Oracle Sales Cloud
Integrating Financial Services with Oracle Banking Platform and Siebel UCM

Release 13 (update 18B)
# Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preface</td>
<td>i</td>
</tr>
<tr>
<td>1 About This Guide</td>
<td>1</td>
</tr>
<tr>
<td>Audience and Scope</td>
<td>1</td>
</tr>
<tr>
<td>Related Guides and Resources</td>
<td>1</td>
</tr>
<tr>
<td>2 Getting Started</td>
<td>3</td>
</tr>
<tr>
<td>Software Requirements and Prerequisites: Overview</td>
<td>3</td>
</tr>
<tr>
<td>Unpacking the Collateral</td>
<td>3</td>
</tr>
<tr>
<td>3 Understanding Oracle Sales for Financials Integration</td>
<td>5</td>
</tr>
<tr>
<td>Integration Overview</td>
<td>5</td>
</tr>
<tr>
<td>Understanding Terminologies Across Applications</td>
<td>6</td>
</tr>
<tr>
<td>Integration Process Flows: Overview</td>
<td>6</td>
</tr>
<tr>
<td>Synchronizing Accounts Between Oracle Sales Cloud and Siebel UCM</td>
<td>7</td>
</tr>
<tr>
<td>Synchronizing Contacts Between Oracle Sales Cloud and Siebel UCM</td>
<td>12</td>
</tr>
<tr>
<td>Contact Point Phone and Email Synchronization</td>
<td>13</td>
</tr>
<tr>
<td>Financial Account Synchronization</td>
<td>16</td>
</tr>
<tr>
<td>Understanding Address Update Synchronization Between Oracle Sales Cloud and Siebel UCM</td>
<td>17</td>
</tr>
<tr>
<td>Party Synchronization</td>
<td>19</td>
</tr>
<tr>
<td>Mapping of Fields across Oracle Banking Platform, Siebel UCM, and Oracle Sales Cloud</td>
<td>23</td>
</tr>
</tbody>
</table>
# Configuring Oracle Sales Cloud

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Defining Source System and Source System Entities</td>
<td>27</td>
</tr>
<tr>
<td>Enabling Trading Community Events in Profile Options</td>
<td>28</td>
</tr>
<tr>
<td>Setting Up Integration Users and Roles</td>
<td>28</td>
</tr>
<tr>
<td>Creating a CSF Key for Event Subscriptions</td>
<td>31</td>
</tr>
<tr>
<td>Phone Number and Email Synchronization for Contact Points: Overview</td>
<td>32</td>
</tr>
<tr>
<td>Configuring Security Policies for Custom Objects</td>
<td>32</td>
</tr>
<tr>
<td>Changing Administrator Profile Values</td>
<td>33</td>
</tr>
<tr>
<td>Creating Legal Entity Map Record to Store Bank Code</td>
<td>34</td>
</tr>
<tr>
<td>Configuring Integration Cloud Service Web Service</td>
<td>34</td>
</tr>
<tr>
<td>Adding the Oracle Banking Platform Application URL</td>
<td>35</td>
</tr>
<tr>
<td>Configuring Page Layouts to Use Oracle Banking Platform and Third Party Servicing</td>
<td>36</td>
</tr>
<tr>
<td>Setting Up Product Servicing</td>
<td>41</td>
</tr>
<tr>
<td>Adding Create Submission Function for Opportunity Submission</td>
<td>41</td>
</tr>
<tr>
<td>Creating Object Workflow to Send Email Notification</td>
<td>46</td>
</tr>
<tr>
<td>Configuring Submission Tracker for Accounts and Contacts</td>
<td>47</td>
</tr>
<tr>
<td>Configuring Application Form and Application Tracker in Opportunities</td>
<td>48</td>
</tr>
<tr>
<td>Configuring Onboarding Customer</td>
<td>49</td>
</tr>
<tr>
<td>Configuring Oracle Banking Platform Mashup in Financial Accounts</td>
<td>51</td>
</tr>
<tr>
<td>Configuring Third Party Servicing System</td>
<td>51</td>
</tr>
<tr>
<td>Populating Value of Onboarding Status Field in Account and Contact</td>
<td>53</td>
</tr>
<tr>
<td>Understanding Financial Account Roll Up for Households</td>
<td>55</td>
</tr>
</tbody>
</table>
## 5 Configuring Siebel UCM

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prerequisites</td>
<td>59</td>
</tr>
<tr>
<td>Siebel UCM Configuration</td>
<td>59</td>
</tr>
<tr>
<td>Configuring System Preferences</td>
<td>60</td>
</tr>
<tr>
<td>Enabling Employees Import and Export as Part of Setting up the Enterprise Organization</td>
<td>60</td>
</tr>
<tr>
<td>Configuring for Know Your Customer Status Synchronization</td>
<td>61</td>
</tr>
<tr>
<td>Configuring Accommodation Type Field</td>
<td>61</td>
</tr>
<tr>
<td>Configuring Email Address Field</td>
<td>62</td>
</tr>
<tr>
<td>Configuring for Financial Asset Synchronization</td>
<td>63</td>
</tr>
<tr>
<td>Configuring for Account Synchronization</td>
<td>64</td>
</tr>
<tr>
<td>Configuring LOV Type Alignment</td>
<td>64</td>
</tr>
<tr>
<td>Configuring the Country to State Hierarchy</td>
<td>64</td>
</tr>
<tr>
<td>Creating the Country Pick List</td>
<td>65</td>
</tr>
<tr>
<td>Creating the State Pick List</td>
<td>66</td>
</tr>
<tr>
<td>Configuring the CUT Address Business Component and Country and State Fields</td>
<td>67</td>
</tr>
<tr>
<td>Configuring the Personal Address Business Component and Country and State Fields</td>
<td>68</td>
</tr>
<tr>
<td>Configuring the Account Business Component and Country Field</td>
<td>69</td>
</tr>
<tr>
<td>Configuring the Contact Business Component and Country Field</td>
<td>70</td>
</tr>
<tr>
<td>Configuring the UCM HE Constituent Identification Business Component and Country Field</td>
<td>71</td>
</tr>
<tr>
<td>Ensuring the Compatibility of S_ADDR_PER Table and COUNTRY and STATE Columns of Lov Type</td>
<td>71</td>
</tr>
<tr>
<td>Ensuring the Compatibility of S_ORG_EXT Table and COUNTRY Column of Lov Type</td>
<td>72</td>
</tr>
<tr>
<td>Ensuring the Compatibility of S_CON_IDNTY_DOC Table and COUNTRY Column of Lov Type</td>
<td>72</td>
</tr>
<tr>
<td>Adjusting Seed data to Match Country and State Business Needs</td>
<td>73</td>
</tr>
<tr>
<td>Address LOV Field Mapping</td>
<td>74</td>
</tr>
<tr>
<td>Configuring the Communication Use Type Pick List</td>
<td>77</td>
</tr>
<tr>
<td>Configuring the Contact with Financial Asset Relationship Pick List</td>
<td>77</td>
</tr>
<tr>
<td>Configuring the Account with Financial Asset Relationship Pick List</td>
<td>79</td>
</tr>
<tr>
<td>Configuring Account Type LOV</td>
<td>80</td>
</tr>
<tr>
<td>Additional Notes for Upgrading Siebel UCM</td>
<td>80</td>
</tr>
</tbody>
</table>
## Setting Up the Product Master

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product Master: Overview</td>
<td>81</td>
</tr>
<tr>
<td>Mapping of Objects and Product Type between Oracle Sales Cloud and Oracle Banking Platform</td>
<td>81</td>
</tr>
<tr>
<td>Exporting Product Catalog from Oracle Banking Platform</td>
<td>82</td>
</tr>
<tr>
<td>Creating a Matching Product Group Structure in Oracle Sales Cloud</td>
<td>84</td>
</tr>
<tr>
<td>Importing Product Groups in Oracle Sales Cloud</td>
<td>84</td>
</tr>
<tr>
<td>Importing Products to Oracle Sales Cloud</td>
<td>87</td>
</tr>
<tr>
<td>Importing Cross References for Products and Product Groups</td>
<td>91</td>
</tr>
</tbody>
</table>

## Configuring Oracle Integration Cloud Service

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuring Oracle Integration Cloud Service: Overview</td>
<td>95</td>
</tr>
<tr>
<td>Importing Certificates to Integration Cloud Service: Overview</td>
<td>95</td>
</tr>
<tr>
<td>Importing the Integration Package: Highlights</td>
<td>96</td>
</tr>
<tr>
<td>Configuring Integration Service Endpoints: Explained</td>
<td>96</td>
</tr>
<tr>
<td>Configuring Integration Connections: Highlights</td>
<td>98</td>
</tr>
<tr>
<td>Cloning and Activating Integrations</td>
<td>102</td>
</tr>
<tr>
<td>Configuring Multiple Email IDs in Integration Service Mail Notification</td>
<td>103</td>
</tr>
<tr>
<td>Adding Emails to the Lookup for Integration Notifications: Explained</td>
<td>104</td>
</tr>
<tr>
<td>Setting Up Error Notifications: Explained</td>
<td>104</td>
</tr>
<tr>
<td>Integration Artifacts Reference for Oracle Integration Cloud Service</td>
<td>104</td>
</tr>
</tbody>
</table>

## Configuring Application Integration Architecture

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration and Deployment of Integration: Overview</td>
<td>109</td>
</tr>
<tr>
<td>Copying Integration Artifacts</td>
<td>109</td>
</tr>
<tr>
<td>Deploying Integration Artifacts Using AIA Installer</td>
<td>110</td>
</tr>
<tr>
<td>Deploying AIA Metadata File Manually: Procedure</td>
<td>111</td>
</tr>
<tr>
<td>Verifying the Deployment</td>
<td>111</td>
</tr>
<tr>
<td>Undeploying Integration Package</td>
<td>113</td>
</tr>
<tr>
<td>Importing ICS Certificate to AIA</td>
<td>113</td>
</tr>
<tr>
<td>Creating CSF Map</td>
<td>114</td>
</tr>
<tr>
<td>Registering ICS as a System in AIA</td>
<td>114</td>
</tr>
<tr>
<td>Setting Up Error Notifications in AIA</td>
<td>115</td>
</tr>
<tr>
<td>Tuning AIA</td>
<td>116</td>
</tr>
<tr>
<td>Integration Artifacts Reference for AIA</td>
<td>116</td>
</tr>
</tbody>
</table>
## 9 Tracing and Managing Synchronization Errors

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tracing and Managing Synchronization Errors in Integration Cloud Service</td>
<td>127</td>
</tr>
<tr>
<td>Tracing and Managing Synchronization Errors in Oracle AIA</td>
<td>128</td>
</tr>
</tbody>
</table>

## 10 Performing Bulk Import and Export of Employee Data

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Importing and Exporting Employee Data for Integration</td>
<td>131</td>
</tr>
<tr>
<td>Importing and Exporting Employee Data in Siebel UCM</td>
<td>131</td>
</tr>
<tr>
<td>Importing and Exporting Employees Data in Oracle Sales Cloud</td>
<td>138</td>
</tr>
<tr>
<td>Importing and Exporting Employees Data in Oracle Banking Platform</td>
<td>151</td>
</tr>
<tr>
<td>Importing and Exporting Cross References</td>
<td>156</td>
</tr>
</tbody>
</table>

## 11 Using Integrated Sales Cloud for Financials

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Using Integrated Sales Cloud for Financials: Overview</td>
<td>167</td>
</tr>
<tr>
<td>Customer Onboarding: Overview</td>
<td>169</td>
</tr>
<tr>
<td>Financial Accounts: Overview</td>
<td>170</td>
</tr>
</tbody>
</table>
Preface

This preface introduces information sources that can help you use the application.

