- Implement tax interface *ITaxCallback2* in the callback class to call the tax engine to calculate the total tax for the cart (Quote / Order) and breakup for each line item.
- Invoke *computeTaxforCart* API to which we just pass the "CartId". The API will create the tax breakup line items as well as the total tax line for the entire cart.

### Shipping Callback Workflow:

- Create Shipping callback class.
- Implement shipping interface *IShippingCallback* in the callback class to call the shipping engine to calculate the total shipping charges for the cart (Quote / Order).
- Invoke *computeShippingForCart* to which we just pass the "CartId". The API will create the shipping charges for the entire cart.

- [Sample Callback for Calculating Shipping](#)  
This section comprises the sample callback class that enables you to apply an appropriate shipping charge to a cart.
- [Sample Callback for Calculating Taxes](#)  
This section comprises the sample callback class that enables you to apply an appropriate tax to a cart line item.
- [To apply taxes and shipping charges](#)

## Sample Callback for Calculating Shipping

This section comprises the sample callback class that enables you to apply an appropriate shipping charge to a cart.

Before you define the shipping callback ensure that you have set the following:

- If shipping charges change based on the ship to or shipping address of the account, ensure that the account associated with the proposal has both specified.  
In addition, you can add additional custom fields on the quote or order line item such as Shipping Speed to calculate the shipping charge.

The sample callback class comprises the following objects:

Data Object - Apttus_Config2.ShippingInput		
Field	Type	Description
ShippingAddress	Address	Shipping Address. The address for shipping charge calculation obtained from the "Ship To" account for the line item. Accordingly, Ship To Account should be populated on each line item by the implementation.
Item	SObject	Apttus_Config2__LineItem__c. The Cart Line item. This is the reference to the cart line items. Implementation can use this cart line item to get any additional custom fields that might be needed to calculate Shipping.

Data Object - Apttus_Config2.ShippingInput		
Field	Type	Description
Handback	Object	<p>Any Apex Object passed by the caller and passed back to the caller in the result. Handback is used only in case when the shipping needs to be calculated for an object other than Cart line item. For example, Billing Schedule line items.</p> <ul style="list-style-type: none"> <li>• For calculating shipping for cart , there is no need to pass the Handback object. It can be blank</li> <li>• For calculating the Shipping for object other than line items, "Item" field will be blank and the actual object will be passed as handback object to identify object for which shipping is calculated.</li> </ul>

Data Object - Apttus_Config2.CustomClass.ShippingResult		
Field	Type	Description
ShippingCharge	Decimal	Shipping amount total of all the cart line items. This is the total shipping amount for the line item. Line item is identified by the sequence in which the shipping result is added. This needs to be calculated and populated by the implementation.
Handback	Object	<p>Any Apex Object passed by the calling class and passed back to the calling class in the result.</p> <p>Handback is used only in case when the shipping charge needs to be calculated for an object other than Cart line item e.g. Billing Schedule line items.</p> <ul style="list-style-type: none"> <li>• For calculating shipping charges for a cart, there is no need to pass the Handback object. It can be blank</li> <li>• For calculating the shipping charges for objects other than line items, "Item" field will be blank and the actual object will be passed as handback object to identify the object for which shipping charges are calculated.</li> </ul>

Once you have set shipping to, shipping address or shipping speed to your quote or order, create a Shipping Callback class and specify the name of the callback class in **Custom Settings > Config Custom Classes > (Edit) Custom Classes > Shipping Callback Class**.

In the sample callback class below, we fetch and create a list of all the products added to the cart. Then we fetch the shipping charges applicable for a product using an SOQL query. If a corresponding shipping code and a valid shipping charge exist for a product in the order or cart, the total shipping amount for a cart is added directly.

## Sample Callback for Calculating Taxes

This section comprises the sample callback class that enables you to apply an appropriate tax to a cart line item.

Before you define the tax callback ensure that you have set the following:

- Go to the product price list item, click **Edit > Tax & Billing** and select the **Taxable?** checkbox. Also, ensure that the **Tax Inclusive?** checkbox is deselected. If it is selected, the product price specified includes the tax amount.
- From the **Tax & Billing** tab, specify the tax code applicable for a product using a lookup. The tax code maps to a code in an external system where various taxes, tax charges, tax types are mapped against a tax code.
- If a tax is applicable based on the billing or shipping address of the account, ensure that the account associated with the proposal has both specified.

