ORACLE

ATG WEB COMMERCE

An Oracle White Paper
April 2014

Commerce-SRM Integration

ORACLE

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

Oracle brings industry leading solutions to market to assist merchants achieve their goals around selling, communicating with customers and the overall customer experience.

Two key elements of Oracle's offerings in this area are Oracle Commerce and Oracle Social Relationship Management (SRM). The Oracle Commerce solution provides industry leading capabilities for online and omni-channel selling, helping power many of the world's largest retailers and service providers. Oracle SRM enables organizations to socially enable the way they do business, without the cost and complexity of social silos.

Oracle has integrated the Commerce and SRM solutions to allow customers to leverage the data and knowledge they have about customers in Commerce and utilize that information for communications via social channels.

This white paper discusses the integration between Commerce and Social and the benefits it provides.

The integration leverages the Commerce product's ability to track user behavior and activity, and data explicitly provided by the shopper in a single location, the user profile. Business users are enabled to create groups of profiles (known as Segments) via the web based Business Control Center (BCC) tool and use those groups for customizing the user's experience when interacting with Commerce powered sites. Segment definition can be as simple as selecting various profile attributes (such as "gender is male" and "age is between 30 and 40"), or very complex combinations of collected data on the shopper. Through intelligent definition of segments, and using the tools that Commerce provides, the business user is provided with extensive control to tailor the experience.

The Commerce-SRM integration allows the members of the Segments to be exported from Commerce through the SRM tools and into Facebook to be used as a Custom Audience. Once the Custom Audience is created in Facebook, merchants can use Facebook provided tools to target ads and contents at the members of the Custom Audience.

Using these tools, merchants can easily segment their customers using Commerce related activity, target content to those users via social channels, and tailor the customer experience using the same segmentation.

Integration Overview

The integration between Commerce and SRM utilizes existing Commerce capabilities to define groups of users, known as Segments. Segments are defined by business users using the Business Control Center (BCC) by setting up rules or criteria for which user profiles should be included in the segment.

When serving pages for users on a Commerce site, Oracle Commerce determines the membership for the given user as to which segments the user is a member.

The integration with SRM allows the Commerce system to export a list of users that fall into specific segments to the SRM tool and ultimately to Facebook as a Custom Audience.

The integration package includes the following:

- Additional properties on Segments that indicate if the segment should be exported to SRM, and a setting as to which Custom Audience the members of the segment will be exported.

- A scheduled service that can run periodically to update all defined Custom Audiences.

When the scheduled service is run, it will perform the following steps:

- Determine which segments are indicated to be exported to SRM

- For each segment that is indicated to be exported, it will create a list of all members and send that to SRM to re-populate the membership of the Custom Audience on Facebook. Users who have the opt-out property set will be excluded.

Note that Commerce will export the data it has stored on the profiles for shoppers. There is no guarantee that users have used the same information on Facebook. Facebook will receive the list of users from Commerce and attempt to match the data provided with that which is stored in Facebook. It is expected that not all Commerce profiles will have a matching profile on Facebook.

Installation

Pre-requisites:

ATG 11.1 should be installed in combination with a supported application server.

While installing CIM, select Oracle Commerce, Oracle Commerce Reference Store, Oracle Commerce Site Administration, Oracle Commerce Platform and Oracle Commerce Content Administration

Installation Steps:

The Social Integration package should be installed on the BCC/CA server.

Social Build should be under ATG Home path:

- Create a new directory named **"social"** in /work/service/ATG/ATG11.1

[service@busgk1114 ATG11.1]$ mkdir social

[service@busgk1114 ATG11.1]$ chmod 777 social/

[service@busgk1114 ATG11.1]$ cd social/

- Download the integration software from Oracle's eDelivery site.
- Suppose the build is OCSocialIntegration1.0_33NEW.zip then go to /work/service/ATG/ATG11.1/social copy it from downloads and unzip the build file

[service@busgk1114 social]$ cp /work/service/downloads/OCSocialIntegration1.0_33NEW.zip **.**