Using Oracle Applications

Using Applications Help

Use help icons 🎨 to access help in the application. If you don’t see any help icons on your page, click your user image or name in the global header and select Show Help Icons. Not all pages have help icons. You can also access Oracle Applications Help.

Watch: This video tutorial shows you how to find help and use help features.

You can also read Using Applications Help.

Additional Resources

- Community: Use Oracle Cloud Customer Connect to get information from experts at Oracle, the partner community, and other users.
- Guides and Videos: Go to the Oracle Help Center to find guides and videos.
- Training: Take courses on Oracle Cloud from Oracle University.

Conventions

The following table explains the text conventions used in this guide.

<table>
<thead>
<tr>
<th>Convention</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>boldface</td>
<td>Boldface type indicates user interface elements, navigation paths, or values you enter or select.</td>
</tr>
<tr>
<td>monospace</td>
<td>Monospace type indicates file, folder, and directory names, code examples, commands, and URLs.</td>
</tr>
<tr>
<td>&gt;</td>
<td>Greater than symbol separates elements in a navigation path.</td>
</tr>
</tbody>
</table>

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website.

Videos included in this guide are provided as a media alternative for text-based help topics also available in this guide.
Contacting Oracle

Access to Oracle Support
Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit My Oracle Support or visit Accessible Oracle Support if you are hearing impaired.

Comments and Suggestions
Please give us feedback about Oracle Applications Help and guides! You can send an e-mail to: oracle_fusion_applications_help_ww_grp@oracle.com.
About This Guide

Audience and Scope

This guide is intended for anyone who is involved in integrating Oracle Sales Cloud with Oracle Banking Platform and Oracle Customer Hub, using Oracle’s Integrated Cloud Service and Application Integration Architecture. The guide describes the implementation and configuration steps that are required for this integration.

This guide is designed to be used as a template. It provides a starting point that shows how Oracle Sales Cloud and Oracle Banking Platform can be used together to create a value-added business process and user experience. You must configure and install patches to create the integration.

If you want to set up and work with the additional features of Oracle Sales Cloud and Oracle Integration Cloud Service, see Oracle Help Center.

Related Topics

- Oracle Help Center

Related Guides and Resources

See the following related guides and resources to learn more about the integration tasks covered in this guide.

<table>
<thead>
<tr>
<th>Title</th>
<th>Description and Resources</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Sales Cloud Getting Started with Your Implementation</td>
<td>Describes how to set up a sales automation solution in Oracle Sales Cloud using a case study to describe concepts and procedures. See Oracle Sales Cloud documentation on Oracle Help Center.</td>
</tr>
<tr>
<td>Oracle Sales Cloud Implementing Sales</td>
<td>Describes how to configure and set up Sales. See Oracle Sales Cloud documentation on Oracle Help Center.</td>
</tr>
<tr>
<td>Oracle Sales Cloud Extending Sales</td>
<td>Describes how to use tools to configure and extend the Oracle Sales Cloud. See Oracle Sales Cloud documentation on Oracle Help Center.</td>
</tr>
<tr>
<td>Oracle Sales Cloud Understanding File-Based Data Import and Export</td>
<td>Describes how to import legacy and other data into Oracle Sales Cloud using File-Based Data Import. See Oracle Sales Cloud documentation on Oracle Help Center.</td>
</tr>
</tbody>
</table>
| Oracle Banking Platform | - See the Oracle Banking Platform Documentation home page link listed in the Related Topics section in this topic.  
- See the Oracle Banking Platform Installation Guide link listed in the Related Topics section in this topic.  
- See the Oracle Banking Platform Administrator’s Guide link listed in the Related Topics section in this topic. |
| Using Oracle Integration Cloud Service | Describes how to connect applications in the cloud and monitor and manage these running integrations. It also describes how to access and use prebuilt integrations. |
Oracle Sales Cloud
Integrating Financial Services with Oracle Banking Platform and Siebel UCM

Chapter 1
About This Guide

<table>
<thead>
<tr>
<th>Title</th>
<th>Description and Resources</th>
</tr>
</thead>
<tbody>
<tr>
<td>Using the SOAP Adapter</td>
<td>Describes how to use the SOAP Adapter with Oracle Integration Cloud Service.</td>
</tr>
<tr>
<td>Using the Oracle Sales Cloud Adapter</td>
<td>Describes how to use the Oracle Sales Cloud Adapter with Oracle Integration Cloud Service.</td>
</tr>
<tr>
<td>Siebel UCM Documentation</td>
<td>See the Siebel UCM Documentation home page link listed in the Related Topics section in this topic.</td>
</tr>
</tbody>
</table>

Related Topics

- Siebel UCM Documentation
- Oracle Integration Cloud Service Documentation
- Oracle Banking Platform Documentation
- Oracle Banking Platform Installation Guide
- Oracle Banking Platform Administrator's Guide
2 Getting Started

Software Requirements and Prerequisites: Overview

This topic lists the software requirements for integrating Oracle Sales Cloud with Oracle Banking Platform and Oracle Customer Hub (also known as Siebel UCM), using Oracle Integrated Cloud Service and Application Integration Architecture.

The software requirements are as follows:

- Oracle Banking Platform (2.6.0.2)
- Oracle Customer Hub (Siebel Innovation Pack 2015 Patchset 9 + POC Patch ID 27019135)

Note: To procure a password, customers must log a service request with the Oracle Customer Support indicating the patch ID.

- Oracle Sales Cloud (12.1 PB11) or greater (including R13.17.09 PB11 limited availability)
- Application Integration Architecture Foundation Pack 11g R1 (11.1.1.7.0) or later
- Oracle Integrated Cloud Service (17.3.3) or later

Unpacking the Collateral

The collateral contains files required to integrate Financial Services with Oracle Banking Platform and Siebel UCM solutions. You must first download and save the collateral to your desktop, and then extract the files.

To download the collateral:

1. Open the Collateral for Integrating Financial Services with Oracle Banking Platform and Siebel UCM (Article Doc ID 2293151.1) from My Oracle Support.
2. Right-click and save the cxfins.zip compressed file.
3. Extract the contents to a folder.

The content structure of the collateral zip file is as follows:

- `<cxfins.zip>/aia`: contains AIA artifacts.
- `<cxfins.zip>/ics`: contains ICS artifacts.
- `<cxfins.zip>/bulkintegration`: contains bulk import and export related artifacts.
3 Understanding Oracle Sales for Financials Integration

Integration Overview

Existing customers of Siebel UCM and Application Integration Architecture (AIA) can use Oracle Integration Cloud Service (ICS) to integrate with Oracle Sales Cloud and Oracle Banking Platform. Such an integration facilitates an automated process from lead qualification to financial account. The automated process in the single application ensures that there is minimal or no user intervention in application processing after an opportunity is submitted.

Note: Integrations are not turnkey solutions. Each implementation of Oracle Sales Cloud and Oracle Banking Platform is unique, and each customer has different needs that require them to implement application configurations that support their unique business requirements.

Integration Component Architecture

Oracle Sales Cloud and Siebel UCM bidirectional integration uses ICS as one of the integration components. Oracle Integration Cloud Service is a complete, secure, but lightweight integration solution that enables you to connect your applications in the cloud. It allows connectivity between various Oracle CX applications, Oracle On-premise applications and other third party customer solutions.

The following Oracle Integration Cloud Service adapters are used for this integration:

- **Oracle Sales Cloud Adapter**: Used to connect to Oracle Sales Cloud.
• **SOAP Adapter**: Used to connect to the Oracle SOA Suite where Application Information Architecture based web services are deployed. Using SOAP Adapter, you can create submissions and update opportunity flows from Oracle Banking Platform to Oracle Sales Cloud.

Integration Cloud Service lookups map the different codes or terms used by the applications you are integrating to describe similar items, such as country or gender codes. The visual data mapper enables you to quickly create direct mappings between the trigger and invoke the data structures. For the mapper, you can also access lookup tables and use standard XPath functions to map data between your applications.

Use the Integration Cloud Service dashboard to monitor the status and processing statistics for each integration. Using the dashboard, measure and track the performance of your transactions by capturing key reporting information, such as throughput, the number of messages processed successfully, and the number of messages that failed processing. You can also manage business identifiers that track fields in messages and manage errors by integrations, connections, or specific integration instances.

### Understanding Terminologies Across Applications

Use the following table as a reference to understand how certain terms in one application are mapped to other applications.

<table>
<thead>
<tr>
<th>Oracle Sales Cloud</th>
<th>Siebel UCM</th>
<th>Oracle Banking Platform</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account</td>
<td>Account</td>
<td>Organization, Trust</td>
</tr>
<tr>
<td>Contact</td>
<td>Contact</td>
<td>Individual</td>
</tr>
<tr>
<td>Financial Account</td>
<td>Financial Account</td>
<td>Account</td>
</tr>
<tr>
<td>Resource</td>
<td>Employee</td>
<td>User</td>
</tr>
<tr>
<td>Enterprise</td>
<td>Top Level Organization</td>
<td>Enterprise</td>
</tr>
<tr>
<td>Legal Entity</td>
<td>N/A</td>
<td>Legal Entity</td>
</tr>
<tr>
<td>Business Unit</td>
<td>Organization</td>
<td>Market Entity</td>
</tr>
<tr>
<td>Business Unit</td>
<td>Organization</td>
<td>Business Unit</td>
</tr>
<tr>
<td>Resource Organization</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Resource</td>
<td>Employee</td>
<td>User</td>
</tr>
<tr>
<td>KYC Status</td>
<td>NXG Identity Verification Status</td>
<td>Final Verification Status</td>
</tr>
<tr>
<td>On boarding Status</td>
<td>Due Diligence Status</td>
<td>Due Diligence Status</td>
</tr>
</tbody>
</table>
Integration Process Flows: Overview

The following major process flows are supported as a part of this integration:

- Bidirectional synchronization of customer accounts and contacts between Oracle Sales Cloud and Siebel UCM
- Synchronization of financial accounts from Siebel UCM to Oracle Sales Cloud
- Opportunity submission and submission synchronization
- Customer onboarding and due diligence status tracking
- Application form management and application tracking

Basic Echo Suppression in Bidirectional Synchronization

For synchronization flow from Siebel UCM to Oracle Sales Cloud, an echo is generated in Oracle Sales Cloud whenever a new account or contact is created. So, handling echo suppression is necessary in Oracle Sales Cloud. You must create a new integration user in Oracle Sales Cloud and use this user to register OSC adapter and SOAP adapter for targeting Siebel UCM in ICS. In the ICS integration, you must also add a filter expression to pick transactions for events generated by all users except the newly created integration user.