The sample callback class comprises the following objects:

Data Object - Apttus_Config2.TaxInput		
Field	Type	Description
TaxCode	String	Tax code for the line item as inherited from the Price list item or overridden at the line item level.
TaxAddress	Address	Tax Address.  Address as obtained for the line item based on precedence hierarchy specified. For example, Line Item Location ==> Line Item Ship To ==> Line Item Bill To ==> Cart Header Ship To ==> Cart Header Bill To ==> Cart Header Sold to

Data Object - Apttus_Config2.TaxInput		
Field	Type	Description
TaxableAmount	Decimal	Net Price from the Line Item in case the tax needs to be calculated on the Net Price. In case the tax needs to be calculated based on List or Base Price then the total amount calculated based on list or base price needs to be used from the line item to calculate tax.
Item	SObject	Apttus_Config2__LineItem__c. This is the reference to the cart line items. Implementation can use this cart line item to get any additional custom fields that might be needed to calculate tax.
Handback	Object	Any Apex Object passed by the caller and passed back to the caller in the result. Handback is used only in case when the tax needs to be calculated for an object other than Cart line item e.g. Billing Schedule line items. For calculating tax for cart line items, there is no need to pass the Handback object. It can be blank. For calculating the tax for object other than line items, "Item" field will be blank and the actual object will be passed as handback object to identify object for which tax is calculated.

Data Object - Apttus_Config2.CustomClass.TaxResult		
Field	Type	Description
TaxAmount	Decimal	This is the total amount for the line item. This is sum of the breakup tax amounts for the line item. Line item is identified by the sequence in which the tax result is added. This needs to be calculated and populated by the implementation.

Data Object - Apttus_Config2.CustomClass.TaxResult		
Field	Type	Description
TaxBreakups	List<Apttus_Config2__TaxBreakup>	Tax breakup provide the different components of the tax for a given line item. This needs to be implemented and provided by the implementation. Tax breakup has structure specified below.
Handback	Object	<p>Any Apex Object passed by the calling class and passed back to the calling class in the result.</p> <p>Handback is used only in case when the tax needs to be calculated for an object other than Cart line item e.g. Billing Schedule line items.</p> <ul style="list-style-type: none"> <li>• For calculating tax for cart line items, there is no need to pass the Handback object. It can be blank</li> <li>• For calculating the tax for object other than line items, "Item" field will be blank and the actual object will be passed as handback object to identify object for which tax is calculated.</li> </ul>

Data Object - Apttus_Config2__TaxBreakup		
Field	Type	Description
TaxType	String	Type of tax to be applied. For example, sales, custom, excise
TaxRate	Decimal	Tax percentage to be applied
TaxAppliesTo	String	The field on which the tax is applied to. For example, net price.

Data Object - Apttus_Config2__TaxBreakup		
Field	Type	Description
TaxAmount	Decimal	The tax amount calculated based on the formulae defined in the callback.
BreakupType	String	If you specify the breakup type as Detail, then the quote or order line items show the detailed tax break up for a cart line item.

**Tax Breakup Type Example**

Breakup Type	Tax Type	Tax Rate	Tax Applies to	Tax Amount
Detail	State Tax	10	Net Price	100
Detail	City Tax	1	Net Price	10
Total				110

Data Object - Apttus_Config2. CustomClass.Address	
Field	Type
Street	String
City	String
State	String
County	String
PostalCode	String
Country	String

Once you have set a Tax Code for each price list item, create a Tax Callback class and specify the name of the callback class in **Custom Settings > Config Custom Classes > (Edit) Custom Classes > Tax Callback Class**.

In the sample callback class below, we fetch and create a list of all the products added to the cart. Then we fetch the tax charges applicable for a product using an SOQL query. If a corresponding tax code and a valid tax charge exists for a product in the order or quote line item, the total tax amount for a product is calculated using the formula-  $\text{taxAmount} += (\text{taxRate} \neq \text{null} ? (\text{taxRate} * \text{input.TaxableAmount}) / 100 : 0)$ . After the total tax on a product is calculated, you can list the tax break up based on Tax types applied for each of the order or quote line items. The sample callback lists the parameters applicable for a tax break up. If you want to apply tax to shipping charges as well, you can capture shipping amount for each line item in a custom field and use the total amount (Net Price + Shipping Amount) from the line item to calculate tax. In this case shipping should be calculated first and then tax.