[service@busgk1114 social]$ chmod 777 OCSocialIntegration1.0_33NEW.zip

[service@busgk1114 social]$ unzip OCSocialIntegration1.0_33NEW.zip

1. Setting the client id and client secret manually:

- Create a property file named **SRMClient.properties** by layering it under

/work/service/ATG/ATG11.1/home/localconfig/atg/

- Create new directories in this fashion /integration/social/webserviceclient

[service@busgk1114 atg]$ mkdir integration

[service@busgk1114 atg]$ cd integration/

[service@busgk1114 integration]$ mkdir social

[service@busgk1114 integration]$ cd social/

[service@busgk1114 social]$ ls

[service@busgk1114 social]$ pwd

/work/service/ATG/ATG11.1/home/localconfig/atg/integration/social

[service@busgk1114 social]$ mkdir webserviceclient

[service@busgk1114 social]$ cd webserviceclient/

[service@busgk1114 webserviceclient]$ pwd

/work/service/ATG/ATG11.1/home/localconfig/atg/integration/social/webserviceclient

- In webserviceclient directory create a file named SRMClient.properties,

and set the properties such as clientId, clientSecret, targetEndPoint, proxy.

Values for clientId, clientSecret, targetEndPoint will vary from account to account. Please obtain the correct values for your account from the Oracle SRM team.

[service@busgk1114 webserviceclient]$ vi SRMClient.properties

Examples of setting these values are shown below:

clientId=a7293e736733e0600033afee369967a08292t17e6bb8c951e5404tf168687444e

clientSecret=bd9e4c4eac5c63f7e32dbd9eeda867fb59ca491cc064180d92o59e25ad509bt3f

targetEndPoint=https://gatekeeper.staging.cloud.vitrue.com

proxy=your proxy server URL

2. Run the sql script of Social in production and publishing schema:

- Drill down to /work/service/ATG/ATG11.1/social/Versioned/sql/ORACLE and for Publishing schema run sql script SocialSegment.sql
- Go to /work/service/ATG/ATG11.1/social/sql/ORACLE/ and for Production schema run sql script SocialRepository.sql

3. .EAR file deletion and creation:

For WebLogic, follow the below steps. For use with other application servers, perform equivalent steps. See the specific application server documentation for details.

1 Below steps 1, 2, 8 and 9 are only for Weblogic. Please see the specific application server documentation for creating and deleting .EAR files for these particular steps rest are to be executed in the machine where ATG is installed. Log into the Weblogic server (e.g. http://<ATG hostname>:<port>/console)

2 Go to 'Environment' -> 'Deployments' and delete atg_publishing_lockserver.ear & atg_production_lockserver.ear

3 Go back to ATG and into '/work/service/ATG/ATG11.1/home/cimEars'

4 Delete 'atg_publishing_lockserver.ear ' & 'atg_production_lockserver.ear'

5 Stop Publishing and Production Servers.

6 Go to /work/service/ATG/ATG11.1/home/bin & rebuild the publishing & production ear files with the Social module

./runAssembler -server atg_publishing_lockserver /work/service/ATG/ATG11.1/home/../home/cimEars/atg_publishing_lockserver.ear -m <You have to add ALL MODULES which were installed when "Creating Ear file" in

publishing server while running cim, obtain the modules from your cim.log > social.Versioned

./runAssembler -server atg_production_lockserver /work/service/ATG/ATG11.1/home/../home/cimEars/atg_production_lockserver.ear -m <You have to add ALL MODULES which were installed when "Creating Ear file" in production server while running cim, obtain the modules from your cim.log> social

7 Go to ' /work/service/ATG/ATG11.1/home/cimEars' and the newly built publishing & production ear files should be there

8 Go back to the Weblogic server & go to 'Environment' -> 'Deployments'

9 Add the new publishing & 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.
- Repeat to add publishing ear
- Restart the servers (publishing and production)

4. SRM Access and Refresh Token Generation:

NOTE: For the first time you have to get the access token and refresh token manually using Google oath playground. For subsequent usage, the tokens are refreshed automatically.

Steps to Create Tokens:

- Go to https://developers.google.com/oauthplayground - This is the google OAuth 2 web client that we will be using to generate an access token.
- Click the gear/cog in the upper right corner to configure the OAuth provider information (Gatekeeper).
- Change the 'OAuth endpoints' value to 'Custom'

- Set the 'Authorization endpoint' value to Oracle SRM Authorization endpoint URL
- Set the 'Token endpoint' value to Oracle SRM Token endpoint URL
- Ensure the 'Access token location' is set to 'Authorization header w/ Bearer prefix'
- Set the 'OAuth Client ID' value to (the application ID Oracle SRM has provided)
- Set the 'OAuth Client Secret' value to (the secret key Oracle SRM has provided)
- Click the 'Close' link and the client is now setup for requests.
- Under Select & authorize APIs insert input as“custom_audiences” and then selectAuthorize API’s. This will take to the SRM server and ask for authorization. After Authorization it will automatically take you to the OAuth 2 page. Now Access Token and Refresh Token will be created

6) Add the access token and refresh token in the SocialRepository using the following:

- Go to Production Server and add the access and refresh tokens.