Synchronizing Accounts Between Oracle Sales Cloud and Siebel UCM

Bidirectional synchronization of accounts between Oracle Sales Cloud and Siebel UCM supports creation of new accounts and update of existing ones. When you update account records in Oracle Sales Cloud, if the update results in an error stating that the record is locked by another user, then close the record and open again to update. To support bidirectional synchronization, the account records in Siebel UCM must be exported and imported into Oracle Sales Cloud initially. The account records in Oracle Sales Cloud are also exported and imported into Siebel UCM if applicable. The synchronization process ensures that each record in Oracle Sales Cloud has a reference to the unique Siebel UCM Row ID, and each record in Siebel UCM has a reference to the unique Party ID in Oracle Sales Cloud, which is used as a unique key for matching records during synchronization.

**Note:** For geography mapping and validation, you must turn off two address fields, City and Postal Code in Oracle Sales Cloud to match their field types in Siebel UCM which are free text fields.

Account Address Synchronization

Multiple address synchronization is supported for an account. Accounts in Oracle Sales Cloud can have primary and secondary addresses and it is possible to have more than one address for each account. The synchronization process supports two address types that is, registered and postal. A new account created in Oracle Sales Cloud is synchronized with Siebel UCM and vice-versa. The changes to the existing accounts are also synchronized bidirectionally.

Siebel UCM acts as a customer master and makes all the decision related to an account’s uniqueness. If an account, or an address associated with an account is identified in Oracle Sales Cloud as unique, it is sent over to Siebel UCM, which as a customer master, can make decision to either accept, reject, or signal a merge.
Note:

- Shared addresses for Oracle Sales Cloud are not supported in this integration. This means that when you create an account in Oracle Sales Cloud, all addresses associated with the account are treated as new addresses in Oracle Sales Cloud. Address deletion is not supported for synchronization since there is no way to identify the deleted addresses in Oracle Sales Cloud.
- The address rules in Oracle Sales Cloud and Siebel UCM are different. So, while creating and editing an account or contact with the address fields, the mandatory address fields across the systems need to be synchronized. So, when creating or editing an account or a contact with address fields, you must fill in the following fields so that Siebel UCM address rules are respected and the record gets synchronized successfully:
  - Address Line 1
  - City
  - Postal Code

The following table provides information about the integration component for address synchronization in Siebel UCM:

<table>
<thead>
<tr>
<th>Object</th>
<th>Integration Object</th>
<th>Integration Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account</td>
<td>SwiOrganizationIO</td>
<td>UCM Account Address</td>
</tr>
</tbody>
</table>

The following table provides information about the user property to switch address business component in Siebel UCM:

<table>
<thead>
<tr>
<th>IO</th>
<th>Integration Component (Version 1)</th>
<th>Integration Component (Version 2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SwiOrganizationIO</td>
<td>Account__Business Address</td>
<td>UCM Account Address</td>
</tr>
</tbody>
</table>

If you as the user want to continue with integration component, use the following property:

<table>
<thead>
<tr>
<th>Name of the Property for Organization Service Routing</th>
<th>Definition</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>Routing. UCMOrganizationService.RouteToAccountBusinessAddress</em></td>
<td><em>An AIA property to control use of either 'Account__Business Address' or 'UCM Account Address' Business Component in 'Account-Address' Sync</em></td>
<td>False(IC : UCM Account Address)</td>
</tr>
</tbody>
</table>

The following table lists rules for account and account address synchronization from Oracle Sales Cloud to Siebel UCM:

<table>
<thead>
<tr>
<th>Oracle Sales Cloud</th>
<th>Siebel UCM</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>A new account with new address is created.</td>
<td>Account and address do not exist.</td>
<td>Account and address are synchronized to Siebel UCM.</td>
</tr>
<tr>
<td>A new account with an existing address is created.</td>
<td>Account and address both exist.</td>
<td>Error: A record with identical values already exists in the Siebel database. Ensure that the field values in the input message are unique.  (SBL-EAI-04381)(SBL-IAI-00512)</td>
</tr>
</tbody>
</table>
A new account with an existing address is created. Account and address are both new. Account with address records is synchronized to Siebel UCM.

A new account with a new or existing address is created. Account is new and address exists. Error: A record with identical values already exists in the Siebel database. Ensure that the field values in the input message are unique. (SBL-EAI-04381)(SBL-IAI-00512)

A new account with a new address is created. Account and address are both new. Synchronize account and address record to Siebel UCM.

The following table lists rules for account and account address synchronization from Siebel UCM to Oracle Sales Cloud:

<table>
<thead>
<tr>
<th>Siebel UCM</th>
<th>Oracle Sales Cloud</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>A new account with a new address is created.</td>
<td>Account is new.</td>
<td>Account record is synchronized in Oracle Sales Cloud.</td>
</tr>
<tr>
<td>A new account with a new address is created.</td>
<td>Account exists.</td>
<td>Cross reference record is captured in Siebel UCM.</td>
</tr>
<tr>
<td>A new account with an existing address is created.</td>
<td>Account is new.</td>
<td>Account record is synchronized in Oracle Sales Cloud.</td>
</tr>
<tr>
<td>A new account with an existing address is created.</td>
<td>Account exists.</td>
<td>Cross reference record is captured in Siebel UCM.</td>
</tr>
</tbody>
</table>

Account Team and Ownership Synchronization

Account owner in Oracle Sales Cloud maps to primary account team member in Siebel UCM. The following points must be considered for bidirectional synchronization of account team and ownership between Oracle Sales Cloud and Siebel UCM:

- When a new account is created in Oracle Sales Cloud, the account owner defaults to the currently signed in user.
- If an account owner is not an employee that is mapped between Oracle Sales Cloud and Siebel UCM:
  - For newly created accounts, owner is set to the integration user.
  - For existing accounts that are being updated, the previous account owner is retained and following error message is displayed: The owner for Account '<Account_Name>' with PartyUID '<Account_PartyUID>' is not synchronized.
- When Account Owner in Oracle Sales Cloud is updated, the synchronization process updates the primary Account Team Member in Siebel UCM.
- When primary Account Team Member in Siebel UCM is updated, the synchronization process updates the Account Owner in Oracle Sales Cloud.
Account Contacts Association Synchronization

The following rules are applicable for synchronization of account contacts association between Siebel UCM and Oracle Sales Cloud:

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>An existing contact is associated to an account in Siebel UCM.</td>
<td>Association is reflected in Oracle Sales Cloud.</td>
</tr>
<tr>
<td>A new contact is associated to an account in Siebel UCM, and the contact is synchronized in Oracle Sales Cloud.</td>
<td>Association is reflected in Oracle Sales Cloud.</td>
</tr>
<tr>
<td>An existing contact is associated to an account.</td>
<td>Association is reflected in Siebel UCM.</td>
</tr>
<tr>
<td>A new contact is associated to an account in Oracle Sales Cloud, and the contact is synchronized in Siebel UCM.</td>
<td>Association is reflected in Siebel UCM.</td>
</tr>
<tr>
<td>One or more contacts are added to an account in Oracle Sales Cloud and contacts are synchronized in Siebel UCM.</td>
<td>Association is reflected in Siebel UCM.</td>
</tr>
<tr>
<td>One or more contacts are added to an account in Siebel UCM and contacts are synchronized in Oracle Sales Cloud.</td>
<td>Association is reflected in Oracle Sales Cloud.</td>
</tr>
</tbody>
</table>

**Note:** Account contact associations that trigger account update events in Oracle Sales Cloud are synchronized to Siebel UCM. Any association that involves creation of the associated account or contact is not synchronized and is skipped if the account or contact created are not synchronized before the association.

In Siebel UCM, for a contact, the first account that gets synchronized from Oracle Sales Cloud is identified as primary account even in cases where Oracle Sales Cloud has another account as primary. There is no primary flag mapping for contact account association between Oracle Sales Cloud and Siebel UCM.

According to the TCA model, if you are creating any relation between an organization and contact, the first relation is considered as the primary relation, irrespective of the relation type. If you are adding the first relation from relationship tab, it is considered as a primary contact. You must always create the first contact from contact subtab, which would act as primary and then create other relationships.

Account Contact Relationship Synchronization

Account contact relationship synchronization is the bidirectional synchronization of account contact relationships. If the contact is not synchronized before adding it as a relationship, the relationship is skipped from synchronization process and an email notification is sent with the details and required action. You can create, modify, or remove account and contact relationship in Siebel UCM, and the change is reflected in Relationship tab of both account and contact in Oracle Sales Cloud. The reverse flow for these scenarios, for synchronization from Oracle Sales Cloud to Siebel UCM is also supported.
The following integration rules are applicable for Account Contact Relationship Synchronization. The reverse flows for these integration rules also hold true.

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Relationship Role</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>A synchronized contact is added to an account as a relationship.</td>
<td>Person who is a contact for an organization</td>
<td>Contact appears as an association for the account in Siebel UCM.</td>
</tr>
<tr>
<td>A synchronized contact is added to an account as a relationship.</td>
<td>Board Member</td>
<td>Contact appears as a party relationship for the account in Siebel UCM.</td>
</tr>
<tr>
<td>A synchronized account is added to a contact as a relationship.</td>
<td>Organization which has contact</td>
<td>Account appears as an association for the account in Siebel UCM.</td>
</tr>
<tr>
<td>A synchronized account is added to a contact as a relationship.</td>
<td>Board</td>
<td>Account appears as a party relationship for the account in Siebel UCM.</td>
</tr>
</tbody>
</table>

✏️ **Note:** Only relationship role of Board Member is mapped from Siebel UCM to Oracle Sales Cloud. However, you can add more roles to this mapping.

**Account Contact Point Synchronization**

Account contact point consists of primary email and primary phone of the account, which are bidirectionally synchronized. This is picked up from the primary contact.

The following table explains how the contact phone details of an account changes based on changes in the contact point of the primary contact in Oracle Sales Cloud:

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Primary contact is not available, which means that Primary check box is</td>
<td>Primary phone and email are not available for that account.</td>
</tr>
<tr>
<td>not selected in the Contact tab of an account.</td>
<td></td>
</tr>
<tr>
<td>Primary contact is removed from the account in Oracle Sales Cloud.</td>
<td>Primary phone and email are also removed from that account.</td>
</tr>
<tr>
<td>Primary contact's primary phone and email are added.</td>
<td>Primary phone and email are also added for the account.</td>
</tr>
<tr>
<td>Primary contact's primary phone and email are changed.</td>
<td>Primary phone and email are also changed for the account.</td>
</tr>
<tr>
<td>Primary contact's phone and email are removed in Oracle Sales Cloud.</td>
<td>Primary phone and email in the account are removed.</td>
</tr>
</tbody>
</table>
**Note:** For more details, see Contact Point Synchronization in Synchronizing Contacts Between Oracle Sales Cloud and Siebel UCM.

The All Email and All Phone on Account are additional details and change through them have no bearing on Primary Phone and Email. Any changes in Account’s contact point are bidirectionally synchronized between Oracle Sales Cloud and Siebel UCM. If All Email or All Phone and, or Primary Email and Phone in Oracle Sales Cloud are exposed, then the Primary Email and Primary Phone must be available there after synchronization from Siebel UCM.

**Account Type Synchronization**

Oracle Sales Cloud supports account type of Prospect and Customer, with Prospect being the initial value. In Siebel UCM, initial value of Account Type for Siebel UCM is Customer. Prospect does not exist, only Customer exists. A new value of Prospect must be added to Siebel UCM AccountTypeCode (Account Class) attribute, which uses `CUT_ACCOUNT_TYPE` list of values. See Configuring Siebel UCM, for Integration and for more information on adding Prospect to Siebel UCM list of values.

