



An Oracle White Paper
November, 2014

Commerce-CPQ Integration

Disclaimer

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Executive Overview	2
Integration Overview.....	3
Prerequisites	4
Installation	4
Configuration	8

Executive Overview

Oracle provides a range of industry leading products to help organizations sell to and service their customers, while providing exceptional customer experience. Oracle views customer experience in terms of a never-ending cycle of assisting customers to research and purchase items or services, and then providing support to those customers. As shown in the following diagram, the cycle of attracting, selling, servicing, and retaining customers is an ongoing process and requires consistent interaction with the customer base.



Two key elements of the Oracle solution are Oracle Commerce and Oracle Configure Price Quote (CPQ). Oracle Commerce is an industry leading commerce solution that orchestrates the user experience during the research, selection, and purchase process. CPQ provides industry leading configuration, pricing and quoting functions to merchants.

The integration discussed in this white paper brings together the capabilities of Oracle Commerce and Oracle CPQ to provide a unified solution for businesses to sell complex products and services to their customers. Commerce provides the overall customer experience, allowing merchants to provide the most relevant content to shoppers at all times. CPQ provides the ability to let shoppers configure complex items to ensure order accuracy, and to process request for quotes.

This release of the integration provides functionality around quoting. Support for integration of Commerce and CPQ to support configuration of products is planned for the future releases.

When used together, Commerce and CPQ provide a powerful quote request and management solution for merchants that require such functionality. Commerce acts as the purchaser interface, and allows the purchaser to create a Request For Quote (RFQ) based on browsing the commerce catalog, searching for products, etc. That RFQ is then passed to CPQ for processing by appropriate sales personnel. The RFQ can be modified, have prices updated, and terms such as validity dates may be added. Similarly, CPQ allows the RFQ to go through appropriate workflow and approvals for the merchant.

Once the quote is finalized, it is returned to Commerce, where it can be viewed by the purchaser, resubmitted with additional comments, or converted into a completed order. Alternatively, the quote can be followed up on directly with the end customer by direct sales and they can turn the quote into an order in CPQ. Once the quote is submitted, either via Commerce or via CPQ, the other system is notified and the status is updated.

Combined CPQ and Oracle Commerce provide a powerful solution for managing the quoting process, quotes and the customer experience for shoppers. The combination allows many tasks to be completed by the purchaser directly, reducing the effort on the direct sales team.

Integration Overview

This Integration mainly targets to extensively use the quoting features of Oracle CPQ within the Oracle Commerce Product. It also includes a set of sample pages developed on the Motorprise reference application of Commerce. This Motorprise application related UI files like .js, .css, html etc comes along with this Reference Integration, hence the customer can use them or modify the files as per their requirements or completely create their own UI for implementation.

The REST APIs exposed by commerce for this integration are **getSessionConfirmationNumber** service, **login** service and **completeQuote** service. Likewise CPQ exposes SOAP webservice namely **login**, **createTransaction** and **updateTransaction** services.

The **getSessionConfirmationNumber** service provides a unique session number that prevents any malware entry into the Commerce system. This session confirmation number is added in the request header in all the succeeding webservice calls.

The **login** service is called to login to the Commerce Motorprise Application with the external user. This external user should be assigned 'CPQ Integration Role' in BCC application of Commerce.

The **completeQuote** webservice exposed by Commerce mainly focuses on synching the quote details from CPQ system into Commerce. This webservice will be called each time the quote/order status is changed in CPQ.

By calling the CPQ **login** service Commerce can login to the CPQ system. The **createTransaction** service of CPQ allows Commerce to submit a request for quote. This service expects the necessary information about the product and part numbers for which the quote is requested. This service returns the 'Transaction Id' to denote the transaction of the quote.

The CPQ **updateTransaction** service is called to complete the quote requesting process by passing the transaction id obtained from above call, shipping details, billing details, etc. into the CPQ system.

Prerequisites

The CPQ Reference Integration module has been tested with Oracle Commerce version 11.1. Below are the prerequisites for installation.

- Oracle CPQ Cloud 2014 R2
- Refer to the CPQ documentation for the CPQ server details

Installation

The CPQ Reference Integration module is distributed as a ZIP package and can be downloaded from <https://edelivery.oracle.com>. Please follow below steps for the complete installation.