In the sample callback class, tax is calculated using hardcoded rates. For actual implementations, tax rates and tax amounts are fetched from external Tax calculation engines such as Avalara and Vertex to calculate tax.

```

1  /**
2   * Apttus Config & Pricing
3   * QATaxCallBack
4   *
5   * @2013-2014 Apttus Inc. All rights reserved.
6   */
7  global with sharing class QATaxCallBack implements
8     Apttus_Config2.CustomClass.ITaxCallback2 {
9
10     /**
11     * Callback invoked to compute tax based on the given input
12     * @param input the tax input
13     * @return the tax result
14     */
15     global Apttus_Config2.CustomClass.TaxResult
16     computeTax(Apttus_Config2.CustomClass.TaxInput input) {
17         // Compute Tax
18         system.debug('Entering into computeTax');

```



```

17     List<Apttus_Config2.CustomClass.TaxResult> results =
computeTaxMultiple(new Apttus_Config2.CustomClass.TaxInput[]{input});
18     return (!results.isEmpty() ? results[0] : new
Apttus_Config2.CustomClass.TaxResult());
19 }
20
21 /**
22  * Callback invoked to compute tax based on the given list of inputs
23  * @param inputs the list of tax inputs
24  * @return the list of tax results
25  */
26 global List<Apttus_Config2.CustomClass.TaxResult>
computeTaxMultiple(List<Apttus_Config2.CustomClass.TaxInput> inputs) {
27
28     List<Apttus_Config2.CustomClass.TaxResult> results = new
List<Apttus_Config2.CustomClass.TaxResult>();
29
30     for (Apttus_Config2.CustomClass.TaxInput input : inputs) {
31
32         Decimal taxAmount = 0;
33
34         Apttus_Config2.CustomClass.TaxResult result = new
Apttus_Config2.CustomClass.TaxResult();
35
36         List<Apttus_Config2__TaxBreakup__c> taxBreakUps = new
List<Apttus_Config2__TaxBreakup__c>();
37
38         // Add the handback object to correlate the result with the
input
39         result.Handback = input.Handback;
40
41         // Compute tax based on tax code and tax address
42
43         // Tax Code is required
44         if (input.TaxCode == null || input.TaxCode.trim().length() ==
0 || input.TaxableAmount == null || input.TaxableAmount == 0) {
45
46             // Tax Amount
47             result.TaxAmount = 0;
48         } else {
49
50             List<QA_Tax_Rate__c> accountTaxRates = new
List<QA_Tax_Rate__c>();
51
52             SObject sObj = input.Item;

```

```

53         Id taxCodeId = (Id)
sObj.get('Apttus_Config2__TaxCodeId__c');
54
55         accountTaxRates = [SELECT Id, Name, Tax_Applies_To__c,
Tax_Rate__c, Tax_Type__c
56         FROM QA_Tax_Rate__c
57         WHERE Tax_Code__c =: taxCodeId];
58
59         for(QA_Tax_Rate__c accountTaxRate : accountTaxRates) {
60
61             Decimal taxRate = accountTaxRate.Tax_Rate__c;
62             String taxAppliesTo =
accountTaxRate.Tax_Applies_To__c;
63             String taxType = accountTaxRate.Tax_Type__c;
64
65             Apttus_Config2.CustomClass.Address addr =
input.TaxAddress;
66
67             // Calculate Tax Amount
68             taxAmount += (taxRate != null ? (taxRate *
input.TaxableAmount) / 100 : 0);
69
70             // Calculate Tax BreakUps
71             Apttus_Config2__TaxBreakup__c taxBreakUp = new
Apttus_Config2__TaxBreakup__c();
72             taxBreakUp.Apttus_Config2__TaxType__c = taxType;
73             taxBreakUp.Apttus_Config2__TaxRate__c = taxRate;
74             taxBreakUp.Apttus_Config2__TaxAppliesTo__c =
taxAppliesTo;
75             taxBreakUp.Apttus_Config2__TaxAmount__c = taxRate *
input.TaxableAmount / 100;
76             taxBreakUp.Apttus_Config2__BreakupType__c = 'Detail';
77
78             taxBreakUps.add(taxBreakUp);
79         }
80
81
82         result.TaxBreakups = taxBreakUps;
83         result.TaxAmount = taxAmount;
84     }
85     results.add(result);
86 }
87 return results;