Any value outside the supported mappings must be added in respective maps for it to be synchronized. Missing mapping for required fields will lead to synchronization failure if default value is not present in the respective system. Partial mapping for required fields will lead to synchronization failure.

**Synchronizing Contacts Between Oracle Sales Cloud and Siebel UCM**

The bidirectional synchronization between Oracle Sales Cloud and Siebel UCM supports creation of new contacts and modifications of existing contacts. When you update contact records in Oracle Sales Cloud, if the update results in an error stating that the record is locked by another user, then close the record and open again to update.

To support bidirectional synchronization, the contact records in Siebel UCM must be exported and imported into Oracle Sales Cloud initially. The contact records in Oracle Sales Cloud are also exported and imported into Siebel UCM if applicable.

For geography mapping and validation, you must turn off two address fields, City and Postal Code in Oracle Sales Cloud to match their field type in Siebel UCM which are free text fields.

**Contact Ownership and Team Member Synchronization**

Contact Owner in Oracle Sales Cloud maps to Primary Contact Team member in Siebel UCM. All the integration rules for account ownership and team synchronization are also applicable to contact ownership and team synchronization. See Synchronizing Accounts Between Oracle Sales Cloud and Siebel UCM for more information.

When a contact is created in Oracle Sales Cloud, the integration user and sales administrator user are added as team members. When the new team member is added to a contact in Siebel UCM, the team member details are synchronized with Oracle Sales Cloud.

The primary sales representative of the contact in Siebel UCM is considered for owner association in Oracle Sales Cloud. The Integration ID of the primary sales representative’s position is used to map the owner. Together with the owner, ICS integration user is also added to the team in Oracle Sales Cloud.

If the contact owner is not an employee that is mapped between Oracle Sales Cloud and Siebel UCM, and for newly created contacts, owner is set to the integration user. For existing contacts that are being updated, the previous contact owner is
Contact Address Synchronization

All the integration rules for account address synchronization are also applicable to contact address synchronization. See Synchronizing Accounts Between Oracle Sales Cloud and Siebel UCM for more information. Multiple address synchronization is supported for a contact. Contacts in Oracle Sales Cloud can have primary and secondary addresses and it is possible to have more than one address for each contact. The synchronization process supports two address types that is, registered and postal. A new contact created in Oracle Sales Cloud is synchronized with Siebel UCM and vice-versa. The changes to the existing contacts are also synchronized bidirectionally.

Siebel UCM acts as a customer master and makes all the decision related to a contact’s uniqueness. If a contact, or an address associated with a contact is identified in Oracle Sales Cloud as unique, it is sent over to Siebel UCM, which as a customer master, can make decision to either accept, reject, or signal a merge.

Note: Shared addresses for Oracle Sales Cloud are not supported in this integration. This means that when you create a contact in Oracle Sales Cloud, all addresses associated with the contact are treated as new addresses in Oracle Sales Cloud. Address deletion is not supported for synchronization since there is no way to identify the deleted addresses in Oracle Sales Cloud.

The following table provides information about the integration component for address synchronization in Siebel UCM:

<table>
<thead>
<tr>
<th>Object</th>
<th>Integration Object</th>
<th>Integration Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contact</td>
<td>SwiPersonIO</td>
<td>UCM HE Constituent Address</td>
</tr>
</tbody>
</table>

The following table provides information about the user property to switch address business component in Siebel UCM:

<table>
<thead>
<tr>
<th>IO</th>
<th>Integration Component (Version 1)</th>
<th>Integration Component (Version 2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SwiPersonIO</td>
<td>Contact_INS Personal Address</td>
<td>UCM HE Constituent Address</td>
</tr>
</tbody>
</table>

If you as the user want to continue with integration component, use the following property:

<table>
<thead>
<tr>
<th>Name of the Property for Person Service Routing</th>
<th>Definition</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Routing. UCMPersonService. RouteToINSPersonalAddress&quot;</td>
<td><em>An AIA property to control use of either <em>Contact_INS Personal Address</em> or <em>UCM HE Constituent Address</em> Business Component in <em>Contact - Address</em> Sync</em></td>
<td>False(C : UCM HE Constituent Address)</td>
</tr>
</tbody>
</table>

Contact Relationship Synchronization

All the integration rules for account contact relationship synchronization are also applicable to contact address synchronization. See Synchronizing Accounts Between Oracle Sales Cloud and Siebel UCM for more information.
Contact Point Phone and Email Synchronization

When contact points are synchronized from Siebel UCM to Oracle Sales Cloud, set up phone number formats for different countries in Oracle Sales Cloud to synchronize the phone numbers in correct format for values for country code, area code, and phone number. When synchronizing phone numbers, the country code is displayed for the phone numbers only where the country code is different from the country specified in Siebel UCM parameter user locale.

For the country in user locale, the country code will not be displayed in the phone number in Siebel UCM and Oracle Sales Cloud. See Working with Telephone Formats section in the Siebel Applications Administration guide for further information on Siebel UCM. Also see the Define Contact Points chapter in the Implementing Customer Data Management guide to understand how to define phone number formats for different countries in Oracle Sales Cloud. Siebel UCM publishes phone numbers together with extension numbers as a single number. The complete number published from Siebel UCM is communicated back to Oracle Sales Cloud. If a matching phone number format is not found, the complete number after country code is communicated to phone number attribute and area code, extension is set empty. Synchronization of phone and email fails when number of characters exceed the limit in the target application such as Oracle Sales Cloud or Siebel UCM. For Contact, in Siebel UCM to Oracle Sales Cloud flow, primary flag is set for the first created phone and email information in Oracle Sales Cloud, even if the primary flag is not set for the same Phone or email record in Siebel UCM. For Account, its primary contact’s primary phone and email are synchronized. Synchronization of an update to an account’s phone number (by navigating to the contact and modifying its phone number), is not initiated unless another attribute such as account name is updated.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Siebel UCM (Field and LOV)</th>
<th>Siebel UCM (Value)</th>
<th>Application Integration Architecture (DVM)</th>
<th>Application Integration Architecture (Corresponding AIA Value)</th>
<th>Oracle Integration Cloud Service (Lookup)</th>
<th>Oracle Integration Cloud Service (Mapped Oracle Sales Cloud Value)</th>
<th>Oracle Sales Cloud (Field)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contact Point</td>
<td>SwiPersonIO/Contact_Comm</td>
<td>Work</td>
<td>CUSTOMPARTY.</td>
<td>WORK</td>
<td>CONTACTPOINT_PERSONAL</td>
<td>BUSINESS</td>
<td>ContactPointPurpose</td>
</tr>
<tr>
<td></td>
<td>Address/Communication</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(Communication Address.Use Type)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SwiPersonIO/Contact_Alte</td>
<td></td>
<td>CUSTOMPARTY.</td>
<td>WORK</td>
<td>CONTACTPOINT_PERSONAL</td>
<td>BUSINESS</td>
<td>ContactPointPurpose</td>
</tr>
<tr>
<td></td>
<td>Phone/AlternatePh</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(Alternate Phone.Use Type)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>LOV:COMM_USE_TYP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Mapping for All Email Types in Oracle Sales Cloud

Following are the steps:

1. Add values in Siebel UCM.

Following are the steps to add email types similar to Oracle Sales Cloud in Siebel UCM:

   a. Sign into Siebel UCM as Siebel Administrator.
   b. Navigate to Site Map > Administration - Data > List of values.
   c. Query for COMM_USE_TYPE LOV type.
   d. Add new row for each type.
   e. Click Clear Cache.
   f. Repeat steps 1.4 and 1.5 for each new type.


Following are the steps for mapping in Application Integration Architecture:

   a. Edit the dvm file to update the mapping for state type CUSTOMERPARTY_COMMUNICATION_TYPE.dvm
   b. Ensure that the UCM_01 value for the UseType that need to be mapped, matches the value in Siebel UCM.
   c. Note the value for COMMON for the UseType.
d. Deploy the file in MDS. See Updating MDS section in Oracle Fusion Middleware Migration guide for Oracle Application Integration Architecture for deploying file to MDS.

\[\textbf{Note:}\] If any of the value for \texttt{UCM\_01} and \texttt{COMMON} is already present in dvm, edit the existing values for mappings.

3. Find the code for Type in Oracle Sales Cloud.

Following are the steps:

a. Sign into Oracle Sales Cloud.
b. Navigate to Contacts.
c. Click Create Contact.
d. Add email of the type for which the mapping must be done.
e. Obtain the value for the \texttt{CONTACT\_POINT\_PURPOSE} column from the \texttt{HZ\_CONTACT\_POINTS} table for the newly added email.


Following are the steps:

a. Sign into Oracle Integration Cloud Service.
b. Navigate to Menu > Designer > Lookups.
c. Edit \texttt{CONTACTPOINT\_PURPOSE\_LOOKUP}.
d. Ensure that the value of Application Integration Architecture in lookup for \texttt{UseType} is same as the mapping for \texttt{COMMON} in Application Integration Architecture. Edit if required and save. Ensure that the value of Oracle Sales Cloud is the value Obtained in step 3. Edit if required and save.

\[\textbf{Note:}\] If any of the value for Oracle Sales Cloud and Application Integration Architecture is already existing in lookup, delete the existing values or edit the existing values.

Mapping for Phone Type in Oracle Sales Cloud

Following are the steps to support fax type in phones:

1. Sign into Oracle Integration Cloud Service.
3. Edit \texttt{CONTACTPOINT\_PHONETYPE\_LOOKUP}.
4. Add a new row and provide the following values:
   - Value for Application Integration Architecture: FAX
   - Value for Oracle Sales Cloud: FAX
5. Click \texttt{Save}.

Related Topics

- Working with Telephone Formats
## Financial Account Synchronization

The following table lists the LOV fields which are mapped and the corresponding AIA and ICS lookup fields.

<table>
<thead>
<tr>
<th>Siebel UCM Field</th>
<th>AIA DVM File</th>
<th>ICS Lookup</th>
<th>Oracle Sales Cloud Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AccountType</td>
<td>FINANCIALACCOUNT_ ACCOUNTTYPECODE. dvm</td>
<td>FINACCOUNT_ ACCOUNTTYPE</td>
<td>_ _ ORAFS__Type_c</td>
<td>Contains mapping for Financial Account Type.</td>
</tr>
<tr>
<td>AccountStatus</td>
<td>FINANCIALACCOUNT_ STATUSCODE. dvm</td>
<td>FINACCOUNT_ STATUS</td>
<td>_ _ ORAFS__Status_c</td>
<td>Contains mapping for Financial Account Status.</td>
</tr>
<tr>
<td>AssetRelationship</td>
<td>FINANCIALACCOUNT_ PERSON_HOLDER_ RELATIONSHIP_ CODE.dvm</td>
<td>Not applicable</td>
<td>_ _ ORAFS__Joint_c</td>
<td>Contains mapping for Financial Account to Contact Relationship.</td>
</tr>
<tr>
<td>AssetRelation</td>
<td>FINANCIALACCOUNT_ ORG_HOLDER_ RELATIONSHIP_ CODE.dvm</td>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Not applicable</td>
</tr>
<tr>
<td>AccountType</td>
<td>FINANCIALACCOUNT_ TYPECODE.dvm</td>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Not applicable</td>
</tr>
<tr>
<td>BillingOption</td>
<td>FINANCIALACCOUNT_ BILLINGCODE. dvm</td>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Relationship</td>
<td>FINANCIALACCOUNT_ RELATIONSHIPCODE. dvm</td>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Currency</td>
<td>CURRENCY_ CODE.dvm</td>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Not applicable</td>
</tr>
<tr>
<td>MM</td>
<td>CONTACT_ SALUTATION. dvm</td>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Not applicable</td>
</tr>
</tbody>
</table>

**Note:** For ICS Lookups and AIA Domain Value Map (DVM) files, the source and target values must have one to one mapping. For example, in STATE_LOOKUP in ICS, multiple entries must not be configured for the same state value since it would cause unique constraint issue.
Understanding Address Update Synchronization Between Oracle Sales Cloud and Siebel UCM

Address synchronization between Oracle Sales Cloud and Siebel UCM for the Oracle Banking Platform integration ensures that Oracle Sales Cloud supports the same address feature in Siebel UCM that Oracle Banking Platform identifies in Siebel UCM. Address synchronization is applicable for both commercial banking and retail banking.