- 1) Install Oracle Commerce 11.1
 - Run the respective REL build for Oracle Commerce Platform 11.1 (.exe for Windows and .bin for Linux)
 - Choose Locale and followed by Next.
 - Accept the License Agreement and click "Next"
 - Choose the folder to install OC Platform and click "Next"
 - Click "Select All" then click "Next"
 - Choose "Oracle WebLogic" and click "Next"
 - Choose the folders where WLS and JDK are installed and click "Next"
 - Click "Install"
- 2) CPQ Reference Integration build
 - Create a CPQ folder in DYNAMO_ROOT i.e. /work/service/ATG/ATG11.1
 - Copy the CPQ zip file downloaded from oracle support site in the same CPQ folder.
 - Check DYNAMO_HOME variable is correct which is DYNAMO_HOME= /work/service/ATG/ATG11.1/home
 - Unzip the contents of Oracle Commerce CPQ Integration 11.1 .zip file into the folder.
- 3) Configure via CIM
 - a) Run CIM
 - Navigate to Oracle Commerce home folder (e.g. "/work/service/ATG/ATG11.1/home").
 - Navigate to the "/bin" subfolder and run ". /cim.sh"
 - b) Set the Administrator Password (e.g. Admin123)
 - c) Product selection:
 - The following products should be selected (NOTE: selected items are marked with '* '):
 - *[1] Oracle Commerce Platform -
Includes, optionally, data warehouse components
 - *[2] Oracle Commerce Site Administration -
Includes Commerce Platform and Content Administration
 - [3] Oracle Commerce Platform-Guided Search Integration -
Includes Commerce Platform. Select this option when Commerce Guided Search is used.
 - *[4] Oracle Commerce Core Commerce -
Includes Commerce Platform and Content Administration. Optional: data warehouse components, Preview and Merchandising UI
 - [5] Oracle Commerce WebCenter Sites Extensions -
Includes Commerce Platform and Commerce CAS Reader.
 - *[6] Content Administration -
Includes Oracle Commerce Platform. Optional: Preview

- [7] Oracle Commerce CAS Reader -
Includes Commerce Platform. Select this option when Commerce Platform is used to import data to Commerce CAS.
- *[8] Oracle Commerce REST -
RESTful Web Services
- [D] Done
- d) Do not Choose any Add-ons:
- [1] Reporting
 - [2] Staging Server
 - [3] Dedicated Lock Servers
 - [4] Single Sign On (SSO)
 - [5] Preview Server
 - [6] Merchandising UI
 - [7] Abandoned Order Services
 - [D] Done
- Select zero or more > D
- e) Include Demo Application:
- [1] Quincy Funds Demo
 - *[2] MotorPriseJSP
 - [D] Done
- Select zero or one > D
- f) Choose data source
- *[1] Non-Switching Data source
 - [2] Switching Data source
- g) Select application server
- Oracle Weblogic Online (Either Production or Developer Mode)
- Enter path to WLS folder, e.g. "/work/service/Oracle/Middleware/Oracle_Home/wlserver" and then domain path
"/work/service/Oracle/Middleware/Oracle_Home/user_projects/domains/base_domain"
- Enter URL to admin server, e.g. "t3://localhost:7001"
- Enter username/password to admin server, such as: weblogic/weblogic1
- Reenter password and Perform Validation. Validation should be successful, WLS is up and running.
- h) This guide will use Oracle DB. We should create 2 schemas: publishing and production which should be created before database configuration process starts.