```

```

88     }
89   }

```

## To apply taxes and shipping charges

1. Create a cart using the [Creating a Cart from a Quote](#).
2. Browse products by categories and subcategories using the search text entered by the user by utilizing [Retrieving Products and List Prices For a Price List Category and Search Text](#) and [Retrieving Products and List Prices For a Price List and Search Text](#). Using this API, the customer can search for a product directly or select a category and enter search text to refine the search further.
3. Browse for categories and option groups using [Retrieving Categories for a Price List and Retrieving Option Groups, Options, and List Prices for a Price List Product](#).
4. Browse for products using [Retrieving Products and List Prices for a Price List](#) and [Retrieving Products and List Prices for a Price List and Category](#).
5. Compare the displayed products using [Comparing Products](#).
6. Add products to the cart by using [Adding Products to a Cart](#) and [Adding a Bundle to a Cart](#).
7. Update the price using [Updating Price For A Cart](#).
8. Finalize the cart using [Finalizing a Cart](#).
9. Define the Tax and Shipping Callbacks using [Sample Callback for Calculating Taxes](#) and [Sample Callback for Calculating Shipping](#).
10. Specify the Callback class names at **Custom Settings > Config Custom Classes (Edit) > Tax Callback and Shipping Callback**.
11. Apply the taxes and shipping charges using [Computing Taxes for Cart Line Items](#) and [Computing Shipping for Cart Line Items](#). The Tax API creates and populates - the tax breakup for each applicable line item in the cart and the total Tax line and related breakup. The shipping API calculates the shipping amount for the entire order.
12. Update the Order Line Items using [Synchronizing a Cart](#).

## Creating and Updating Quote and Quote Line Items

A customer with a valid account can create a quote from an existing opportunity. Using the Quote ID and a valid price list, you can create a cart from scratch and by browsing the product catalog you can navigate for a product and select the required product manually or can search for a product. You can enable searches using the search text field in the cart page and enable the user to search for a desired product by category, sub-category, or

options available with the product. This section lists all the possible scenarios for which a customer can create a quote and or view a product listing.

## To create and update quote and quote line items

1. Create a cart using the [Creating a Cart from a Quote](#).
2. Browse products by categories and subcategories using the search text entered by the user by utilizing [Retrieving Products and List Prices For a Price List Category and Search Text](#) and [Retrieving Products and List Prices For a Price List and Search Text](#). Using this API, the customer can search for a product directly or select a category and enter search text to refine the search further.
3. Browse for categories and option groups using [Retrieving Categories for a Price List](#) and [Retrieving Option Groups, Options, and List Prices for a Price List Product](#).
4. Browse for products using [Retrieving Products and List Prices for a Price List](#) and [Retrieving Products and List Prices for a Price List and Category](#).
5. Compare the displayed products using [Comparing Products](#).
6. Add products to the cart by using [Adding Products to a Cart](#) and [Adding a Bundle to a Cart](#).
7. Update the price using [Updating Price For A Cart](#).
8. Finalize the cart using [Finalizing a Cart](#).
9. Update the Quote Line Items using [Synchronizing a Cart](#).

## Creating and Updating Order and Order Line Items

A customer with a valid account can create an order from an existing opportunity. Using the order ID and a valid price list, you can create a cart from scratch and by browsing the product catalog you can navigate for a product and select the required product manually or can search for a product. You can enable searches using the search text field in the cart page and enable the user to search for a desired product by category, sub-category, or options available with the product. This section lists all the possible scenarios for which a customer can create a order and or view a product listing.