Oracle Sales Cloud

Following is the process flow:

1. Sign into Oracle Sales Cloud as a Commercial Banker.
2. From the landing page, click Sales.
3. Click Companies.
   
   Accounts page opens. The list of companies are displayed.
4. Click any company name.
   
   Edit Account Overview page opens.
5. Click Profile tab.
   
   Edit Account Profile page opens. View Address section. Addresses are displayed in the Address section.
6. Click the address displayed.
   
   Edit Address page opens. An address is already available. Country, City, Address Line 1, Address Line 2, Address Line 3, City, State, and Postal Code details are displayed.
7. Edit address line 1 and click OK.
   
   Edit Account Profile page opens.
8. Click Save.

Siebel UCM

Following is the process flow:

1. Sign into Siebel UCM.
2. After signing into Siebel UCM, click Navigator > All Accounts.
   
   List of accounts are displayed.
3. Select Name from Accounts List Applet drop-down list and type the name of the account that you want to search.
4. Click Accounts: Go icon.
   
   The account that you searched for, is displayed.
5. Click the account.
   
   The account page is displayed.
6. Click Contacts drop-down list.
7. Click Addresses.
8. Click **Address Profile** drop-down list, click **More Info**.
   The account page is displayed. In the **Address** field, view the address.

9. Click **Account Form** Applet icon corresponding to the address.
   Account Addresses window opens. In the **Selected** section, view the address that you updated in Oracle Sales Cloud.

**Note:** Address exists for an account or a contact in both Oracle Sales Cloud and Siebel UCM. When you make an address change for these accounts or contacts that are synchronized between Oracle Sales Cloud and Siebel UCM in one of these two applications, then automatically the address is updated in the other application and vice-versa.

### Party Synchronization

#### Party Status Synchronization

A party is any physical entity, such as a person, organization or group, that the deploying company has an interest in tracking. Use this topic to understand how party status, also known as Due Diligence or KYC status, synchronization happens across all the applications involved in this integration.

#### Understanding Customer Onboarding Stages

There are five distinct onboarding stages that a typical financial services customer goes through. The following table considers the scenario when a request is originated from Oracle Banking Platform, and explains all the stages and the actions taken in Oracle Sales Cloud at each stage.

<table>
<thead>
<tr>
<th>Status in Oracle Banking Platform</th>
<th>Oracle Sales Cloud</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>New</td>
<td>CGUID is not set.</td>
<td>At this stage, bank only has a record of a contact with a customer.</td>
</tr>
<tr>
<td>Registered</td>
<td>CGUID is set to not NULL.</td>
<td>The bank allocates a unique identifier to the customer.</td>
</tr>
<tr>
<td>Mastered</td>
<td>Mastered flag set to True.</td>
<td>The bank knows minimal information necessary to uniquely identify this customer, in the bank records. However this information is not sufficient to identify the customer in the real world. This information is synchronized with Siebel UCM, which means that the data for the customer is mastered.</td>
</tr>
<tr>
<td>On Boarded</td>
<td>Onboarding Status is set to Done.</td>
<td>The bank at this stage has minimal information necessary to positively identify the customer. However, this may not be sufficient to build financial relationship with the customer.</td>
</tr>
</tbody>
</table>
Oracle Sales Cloud
Integrating Financial Services with Oracle Banking Platform
and Siebel UCM

Chapter 3
Understanding Oracle Sales for Financials Integration

<table>
<thead>
<tr>
<th>Status in Oracle Banking Platform</th>
<th>Oracle Sales Cloud</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Known</td>
<td>KYC status is set.</td>
<td>A known customer is the one with whom the bank can diligently have financial relationship.</td>
</tr>
</tbody>
</table>

**Note:** CGUID in Oracle Banking Platform and Oracle Sales Cloud is a unique identifier assigned to each customer. It is known as UUID in Siebel UCM.

To understand how various terminologies in different applications are mapped, see Understanding Terminologies Across Applications.

**Mapping of Onboarding and KYC Status**

ICS lookup and AIA DVM files are used to map fields and column names, and their values for Oracle Sales Cloud and Siebel UCM.

- Siebel UCM Field Name: FinalVerificationStatus
- Siebel UCM Column Name: FINAL_STATUS_CD
- Oracle Sales Cloud Field Name: __ORAFS__KYCStatus_c
- ICS Lookup: KYCSTATUS_LOOKUP
- AIA DVM File: CUSTOMERPARTY_ONBOARDINGSTATUS.dvm

The following table contains mapping of KYC status values for all the three applications.

<table>
<thead>
<tr>
<th>Oracle Sales Cloud</th>
<th>AIA</th>
<th>Siebel UCM</th>
</tr>
</thead>
<tbody>
<tr>
<td>ORA_AFS_CONFIRMED</td>
<td>CONFIRMED</td>
<td>CNF</td>
</tr>
<tr>
<td>ORA_AFS_UNCONFIRMED</td>
<td>UNCONFIRMED</td>
<td>UNC</td>
</tr>
<tr>
<td>ORA_AFS_NOT_APPlicable</td>
<td>NOT_APPlicable</td>
<td>NAP</td>
</tr>
<tr>
<td>ORA_AFS_EXPIRED</td>
<td>EXPIRED</td>
<td>EXP</td>
</tr>
</tbody>
</table>

If Due Diligence Status in Oracle Banking Platform is NULL then Onboarding Status in Oracle Sales Cloud is set to PENDING. For any other value, Onboarding status in Oracle Sales Cloud is set to SUCCESS.

**Party Type Synchronization**

For the accounts created in Siebel UCM, the account Type (Type field) is automatically set to Trust, and Party Type field in Oracle Sales Cloud is set to Trust. For any other value in Siebel UCM, the Party Type field in Oracle Sales Cloud is set to Organization. The default value for Party Type in Oracle Sales Cloud is Organization. The value in Siebel UCM for Account Party Type Code is set to Organization for all the account records originating from Oracle Sales Cloud. For Contact Party Type Code, the value in Siebel UCM is set to Person.
Party Type Field and Column Mapping

The following table lists Party Type field and column mapping across applications for accounts.

<table>
<thead>
<tr>
<th>Application</th>
<th>Field Name</th>
<th>Column Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Sales Cloud</td>
<td>NA</td>
<td>_ORAFS__PartyType_c</td>
</tr>
<tr>
<td>Oracle Banking Platform</td>
<td>NA</td>
<td>IntegrationPartyOnBoardingDTO. PartyOnBoardingDTO. OrganizationDTO. PartyType</td>
</tr>
<tr>
<td>Siebel UCM</td>
<td>Party Type Code</td>
<td>PARTY_TYPE_CD</td>
</tr>
</tbody>
</table>

The following table lists Party Type field and column mapping across applications for contacts.

<table>
<thead>
<tr>
<th>Application</th>
<th>Field Name</th>
<th>Column Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Sales Cloud</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Oracle Banking Platform</td>
<td>NA</td>
<td>IntegrationPartyOnBoardingDTO. IndividualDTO. IndividualDemographicsDTO. PartyDemographicsKeyDTO. PartyType</td>
</tr>
<tr>
<td>Siebel UCM</td>
<td>Party Type Code</td>
<td>PARTY_TYPE_CD</td>
</tr>
</tbody>
</table>

Names of ICS lookup and AIA DVM (Domain Value Map) files:

- ICS Lookup: PARTYTYPE_LOOKUP
- AIA DVM file: CUSTOMERPARTY_ACCOUNTTYPECODE.dvm

⚠️ Caution: Any value outside the supported mappings must be added in respective maps for it to be synchronized. Missing mapping for required fields leads to synchronization failure or data mismatch if default value is not present in the respective system. Partial mapping for required fields also leads to synchronization failure or data mismatch.

Mapping Party Type Code for Trust

You must map the Party Type code for Trust from Siebel UCM to ICS and AIA using the following steps:

1. Find the party type code for Trust used by Oracle Banking Platform in Siebel UCM:
   a. Sign into Siebel UCM as an administrator.
   b. Navigate to Site Map > Administration - Data > List of values.
   c. Query for List of Values Type ACCOUNT_TYPE and Display Value Trust.
   d. Note the LIC for this List of Values entry.

2. To map the Party Type code in AIA:
   a. Edit the AIA .dvm file to update the mapping for Party Type CUSTOMERPARTY_ACCOUNTTYPECODE.dvm.
3. To map Party Type in ICS:
   a. Sign into ICS.
   c. Edit PARTYTYPE_LOOKUP.
   d. Ensure that the value of SOAP for Trust is same as the mapping for COMMON in AIA. Edit the value if required and click Save.

   **Note:** Additionally, Siebel UCM also has Type field for accounts and contacts. For Accounts, the Type field is S_ORG_EXT.OU_TYPE_CD, and for contacts, it is S_CONTACT.CON_CD.

### Party Relationship Synchronization

In Siebel UCM, party relationships are created by associating contacts to an account using OrganizationService web service. These party relationships are displayed on the Relationship Hierarchy tab of the accounts. However, for associated contacts, the account is not displayed in their Relationship Hierarchy tab. Similarly for contacts, party relationships are created by associating accounts to the contact, using PersonService web service. These relationships are displayed on the Relationship Hierarchy tab for contacts, however, for associated accounts, the contact is not displayed on their Relationship Hierarchy tab. When such party relationships are synchronized to Oracle Sales Cloud, the associations can be on Relationships tab of both accounts and contacts, unlike in Siebel UCM.

As per Trading Community Architecture model that is used by Oracle Sales Cloud, if you are creating any relationship between an organization and contact, the first relationship is considered as a primary relationship, irrespective of the type of relationship. So, if you are adding the first relationship from the Relationship tab of the account, it is considered as a primary contact. You must create the first contact from the Contact subtab which acts as a primary contact, and then create other relationships.

### Mapping of List of Values

The list of values are mapped across application as follows:

- **CS Lookup code:** RELATIONSHIPTYPE_LOOKUP
- **AIA Domain Value Map file:** CUSTOMERPARTY_RELATIONSHIPCODE.dvm
- **Siebel UCM field name:** RelationshipType
- **Siebel UCM column name:** REL_TYPE_CD
- **Oracle Sales Cloud field:** RelationshipCode

In ICS Lookup and AIA .dvm file, the values are mapped based on the following table:

<table>
<thead>
<tr>
<th>Oracle Sales Cloud</th>
<th>AIA</th>
<th>Siebel UCM</th>
</tr>
</thead>
<tbody>
<tr>
<td>BOARD_MEMBER</td>
<td>Board Member</td>
<td>Board Member</td>
</tr>
</tbody>
</table>
Creating New Relationship Types in Oracle Sales Cloud and Siebel UCM

To create new relationships in Siebel UCM:

1. Sign into Siebel UCM as an administrator.
2. Navigate to Site Map > Administration - Data > List of Values
3. Query for List of Value Type `PARTY_RELATIONSHIP_TYPE`
4. Add a new row for the new value you want to add.
5. Click Clear Cache.