DATABASE CONFIGURATION

Publishing (for example)

Enter User Name > pkt5CPQ_Pub

Enter Password > *****

Re-enter the password > *****

Enter Host Name > hostname.us.company.com

Enter Port Number [[1521]] >

Enter Database Name > utf8112

Enter Database URL

[[jdbc:oracle:thin:@hostname.us.company.com:1521:utf8112]]>

Enter Driver Path > /work/service/download/instantclient_11_2/ojdbc6.jar

Enter JNDI Name [[ATGPublishingDS]] >

Connection to database is successful

Create the schema for it

Import the data by giving the password for Merchandising & Publishing admin User

Importing (1 of 5) /CIM/tmp/import/management-import1.xml:

/DAS/install/data/dynAdminRepo.xml to /atg/dynamo/security/AdminSqlRepository

/DPS/InternalUsers/install/data/das-security.xml to /atg/userprofiling/

InternalProfileRepository

Later it will show “all imports completed successfully.”

```

Production (for example)
Enter User Name > pkt5CPQ_Prod
Enter Password > *****
Re-enter the password > *****
Enter Host Name > hostname.us.company.com
Enter Port Number [[1521]] >
Enter Database Name > utf8112
Enter Database URL
[[jdbc:oracle:thin:@hostname.us.company.com:1521:utf8112]]>
Enter Driver Path > /work/service/download/instantclient_11_2/ojdbc6.jar
Enter JNDI Name [[ATGProductionDS]] >
Connection to database is successful
Create the schema for it & followed by importing data.

```

```

Importing ( 1 of 3 ) /CIM/tmp/import/nonswitchingCore-import1.xml:
/DAS/install/data/dynAdminRepo.xml to /atg/dynamo/security/AdminSqlRepository
/DCS/install/data/returnData.xml to /atg/commerce/custsvc/CsrRepository

```

Later it will show “all imports completed successfully.”

- i) After the completion of database configuration of both production and publishing server, finish the OPSS security and later followed by Server Instance Configuration which is elaborated below.

SERVICE INSTANCE CONFIGURATION

Configure publishing server:

Publishing Server General Configuration – REQUIRED (for example :)

Enter Production Lock Server Hostname > hostname.us.company.com

Enter Production Lock Server Port [[9012]] >

Instance Management – REQUIRED

Add Server Instance

Publishing with a Server Lock Manager: Minimum 1 Required

Enter Server Instance Name: atg_publishing_lockserver

Choose Default Port Binding port

Enter HTTP Port [[7003]] >

Enter HTTPS Port [[7004]] >

Enter Site HTTP Port [[7003]] >

Enter RMI Port [[8860]] >

Enter DRP Port [[8850]] >

Enter File Deployment Port [[8810]] >

Enter File Synchronization Deploy Server Port [[8815]] >

Enter Lock Server Port [[9010]] >

Configure production server:

Production Server General Configuration – REQUIRED (for example:)

Enter Default MDEX Host Name [[localhost]] >

Enter Default MDEX Port Number [[15000]] >

Instance Management - REQUIRED

Add Server Instance

Production with a Server Lock Manager: Minimum 1 Required

Enter Server Instance Name: atg_production_lockserver

Use Custom Port Binding. Ports 01

Enter HTTP Port [[7103]] >

Enter HTTPS Port [[7104]] >


```

Enter Site HTTP Port [[7103]] >
Enter RMI Port [[8861]] >
Enter DRP Port [[8851]] >
Enter File Deployment Port [[8811]] >
Enter Lock Server Port [[9012]] >

```

This will complete the configuration of the 2 Instances of both production and publishing respectively.

- j) Application Assembly & Deployment
Accept default values on all steps.

- k) Quit CIM

- 4) Verify server instances created

Navigate to Oracle Commerce home folder (e.g. "/work/service/ATG/ATG11.1/home")

Navigate to the "/servers" subfolder. There are 2 servers we've just created:

```

atg_production_lockserver
atg_publishing_lockserver

```

- 5) CPQ data addition.

Execute all the sql files under CPQ/sql/db_components/<dbname> in production db of Oracle Commerce. <dbname> has four options namely db2, mssql, mysql and oracle.

Go to the location /work/service/ATG/ATG11.1/CPQ/sql/db_components/oracle/

After logging into production database execute the below scripts

```

sqlplus pkt6CPQ_Prod/pkt6CPQ_Prod@utf8112.hostname.us.company.com

```

```

SQL> @cpq_b2b_multisite_ddl.sql

```

```

Table created.

```

```

SQL> @cpq_order_ddl.sql

```

```

Table created.

```

```

Table created.

```

```

Table created.

```

```

Table created.

```

```

SQL> @cpq_user_ddl.sql

```

```

Table created.