## To create and update order and order line items

1. Create an order using [Creating an Order](#)
2. Create a cart for an order using [Creating a Cart for an Order](#).

3. Browse products by categories and subcategories using the search text entered by the user by utilizing [Retrieving Products and List Prices For a Price List Category and Search Text](#) and [Retrieving Products and List Prices For a Price List and Search Text](#). Using this API, the customer can search for a product directly or select a category and enter search text to refine the search further.
4. Browse for categories and option groups using [Retrieving Categories for a Price List and Retrieving Option Groups, Options, and List Prices for a Price List Product](#).
5. Browse for products using [Retrieving Products and List Prices for a Price List](#) and [Retrieving Products and List Prices for a Price List and Category](#).
6. Compare the displayed products using [Comparing Products](#).
7. Add products to the cart by using [Adding Products to a Cart](#) and [Adding a Bundle to a Cart](#).
8. Update the price using [Updating Price For A Cart](#).
9. Update the order line items using [Synchronizing a Cart to an Order](#).
10. Create Asset Line Items using [Create Asset Line Items for an Order](#).  
For more information on the Orders APIs, see [Order Web Service](#).

## Asset Based Ordering

The end user or a renewals manager can view and modify products that the organization owns and should be able to filter the assets to renew orders. Display and modify the assets that belong to a company based on the following criteria:

- All assets with recurring, usage, and one-time charge types for which renewal date is approaching.
- All licenses that expire within 30, 60, or 90 days.
- Expired licenses.
- All services that expire in 30 days.
- All assets that are in the End of Life or End of Sale state by the vendor.
- All services that are no longer active or are cancelled should be displayed.
- All software licenses setup for a site.
- All products under the CPQ product category.

The assets should be filtered and displayed based on asset status, price type, and subscription expiration.

The displayed results of the asset list the product name, ID, asset name, asset ID, SKU ID, quantity, price type, price UOM, purchase date, start and end date, base price, adjustment, net price, asset status, and renewal term.

## To view assets

Retrieve asset results that fulfill the criteria using the [Retrieving Asset Line Items](#) API. Using this API, you can fetch the assets.

## To modify assets

1. Retrieve asset results that fulfill the criteria using the [Retrieving Asset Line Items](#) API. Using this API, you can fetch the assets.
2. You can renew, amend, increment, or terminate the assets using the code snippets for [Modifying Assets \(Deprecated\)](#).

## To create assets from orders

Create assets from an order using the Create Asset Line Items for an Order API. Using this API, you can create assets.

## Finalizing the Cart

The customer can finalize the cart in the following scenarios:

- All the required products have been added.
- All requisite prices have been adjusted accordingly.

Invoke the [Finalizing the Cart](#) to enable the user to finalize the cart.

## Updating Quote Terms to Modify Dates

For a finalized quote, a customer can choose to apply add-ons. The term, validity period, and pricing have to be recalculated after the quote is finalized, approved, and a subsequent proposal is generated. Use the [Updating Quote Terms](#) to achieve the scenarios highlighted.

The *updateQuoteTerms* API behavior is as follows:

- Update the carts in Finalized or Ready for Finalization status. The latest cart from the quote in any of the two statuses are used as basis to updating the quote.
- Create a new version of the cart in case of Finalized cart and supersede the current version of the cart. The status of the new version of the cart will be same as the status of the current version of the cart. If the cart was in Finalized status, the new cart will also be in finalized status and system will update the "Finalized date" on the quote header.
- Auto-Synch cart line items with the Proposal Line Items.

## To synchronize cart and update quote

1. Finalize the cart using the [Finalizing the Cart](#) API.
2. Synchronize the finalized cart with the quote using the [Synchronize](#) API.
3. Update the selling terms of the products associated to the quote using the [Update Quote Terms](#) API.

# Troubleshooting CPQ SOAP APIs

Refer to this topic for descriptions of API error messages thrown by Salesforce and suggested actions to take. An exception is an error caused during code execution. There are a number of built-in Apex exceptions as well as custom exceptions provided by Conga.

The following exceptions that are thrown by Salesforce that can be handled by Conga and converted to custom error messages:

- System.Exception: Exception thrown under most circumstances.
- System.CalloutException: Exception thrown when communicating with external systems.
- DMLException: Exception thrown when performing CRUD ('Create', 'Read', 'Update', 'Delete') actions.

Refer to the following table for a list of custom error messages provided by Conga for some SOAP APIs, included suggested actions to take to resolve the exception.