To create a new relationship type in Oracle Sales Cloud, see Define Relationships in Oracle Sales Cloud Implementing Customer Data Management.

Note: You must create appropriate mappings for any new relationship types in respective ICS and AIA lookups or .dvm files. Synchronization fails if relationship type is not mapped or partially mapped.

Error Handling Scenarios

Two error notifications are configured for data flows originating from Siebel UCM to Oracle Sales Cloud:

- Skipped Account-Contact or Contact-Account relationship synchronization, when the record is not synchronized.
- Skipped Account-Contact or Contact-Account relationship synchronization, when the role is not mapped.

Additional Considerations for Relationship Roles Synchronization

When an existing relationship role is updated in Siebel UCM, in Oracle Sales Cloud, existing relationship role that was synchronized is retained and a new relationship with a new role is synchronized to Oracle Sales Cloud.

Related Topics

- Creating Relationship Types: Worked Example

Mapping of Fields across Oracle Banking Platform, Siebel UCM, and Oracle Sales Cloud

Mapping of fields is required across the integrating applications Oracle Banking Platform, Siebel UCM, and Oracle Sales Cloud.

Mapping consists of the following:

- Account field mapping
- Contact field mapping
- Mapping of address fields across Oracle Sales Cloud and Siebel UCM.

Account Field Mapping

The following table lists the required attributes for mapping account fields between Oracle Sales Cloud and Siebel UCM:
<table>
<thead>
<tr>
<th>Oracle Sales Cloud Field Name</th>
<th>Siebel UCM Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>OrganizationName</td>
<td>Account.Name</td>
</tr>
<tr>
<td>OrigSystemReference of the Resource for associated OwnerPartyId</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Party ID</td>
<td>Not applicable</td>
</tr>
<tr>
<td>/ onEvent/ getAccountResponse/ result/ OrganizationDEO_ __ ORAFS__</td>
<td>Account.Type</td>
</tr>
<tr>
<td>_PartyType_c</td>
<td></td>
</tr>
<tr>
<td>Type</td>
<td>Not applicable</td>
</tr>
<tr>
<td>AddressLine1</td>
<td>Account.UcmAccountAddress.StreetAddress</td>
</tr>
<tr>
<td>AddressLine2</td>
<td>Account.UcmAccountAddress.StreetAddress2</td>
</tr>
<tr>
<td>City</td>
<td>Account.UcmAccountAddress.City</td>
</tr>
<tr>
<td>State</td>
<td>Account.UcmAccountAddress.State</td>
</tr>
<tr>
<td>Country</td>
<td>Account.UcmAccountAddress.Country</td>
</tr>
<tr>
<td>Postal Code</td>
<td>Account.UcmAccountAddress.PostalCode</td>
</tr>
<tr>
<td>OrigSystemReference of Address OSR</td>
<td>Not applicable</td>
</tr>
<tr>
<td>REG</td>
<td>Not applicable</td>
</tr>
<tr>
<td>OrganizationDEO_ Mastered_c</td>
<td>true</td>
</tr>
<tr>
<td>OrganizationDEO_ __ ORAFS__ _CGUID_c</td>
<td>Account.PartyUId</td>
</tr>
<tr>
<td>ORAFS__ OnboardingStatus_ c</td>
<td>Account.UcmAccountDueDiligenceHistory. DueDiligenceStatus</td>
</tr>
<tr>
<td>ORAFS__ KYCStatus_c</td>
<td>Account.UcmAccountDueDiligenceHistory. UcmAccountIdentityVerificationHistory. FinalVerificationStatus</td>
</tr>
<tr>
<td>SourceSystem</td>
<td>AIA</td>
</tr>
<tr>
<td>Sales Team Member account, salesteammember, ResourceId</td>
<td>Account.RelatedSalesRep. PositionIntegrationId</td>
</tr>
<tr>
<td>Owner</td>
<td>Account.RelatedSalesRep. PositionIntegrationId</td>
</tr>
</tbody>
</table>
Contact Field Mapping

The following table lists the required attributes for mapping Contact fields between Oracle Sales Cloud and Siebel UCM:

<table>
<thead>
<tr>
<th>Oracle Sales Cloud Field Name</th>
<th>Siebel UCM Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>FirstName</td>
<td>Contact/ FirstName</td>
</tr>
<tr>
<td>LastName</td>
<td>Contact/ LastName</td>
</tr>
<tr>
<td>SourceSystem</td>
<td>AIA</td>
</tr>
<tr>
<td>SourceSystemReferenceValue</td>
<td>Contact/ PartyUId</td>
</tr>
<tr>
<td>OwnerPartyId</td>
<td>Contact/ ContactPosition/ PositionIntegrationId</td>
</tr>
<tr>
<td>PersonDEO_ _ _ ORAFS_ _ _ CGUID_c</td>
<td>Contact/ PartyUId</td>
</tr>
<tr>
<td>PersonDEO_ Mastered_c</td>
<td>true</td>
</tr>
<tr>
<td>Address Source System</td>
<td>AIA</td>
</tr>
<tr>
<td>Address SourceSystemReferenceValue</td>
<td>Not applicable</td>
</tr>
<tr>
<td>AddressLine1</td>
<td>ListOfUCMHEConstituentAddress/ ns0UCMHEConstituentAddress/ StreetAddress</td>
</tr>
<tr>
<td>AddressLine2</td>
<td>ListOfUCMHEConstituentAddress/ ns0UCMHEConstituentAddress/ StreetAddress2</td>
</tr>
<tr>
<td>City</td>
<td>ListOfUCMHEConstituentAddress/ ns0UCMHEConstituentAddress/ City</td>
</tr>
<tr>
<td>State</td>
<td>ListOfUCMHEConstituentAddress/ ns0UCMHEConstituentAddress/ State</td>
</tr>
<tr>
<td>Country</td>
<td>ListOfUCMHEConstituentAddress/ ns0UCMHEConstituentAddress/ Country</td>
</tr>
<tr>
<td>PostalCode</td>
<td>ListOfUCMHEConstituentAddress/ ns0UCMHEConstituentAddress/ PostalCode</td>
</tr>
<tr>
<td>ResourcedId</td>
<td>Contact/ ContactPosition/ PositionIntegrationId</td>
</tr>
<tr>
<td>ORAFS_ _ _ OnboardingStatus_ c</td>
<td>Contact. UcmContactDueDiligenceHistory. DueDiligenceStatus</td>
</tr>
<tr>
<td>ORAFS_ _ _ KYCStatus_ c</td>
<td>Contact. UcmContactDueDiligenceHistory. UcmContactIdentityVerificationHistory. FinalVerificationStatus</td>
</tr>
<tr>
<td>Phone</td>
<td>Contact/ns0: ListOfContactAlternatePhone/ ns0ContactAlternatePhone/ AlternatePhone</td>
</tr>
</tbody>
</table>
### Mapping of Address Fields Across Oracle Sales Cloud and Siebel UCM

The following table lists the corresponding field attributes across Oracle Sales Cloud and Siebel UCM for addresses of persons and organizations:

<table>
<thead>
<tr>
<th>Oracle Sales Cloud Field Name (Address)</th>
<th>Siebel UCM Field Name (Organization.UcmAccountAddress)</th>
<th>Siebel UCM Field Name (Person.UCMHEConstituentAddress)</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddressLine1</td>
<td>StreetAddress</td>
<td>StreetAddress</td>
</tr>
<tr>
<td>AddressLine2</td>
<td>StreetAddress2</td>
<td>StreetAddress2</td>
</tr>
<tr>
<td>AddressLine3</td>
<td>StreetAddress3</td>
<td>StreetAddress3</td>
</tr>
<tr>
<td>AddressLine4</td>
<td>StreetAddress4</td>
<td>StreetAddress4</td>
</tr>
<tr>
<td>City</td>
<td>City</td>
<td>City</td>
</tr>
<tr>
<td>State</td>
<td>State</td>
<td>State</td>
</tr>
<tr>
<td>Country</td>
<td>Country</td>
<td>Country</td>
</tr>
<tr>
<td>Postal Code</td>
<td>PostalCode</td>
<td>PostalCode</td>
</tr>
</tbody>
</table>
4 Configuring Oracle Sales Cloud

Defining Source System and Source System Entities

To maintain cross-reference between Oracle Sales Cloud records and the records created in AIA (Application Integration Architecture), you must define the source system for AIA in Oracle Sales Cloud. Use this topic to verify if a source system for AIA already exists in Oracle Sales Cloud or create one and manage the related source system entities.

Defining Source System for AIA

Source systems enable users to identify the source of data being imported. You can specify whether the source system is a spoke system, such as a legacy system, or a purchased system, such as data from a third party provider. You can also specify what types of entities the source system contains. For example, you can specify that a source system contains customer data.

Perform the following steps to verify and edit the existing source system definition or create a new definition:

1. Sign in to Oracle Sales Cloud as an administrator.
2. In the Setup and Maintenance work area, go to the following:
   - Functional Area: Application Extensions
   - Task: Manage Trading Community Source Systems
3. On the Manage Trading Community Source Systems page, search for Source System with Code AIA.
4. After you find the definition for AIA, perform the following steps:
   a. Select the row from the search results and click Edit icon.
   b. Select Enable for Trading Community Member check box, if not selected already.
   c. Click Save and Close.
5. If the source system definition for AIA is not found in Search Results, perform the following steps to create a new definition:
   a. Click Actions > Create in the Search Results region, on the Manage Trading Community Source Systems page.
   b. Specify the following on the Create Source System page:
      - Code: AIA
      - Name: Application Integration Architecture
      - Description: Maintains cross-reference between Oracle Sales Cloud records and the records created in AIA.
   c. Select Spoke from the Type drop-down list.
   d. Select check boxes for Enable for Items, Enable for Trading Community Members, Enable for Order Orchestration and Planning, and Enable for Assets.
   e. Click Save and Close.
6. Click Done on the Manage Trading Community Source Systems page.
Editing Source System Entity

Source system entities are entities that you can import using a specific source system, such as addresses and parties. When you import data from a source system, all the entities in the source system data are also imported. You can select multiple source references in the Manage Source System Entities task to allow multiple source system records to map to a single record.

You must configure the related source system entity so as to allow multiple source references from AIA based on the following steps:

1. Sign in to Oracle Sales Cloud as an administrator.
2. In the Setup and Maintenance work area, go to the following:
   - Functional Area: Application Extensions
   - Task: Manage Source System Entities
3. Search for AIA in the Code column on the Manage Source System Entities page and select the record from the search results.
4. Deselect all the check boxes for Allow Multiple Source References in the Entities region.
5. Click Save and Close.

Enabling Trading Community Events in Profile Options

This topic explains how to enable trading community events in profile options.