  (e.g. http://<ATG hostname>:<port>/atg/integration/social/SocialRepository)

```
<add-item item-descriptor="socialTokens" id="accessToken">
<set-property name="tokenValue"><![CDATA[ACCESS TOKEN VALUE]]></set-property>
</add-item>
<add-item item-descriptor="socialTokens" id="refreshToken">
<set-property name="tokenValue"><![CDATA[REFRESH TOKEN VALUE]]></set-property>
</add-item>
```

7) Set the schedule property:

- For the first time scheduler will be up and running in Publishing server.
- To change the scheduler timings. Go to Publishing server http://<hostname>: <publishing server port>/atg/integration/social/scheduler/SocialSegmentScheduler/ and select ‘schedule’ under properties.
- Set the scheduler as per your requirement. Please see the Oracle ATG Web Commerce Platform Programming Guide, “Scheduler Services” topic to get additional details.

8) Set the opt-out property:

- At Present Profile repository in Commerce has properties ‘receiveEmail’ and ‘receivePromoEmail’(receivePromoEmail comes with CRS module) which can be used as opt-out fields. More properties can be added by extending the Profile repository.
- To change the opt-out property. Go to Publishing server http://<hostname>: <publishing server port>/atg/integration/social/scheduler/SocialSegmentScheduler/ and select ‘optOutFields’ under properties and enter the opt-out fields to be considered. eg:
  “receivePromoEmail=no” , “receiveEmail=no” <together can be used> else any of the particular optOutField can be used according to the need.
- Profiles having all the opt-out fields won’t be sent by scheduler

9) Configuring the properties needs to be sent to SRM

- The properties present in the ProfileMapping.xml file are considered to be sent to Oracle SRM. Email is configured by default
- To configure a new property to be sent, the property needs to be added in the ProfileMapping.xml file and the server needs to be re-started

Uninstallation:

1) Go to /work/service/ATG/ATG11.1, delete the social folder in it.
2) Go to /work/service/ATG/ATG11.1/social/sql/ORACLE/ and drop the schema for production server by running SocialRepositoryRollBack.sql
3) Go to /work/service/ATG/ATG11.1/social/Versioned/sql/ORACLE and drop the schema for publishing server by running SocialSegmentRollBack.sql
4) Repeat the Installation step as mentioned above for Re-installation.

Operations

The scheduled service should be set to run at a frequency that is appropriate for the business. While running it very frequently will result in the latest membership in the Custom Audiences, the service will place some load on the production database. As such, careful consideration should be given to running the scheduled service during peak times.

Customization

If any data format change happens, the request needs to be re created in new format for making the webservice call.

For creating a request, social integration has the different request mappers components.
OOTB request mappers in Social Integration.

1. Component: atg/integration/social/mapper/profile/ProfileRequestMapper
   Underlying class: atg.integration.social.mapper.profile.ProfileRequestMapper.java
   This component creates the request for exporting the profile to Oracle SRM
2. Component: atg/integration/mapper/account/BundleRequestMapper
   Underlying class: atg.integration.social.mapper.account.BundleRequestMapper.java
   This component creates the request for getting bundles from Oracle SRM
3. Component: atg/integration/mapper/customaudience/CustomAudienceRequestMapper

Underlying class:
atg.integration.social.mapper.customaudience.CustomAudienceRequestMapper.java
This component creates the request for getting custom audiences of the selected bundle from Oracle SRM

Each underlying request mapper class is a subclass of abstract class "atg.integration.social.mapper.RequestMapper.java"
class "atg.integration.social.mapper.RequestMapper.java" has below abstract methods
a) public abstract HttpRequestBase createHttpMethod(Object... pInputData) throws SocialIntegrationException;
b) public abstract String createJSONInput(Object... pInputData) throws SocialIntegrationException;

For creating requests with input of new data format below activity needs to be performed
1. create a extension of RequestMapper.java
2. override the method "public abstract HttpRequestBase createHttpMethod(Object... pInputData)" with a logic for input in new data format
3. give an empty implementation of "public abstract String createJSONInput(Object... pInputData)"
4. configure this new extension class as underlying class in the corresponding mapper component

Conclusion

The capabilities provided by this integration allow merchants to better leverage their customer information and engage their customers at a more personal level via social channels.

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

Hardware and Software, Engineered to Work Together