API	Error Message	Suggested Action
Apttus_API.CPQWebService.addCustomBundle	You do not have the level of access necessary to perform the operation you requested. Contact the owner of the record or your administrator for access.	Authentication or access (system permissions) issue. Contact an administrator or check the appropriate topic for authentication instructions.
Apttus_API.CPQWebService.addMultiProducts		
Apttus_API.CPQWebService.abandonCart		
Apttus_API.CPQWebService.removeMultiBundles		
Apttus_API.CPQWebService.removeOptions		
Apttus_Config2.ConstraintWebService2.associateConstraintRulesForProducts		
Apttus_Config2.ConstraintWebService2.applyConstraintRulesOnAddBundle		



API	Error Message	Suggested Action
Apttus_Config2.CPQAdmin WebService.createProducts		
Apttus_Config2.CPQAdmin WebService.createCategories		
Apttus_Config2.CPQAdmin WebService.associateProductToCategory		
Apttus_Config2.CPQAdmin WebService.removeProductFromHierarchy		
Apttus_Config2.CPQWebService. createBundleLineItems		
Apttus_Config2.CPQWebService. createProductLineItems		
Apttus_Config2.CPQWebService. finalizeConfiguration		
Apttus_Proposal.Proposal WebService.createProposalFromAccount		
Apttus_Proposal.Proposal WebService.createProposalFromOpportunity		
Apttus_QPConfig.QPConfig WebService.cloneProposal		

API	Error Message	Suggested Action
Apttus_Config2.ConstraintWebService2.applyConstraintRulesOnAddAll	You do not have the level of access necessary to perform the operation you requested. Contact the owner of the record or you administrator for access.	Authentication or access (system permissions) issue. Contact an administrator or check the appropriate topic for authentication instructions.
	Too many products found constraint rule actionName.	You must reduce the number of products associated with constraint rule actions.
Apttus_Config2.CPQAdminWebService.buildHierarchy	You do not have the level of access necessary to perform the operation you requested. Contact the owner of the record or your administrator for access.	Authentication or access (system permissions) issue. Contact an administrator or check the appropriate topic for authentication instructions.
	The category is currently associated as an Option Group to Product(s).	This is an invalid input. You must provide only category hierarchies and not option groups.
Apttus_Config2.CPQWebService.AbandonConfiguration	You do not have the level of access necessary to perform the operation you requested. Contact the owner of the record or your administrator for access.	Authentication or access (system permissions) issue. Contact an administrator or check the appropriate topic for authentication instructions.
	Error encountered in callback processing.	Check that the appropriate callback is being processed for this call.

The following are standard Salesforce exceptions that cannot be handled or changed:

- LimitException
- UnexpectedException
- AssertException
- NoAccessException
- NoDataFoundException
- SerializationException
- VisualforceException

You can learn more about these exceptions and recommended actions to take by consulting [Salesforce documentation](#).

## **Conga Copyright and Disclaimer**

Copyright © 2024 Conga Corporation (“Conga”) and/or its affiliates. All rights reserved.

No part of this document, or any information linked to or referenced herein, may be reproduced, distributed, or transmitted in any form or by any means, including photocopying, recording, or other electronic or mechanical methods, without the prior written consent of Conga. All information contained herein is subject to change without notice and is not warranted to be error free.

This document may describe certain features and functionality of software that Conga makes available for use 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, in any form, or by any means, use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part of the software. Reverse engineering, disassembly, decompilation of, or the creation of derivative work(s) from, the software is strictly prohibited. Additionally, this document may contain descriptions of software modules that are optional and for which you may not have purchased a license. As a result, your specific software solution and/or implementation may differ from those described in this document.

U.S. GOVERNMENT END USERS: Conga software, including any operating system(s), 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.

Neither the software nor the documentation were 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. Conga and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Apttus, AI Analyze, Conga, Conga AI, Conga AI Discover, Conga Batch, Conga Collaborate, Conga Composer, Conga Conductor, Conga Connect, Conga Courier, Conga Grid, Conga Mail Merge, Conga Merge, Conga Orchestrate, Conga Sign, Conga Trigger, Digital Document Transformation, True-Up, and X-Author are registered trademarks of Conga and/or its affiliates.

The documentation and/or software may provide links to web sites and access to content, products, and services from third parties. Conga is not responsible for the availability of, or any content provided by third parties. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Conga is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Conga is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

For additional resources and support, please visit <https://community.conga.com>.