Perform the following steps to enable trading community events in profile options:

1. Sign in to Oracle Sales Cloud as an administrator.
2. In the Setup and Maintenance work area, go to the following:
   - Functional Area: Application Extensions
   - Task: Manage Trading Community Common Profile Options
3. On the Manage Trading Community Common Profile Options page:
   a. Select the profile options from the following table.
   b. In the Profile Values region, select the profile level and the profile value for each of the profile option listed in the table.
   c. Click Save and Close.

<table>
<thead>
<tr>
<th>Profile Option</th>
<th>Profile Level</th>
<th>Profile Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>HZ_ENABLE_EVENT_TRACKING</td>
<td>Site</td>
<td>Yes</td>
</tr>
<tr>
<td>HZ_INVOKE_OBJ_WF_ON_TRACKING</td>
<td>Site</td>
<td>Yes</td>
</tr>
</tbody>
</table>

4. On the Manage Trading Community Common Profile Options page, click Done.
Setting Up Integration Users and Roles

To trigger an Oracle Sales Cloud Service Catalog or Event Catalog web services from Oracle Integration Cloud Service, you must create a unique integration user and assign a unique integration role to it. This topic provides steps to set up this integration user and role.

⚠️ **Caution:** You must have privileges sufficient to create new roles, such as IT Security Manager.

**Creating Integration User**

To create an integration user account, perform the following steps:

1. Sign in to Oracle Sales cloud.
2. Click **Navigator > My Team > Manager Users**.
3. On the Manager Users tab, click **Create** icon.
4. On the Create User page:
   - Enter information using following table.
   - Provide valid **Email**, **Hire Date**, **Legal Organization**, and **Business Unit**.
   - In the User Details region, select **Send User Name and Password** check box.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value or Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Last Name</td>
<td>FUSION_APPS_ICS_APPID</td>
</tr>
<tr>
<td>User Name</td>
<td>FUSION_APPS_ICS_APPID</td>
</tr>
<tr>
<td>Person Type</td>
<td>Employee</td>
</tr>
<tr>
<td>Resource Role</td>
<td>Sales Person</td>
</tr>
</tbody>
</table>

5. Click **Save and Close**. An email is sent to the user after the user record is created.
6. Sign out of Oracle Sales Cloud and sign in again using **FUSION_APPS_ICS_APPID** as user name and the temporary password provided in the notification email.
7. Change the password when prompted during the first time sign in. The Oracle Sales Cloud Welcome page is displayed.
Enabling Add User Button

Following are the steps to enable the Add User button:

1. Sign in to Oracle Sales Cloud as a user with IT_Security_Manager role.
2. Click Navigator > Security Console.
3. Click Administration.
4. Select Enable edit of user role membership check box.

Setting Password for Integration User

Optionally, to set a password for the integration user, perform the following steps:

1. Sign in to Oracle Sales Cloud.
2. Click Navigator > Security Console and navigate to Users tab.
3. On the User Accounts page, search for user FUSION_APPS_ICS_APPID.
4. Click User Login link for FUSION_APPS_ICS_APPID.
5. On the User Account Details page, click Edit.
7. On the Reset Password page, select Manually change the password.
9. Click Reset Password.
10. Sign out of Oracle Sales Cloud.

Creating and Assigning Integration Roles

To create and assign integration roles:

1. Sign in to Oracle Sales Cloud.
2. Click Navigator > Security Console.
4. In the Basic Information step, enter the following and click Next.
   - Role Name: ICS Integration Role
   - Role Code: INT_ICS_Integration_Role
   - Role Category: CRM - Job Roles
   - Description: Custom role for accessing Oracle Sales Cloud services catalog.

5. In the Function Security Policies step:
   - Click Add Function Security Policy.
   - On the Add Function Security Policy page, search for FND_MANAGE_CATALOG_SERVICE_PRIV privilege, and select the privilege.
   - Click Add Privilege to Role and click OK on the confirmation message.
   - In the Function Security Policies step, click Next.

7. In the Role Hierarchy step:
   - Click Add Role icon.
b. On the Add Role Membership page, search for the following Role Memberships and click **Add Role Membership**:
   - Sales Administrator
   - SOA Operator
   - Custom Objects Administration

c. Repeat following steps for each of the role membership:
   i. Select **Sales Administrator** from the Search Results. (Job Code: `ORA_ZBS_SALES_ADMINISTRATOR_JOB`)
   ii. Click **Add Role Membership**.
   iii. Click **OK** on the confirmation message.

d. On the Add Role Membership page, click **Close** icon.
e. In the Role Hierarchy step, click **Next**.

8. In the Users step:
   a. Click **Add User**.
   b. On the Add User page, search for `FUSION_APPS_ICS_APPID` user.
   c. Select the user from the Search Results and click **Add Users to Role**.
   d. Click **OK** on the confirmation message.
   e. On the Add User page, click **Close** icon.
   f. In the Users step, click **Next**.

9. In the Summary and Impact Report step, review the details and click **Save and Close**.

**Note:** The sales administrator role has out of the box access (OOTB) to personally identifiable information (PII) DSPs. If you intend to provide the PII access to an integration user, you can provide it using security console. For more information, see Oracle Sales Cloud Assigning Personally Identifiable Information Privileges Using Security Console (My Oracle Support Article Document ID 2224401.1).

**Related Topics**
- Sales Cloud Users and Role Provisioning
- Oracle Sales Cloud Assigning Personally Identifiable Information Privileges Using Security Console

## Creating a CSF Key for Event Subscriptions

You must create a CSF key to subscribe to events in Oracle Sales Cloud. This key is required by the event handler framework when it invokes the integration. The credentials of the integration are managed by the CSF key. Create the CSF key in Oracle SOA Composer.

**Note:** There is only one SOA instance in the release. The Oracle Integration Cloud Service password may expire periodically. Your application administrator must contact the Oracle Integration Cloud Service administrator to get the refreshed user credentials. The application administrator must update the CSF key when this password is refreshed.

To ensure that you specify information correctly while creating the CSF key, see Configuring Oracle Sales Cloud for Event Subscriptions section in the Integration Cloud Service guide.
Related Topics
- Configuring Oracle Sales Cloud for Event Subscriptions

Phone Number and Email Synchronization for Contact Points: Overview

When contact points are synchronized from Siebel UCM to Oracle Sales Cloud, you must setup phone number formats for different countries in Oracle Sales Cloud to synchronize the phone numbers in correct format for country code, area code, and phone number.

Note:
- The country code is displayed for the phone numbers where the country code is different from the country specified in Siebel UCM parameter user locale.
- For more information to define phone number formats for different countries, see Define Contact Points in Implementing Customer Data Management guide.
- Siebel UCM recognizes and maintains phone numbers together with extension numbers as a single number. The complete phone number maintained in Siebel UCM is communicated to Oracle Sales Cloud. If a matching phone number format is not available in Oracle Sales Cloud, then the complete number after the country code is considered as the phone number. Area code and extension values are left empty.

Synchronization fails for both phone and email when the number of characters exceed the limit in the target Oracle Sales Cloud application or in the Siebel UCM application.

Note: For a contact, the primary flag is set for the phone and email first created in Oracle Sales Cloud, even if the primary flag is not set for the same phone or the email record in Siebel UCM.

Related Topics
- Telephone Numbering Plan: Explained
- Updating Telephone Numbering Plans: Procedure

Configuring Security Policies for Custom Objects

A security policy defines who can access a custom object data at runtime. When a custom object is created, access is granted only to a default duty role specified by the application. Any additional access, either at the object level or role level, must be granted manually.

The following table indicates the access provided to different roles for custom objects.

<table>
<thead>
<tr>
<th>Object</th>
<th>Retail Banker</th>
<th>Retail Bank Manager</th>
<th>Commercial Banker</th>
<th>Commercial Bank Manager</th>
<th>Sales Admin for Financial Services</th>
<th>Application Implementation Consultant</th>
</tr>
</thead>
<tbody>
<tr>
<td>XREF</td>
<td>View, View All</td>
<td>View, View All</td>
<td>View, View All</td>
<td>View, View All</td>
<td>Full</td>
<td>NA</td>
</tr>
</tbody>
</table>
To configure security for objects:

1. Sign in to Oracle Sales Cloud as an administrator.
2. Ensure that you are in an active sandbox.
3. Navigate to the Application Composer.
4. Depending on the object for which you are configuring security, select the application as mentioned in the preceding table from the Application drop-down list.
5. Expand object for which you want to configure security.
6. On the Define Policies page, configure the security for each of the roles.
7. Click Save and Close.
8. Publish the sandbox.

### Changing Administrator Profile Values

You can change administrator profile values.

Following are the steps:

1. Sign in to Oracle Sales Cloud as an administrator.
2. Navigate to the Setup and Maintenance work area.
   - Functional Area: Application Extensions
   - Task: Manage Administrator Profile Values
3. In the Search Results, click Manage Administrator Profile Values link.
4. For Profile Option Code field, enter value as MOO_CLOSE_WINLOSS_REQD.
5. Click Search.
6. If the Profile Level is displayed as Site, change Profile Value to No.
7. Click Save.
8. If the Profile Value is not displayed, then create a new profile value in the Profile Option: Profile Values section by selecting New in the Actions drop-down list.
9. In the Profile Level drop-down list, select Site.
10. For Profile Value, select value as No.
11. Click Save.
12. Search for Profile Option Code as MOO_CLOSE_COMP_REQD.
13. Click Search.
Creating Legal Entity Map Record to Store Bank Code

The bank code of a legal entity must be passed from Oracle Sales Cloud to Oracle Banking Platform to submit an opportunity to the Oracle Banking Platform. The bank code must be stored in Oracle Sales Cloud to pass the bank code of the legal entity.

We must create a legal entity in Oracle Sales Cloud for legal entity in Oracle Banking Platform and store the bank code and the legal entity associated with it. Complete the following steps to configure the legal entity map record:

1. Create the legal entity in Oracle Sales Cloud.

   **Note:** For more information on creating legal entities, see
   - Oracle Sales Cloud Implementing Sales Cloud guide
   - Creating Legal Entities and Business Units in Oracle Sales Cloud section of the Integrating Financial Services with Oracle Banking Platform and Siebel UCM guide.

2. Obtain the bank code of the legal entity in Oracle Banking Platform. For more information, see Oracle Banking Platform guide.

3. Use the Custom Object web service to create the legal entity map record to store the bank code of the legal entity. For more information, see Creating Oracle Sales Cloud Custom Objects and Fields to Support Extensibility chapter of the PaaS-SaaS Integration guide.

   **Object Name:** __ORAFS__LegalEntityMap_c

   The following table contains information about field names and their corresponding values.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Record Name</td>
<td>Legal Entity Name</td>
</tr>
<tr>
<td>__ORAFS__BankCode_c</td>
<td>Bank Code of the legal entity in Oracle Banking Platform</td>
</tr>
</tbody>
</table>

**Related Topics**
- PaaS-SaaS Integration
- Oracle Banking Platform Installation

Configuring Integration Cloud Service Web Service

Configure the Integration Cloud Service web service to submit opportunities to Oracle Banking Platform. To configure Integration Cloud Service web service to submit opportunities to Oracle Banking Platform:

1. Sign in to Oracle Sales Cloud as an administrator.
2. Ensure that you are in an active sandbox.
3. Navigate to the **Application Composer**.
4. Select **Web Services** from the Common Setup region.
5. Click **Create Web Service Reference** and select **Connection Type** as **SOAP**.
6. Provide the following information on the Create SOAP Web Service Connection page.
   a. **Name**: SubmitOpportunity
   b. **WSDL URL**: <Endpoint WSDL URL obtained from your integration for opportunity submission on the Integration Cloud Service>

   ![Note](image)
   To find out the correct Endpoint URL for the integration:
   i. Sign in to ICS and navigate to Integrations page.
   ii. Identify the integration that you must use for opportunity submission and click **Details** icon.
   iii. Copy the **Endpoint URL** displayed for that integration.

   c. Select the available service and port.
   d. Select **Invoke with separate user credentials over SSL security scheme**.
   e. Use the **ICS credentials** as **Credential key**.
   f. Disable the timestamp verification.
   g. Click **Save and Close**.