```

```

SQL> exit

```

- 6) Rebuilding production EAR file with CPQ module & its deployment.

a) Log into the Weblogic server ui which is http://<ATG hostname>:<port>/console)

b) Go to 'Environment' -> 'Deployments' and delete atg_production_lockserver.ear

c) Later go to this location /work/service/ATG/ATG11.1/home/cimEars & delete atg_production_lockserver.ear.

d) Now in order to create a new ear file give the below run assembler command from below location /work/service/ATG/ATG11.1/home/bin

```

./runAssembler -server "atg_production_lockserver"

```

```

"/work/service/ATG/ATG11.1/home/./home/cimEars/atg_production_lockserver.ear" -m MotorpriseJSP
DafEar.Admin DPS DSS ContentMgmt DCS.PublishingAgent REST PublishingAgent CPQ

```

- e) Go to ' /work/service/ATG/ATG10.2/home/cimEars' and the newly built production ear file should be there
- f) Go back to the Weblogic server & go to 'Environment' -> 'Deployments'
- g) Add the new production ear files e.g. add 'atg_production_lockserver.ear' as follows:

- Click on the 'Install' button
- Ensure the Path is set to /work/service/ATG/ATG11.1/home/cimEars , click on the 'atg_production_lockserver.ear' checkbox then click on the 'Next' button
- 'Choose targeting style' leave as the default selected option 'Install this deployment as an application' and click on the 'Next' button
- 'Select deployment targets' screen click on the 'atg_production_lockserver' checkbox and then on the 'Next' button
- Under 'General' add '.ear' to the end of 'atg_production_lockserver' so it looks like 'atg_production_lockserver.ear' and at the bottom of the page click on the 'I will make the deployment accessible from the following location' checkbox then click on the 'Finish' button. UI should update with 'atg_production_lockserver.ear' displayed.

h) Starting the servers.

Go to /work/service/Oracle/Middleware/Oracle_Home/user_projects/domains/base_domain/bin/ & hit the below commands to run production and publishing servers respectively.

```
./startManagedWebLogic.sh atg_production_lockserver
./startManagedWebLogic.sh atg_publishing_lockserver
```

After production server starts successfully, refer the below url which should open Motorprise application which assures the successful installation of CPQ build.

<http://<hostname>.us.company.com:<Prod-http-port>/cpqIntegration>

Configuration

1) Configuring CPQ server

Create a Configuration.properties which should have the below content and place the file in the below locations after creating 'integration' and 'cpq' folders respectively.

```
/work/service/ATG/ATG11.1/home/servers/atg_production_lockserver/localconfig/atg/integration/cpq
/work/service/ATG/ATG11.1/home/localconfig/atg/integration/cpq
```

```
#the bmi sever
environmentName=slc02xdh
domainName=us.oracle.com
cpqEndPoint=http://slc02xdh.us.oracle.com/
cpqApp=v1_0/receiver
#the login app
loginApp=v1_0/receiver
soapLoginProtocol=http
userName=Commerce_Dev
password=Welcome
```

2) Permissions for OC Motorprise user

Go to OC Business Control Center by referring the below url and click on External Users in the Commerce Merchandising section.

<http://<hostname>.us.company.com:<Pub-http-port>/atg/bcc>

Under External user go to "ROLES" and create a new role named as "CPQIntegrationUser" and followed by hitting SAVE button.

Now Go to "USERS" and search for the respective Motorprise user for whom you want to give the permissions, afterwards assign the above role to that respective Motorprise user and save it. Later go to the Billing & Shipping address of the same user under Commerce tab and give the First and last name of the user to make sure that these details reflect in CPQ instance accordingly.

Go to below location & hit the invalidateCaches method

<http://<hostname>.us.company.com:<prod-port>/dyn/admin/nucleus/atg/userprofiling/ProfileAdapterRepository/>



ATG-SRM Integration
April 2014

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200

oracle.com



Oracle is committed to developing practices and products that help protect the environment

Copyright © 2015, Oracle and/or its affiliates. All rights reserved.

This document is provided for information purposes only, and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document, and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

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

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group. 0113

Hardware and Software, Engineered to Work Together