**Adding the Oracle Banking Platform Application URL**

To use Oracle Banking Platform for account origination flow and customer onboarding, you must configure the Oracle Banking Platform application URLs for application form, application tracker, customer onboarding, financial account transaction, and so on for the Manage Custom Setup Content task.

To add the Oracle Banking Platform application URL:
1. Sign in to Oracle Sales Cloud using Application Implementation Consultant role.
2. In the Setup and Maintenance work area, go to the following:
   - Functional Area: Application Extensions
   - Task: Manage Setup Content
3. Select **Manage Third Party Applications** from the Topology Definition region.
4. Click **Create** to add third party applications in the following table.

<table>
<thead>
<tr>
<th>Application Name</th>
<th>Full URL</th>
<th>Partner Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>OBP_Application_Form</td>
<td>&lt;OBP Application Form URL&gt;</td>
<td>OBP</td>
</tr>
<tr>
<td>OBP_Application_Tracker</td>
<td>&lt;OBP Application Tracker URL&gt;</td>
<td>OBP</td>
</tr>
<tr>
<td>OBP_Onboard_Customer</td>
<td>&lt;OBP Customer Onboarding URL&gt;</td>
<td>OBP</td>
</tr>
<tr>
<td>OBP_Servicing</td>
<td>&lt;OBP Financial Account Transaction Mashup URL&gt;</td>
<td>OBP</td>
</tr>
</tbody>
</table>
5. Click **Save and Close**.

### Configuring Page Layouts to Use Oracle Banking Platform and Third Party Servicing

Default layouts delivered with Oracle Sales Cloud are automatically set to inactive status and cannot be edited. You can configure the default layouts.

You can dynamically control the display of page layouts based on:

- Roles or privileges of the users
- Groovy expression
- Type of record

For more information and examples on how you can use page layouts, see the Oracle Sales Cloud Extending Sales guide.

To duplicate these pages, you must:

1. Duplicate the layouts provided.
2. Add the provided groovy script.
3. Associate the pages with specific Oracle Sales Cloud job roles.

### Page Layouts to Use Oracle Business Platform

Account origination and onboarding using Oracle Banking Platform requires the use of specific fields, buttons, and actions which are available in the layouts listed in the following table. You must duplicate and configure these specific layouts correctly.

<table>
<thead>
<tr>
<th>Object Type</th>
<th>Object</th>
<th>Landing Page Layout Name</th>
<th>Create Page Layout Name</th>
<th>Detail Page Layout Name</th>
<th>Role(s)</th>
<th>Expression</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard</td>
<td>Company</td>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Cx Fins Commercial Banking Layout</td>
<td>Commercial Banker, Commercial Bank Manager</td>
<td>Not applicable</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Commercial Banker, Commercial Bank Manager</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Custom</td>
<td>Product Servicing</td>
<td>Cx Fins Vertical Layout</td>
<td>Cx Fins Vertical Layout</td>
<td>Cx Fins Vertical Layout</td>
<td>Any</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Standard</td>
<td>Opportunity</td>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Cx Fins Commercial Banking Layout</td>
<td>Commercial Banker, Commercial Bank Manager</td>
<td>Not applicable</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Commercial Banker, Commercial Bank Manager</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Standard</td>
<td>Opportunity</td>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Cx Fins Retail Banking Layout</td>
<td>Retail Banker, Retail Bank Manager</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Standard</td>
<td>Household</td>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Cx Fins Vertical Layout</td>
<td>Retail Banker, Retail Bank Manager</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Object Type</td>
<td>Object</td>
<td>Landing Page Layout Name</td>
<td>Create Page Layout Name</td>
<td>Detail Page Layout Name</td>
<td>Role(s)</td>
<td>Expression</td>
</tr>
<tr>
<td>-------------</td>
<td>--------</td>
<td>--------------------------</td>
<td>-------------------------</td>
<td>-------------------------</td>
<td>---------</td>
<td>------------</td>
</tr>
<tr>
<td>Standard</td>
<td>Contact</td>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Cx Fins Retail Banking Layout</td>
<td>Retail Banker, Retail Bank Manager</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Standard</td>
<td>Contact</td>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Cx Fins Commercial Banking Layout</td>
<td>Commercial Banker, Commercial Bank Manager</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Custom</td>
<td>Financial Account</td>
<td>Financial Services Retail Banking Layout</td>
<td>FINS Retail Layout</td>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Custom</td>
<td>Financial Account</td>
<td>Not applicable</td>
<td>Not applicable</td>
<td>FINS Retail Layout</td>
<td>Retail Banker, Retail Bank Manager</td>
<td></td>
</tr>
</tbody>
</table>

```java
def TypeName = __ORAFS_SalesProductName
    def vo = newView('__ORAFS_ProductServicing_c')
def vc = vo.createViewCriteria()
def vcr = vc.createRow()
def vci1 = vcr.ensureCriteriaItem('RecordName')
vci1.setOperator('=')
vo.appendViewCriteria(vc)
vo.executeQuery()
if(vo.first() != null){
    def row = vo.first()
def servicing = row.getAttribute('__ORAFS_SalesProductServicing_c')
    if(servicing != null){
        return true
    }
}
return servicing.equals('ORA_AFS_SERVICE_OSC')
```
Configuring Oracle Sales Cloud

<table>
<thead>
<tr>
<th>Object Type</th>
<th>Object</th>
<th>Landing Page Layout Name</th>
<th>Create Page Layout Name</th>
<th>Detail Page Layout Name</th>
<th>Role(s)</th>
<th>Expression</th>
</tr>
</thead>
</table>
| Custom      | Financial Account | Not applicable           | Not applicable    | Commercial Banking Layout | Commercial Banker, Commercial Bank Manager | `vc11.setValue(TypeName)`  
  `vc.insertRow(vcr)`  
  `vo.appendViewCriteria()`  
  `vo.executeQuery()`  
  `if(vo.first() != null){`  
  `def row = vo.first()`  
  `def servicing = row.getAttribute('__ORAFS__ServicingSystem_c')`  
  `if(servicing != null){`  
  `return servicing.equals('ORA_AFS_SERVICE_OBP')`  
  `}`  
  `return false` |
| Custom      | Financial Account | Not applicable           | Not applicable    | Cx Fins Commercial Banking Layout | Commercial Banker, Commercial Bank Manager | `def TypeName = __ORAFS__SalesProductName_Obj_c?.__ORAFS__ProductType_cMeaning`  
  `def vo = newView('__ORAFS__ProductServicing_c')`  
  `def vc = vo.createViewCriteria()`  
  `def vcr = vc.createRow()`  
  `def vc11 = vcr.ensureCriteriaItem('RecordName')`  
  `vc11.setOperator('=')`  
  `vc11.setValue(TypeName)`  
  `vc.insertRow(vcr)`  
  `vo.appendViewCriteria(vc)`  
  `vo.executeQuery()`  
  `if(vo.first() != null){`  
  `def row = vo.first()`  
  `def servicing = row.getAttribute('__ORAFS__ServicingSystem_c')`  
  `if(servicing != null){`  
  `return true`  
  `}`  
  `return servicing.equals('ORA_AFS_SERVICE_OSC')`  
  `return true` |
### Page Layouts to Use Third Party Servicing for Product Types
The following table lists the layouts which must be duplicated to set up third party servicing.

<table>
<thead>
<tr>
<th>Object Type</th>
<th>Object</th>
<th>Layout Name</th>
<th>Role(s)</th>
<th>Expression</th>
</tr>
</thead>
<tbody>
<tr>
<td>Custom</td>
<td>Financial Account</td>
<td>FINS Retail Layout</td>
<td>Retail Banker, Retail Bank Manager</td>
<td>def TypeName = __ORAFS__SalesProductName_Obj_c?.__ORAFS__ProductType_cMeaning def vo = newView('__ORAFS__ProductServicing_c') def vc = vo.createViewCriteria() def vcr = vc.createRow() def vci1 = vc.ensureCriteriaItem('RecordName') vci1.setOperator('=') vci1.setValue(TypeName) vc.insertRow(vcr) vo.appendViewCriteria(vc) vo.executeQuery() if(vo.first() != null){ def row = vo.first() def servicing = row.getAttribute('__ORAFS__ServicingSystem_c') if(servicing != null){ return servicing.equals('ORA_AFS_SERVICE_THIRD_PARTY') } return false</td>
</tr>
</tbody>
</table>
Duplicating Layouts

Perform the following steps to duplicate pages and assign roles for each object listed in the preceding tables using the values specified:

1. Sign in to Oracle Sales Cloud as an administrator.
2. Ensure that you are in an active sandbox.
3. Navigate to the Application Composer.
4. Select the application from the Application list for the object you want to create a duplicate layout.
5. Expand Object Type node as mentioned in the table for the object you are duplicating layouts.
6. Expand the object for which you want to create duplicate layout.
7. Click Pages.
8. On the Simplified Pages tab, perform the following steps for the correct layout type, as provided in the preceding table.

   a. Select All layouts from the Layout Status drop-down list.
   b. Select the layout name specified in the preceding table in the Layout Name column.
   c. Select Actions > Duplicate Layout.
   d. On the Duplicate Layout dialog box, enter a New Layout Name for each page.
   e. Click Save and Close.
   f. Click Role Name in the Role field corresponding to the duplicate page layout you created. The Select: Roles dialog box opens.
   g. Select specific role for Roles.
   h. Select the roles you want to associate with the duplicated layout from the Available Roles list, and click OK.
Setting Up Product Servicing

Enable product servicing for the financial services Sales Admin and the Application Implementation Specialist roles. Disable product servicing for any other roles.

To disable product servicing:

1. Sign in to Oracle Sales Cloud using Sales Administrator role.
2. Ensure that you are in an active sandbox.
4. Expand Sales and select Product Servicing.
5. Perform the following steps on the Edit Page Entry page.
   a. Select EL Expression for the Navigator.
   b. Click Edit and provide the following expression and click Validate > OK.

\[
\text{#{securityContext.userInRole['AFS_SALES_ADMIN_FINANCIAL_SERVICES,ORA_ASM_APPLICATION_IMPLEMENTATION_CONSULTANT_JOB']}}
\]
6. Select No for the Springboard field.
7. Click Save and Close.
8. Publish the sandbox.

Oracle Sales Cloud pages are displayed by default for financial account transactions if the servicing system is not configured for a product type. A servicing system must be configured for each product type.

To configure servicing system for a product type:

1. Sign in to Oracle Sales Cloud using the financial service sales admin or the application implementation specialist role.
2. Navigate to the Product Servicing in Sales.
3. Click Create to configure a product type and select the corresponding service system.
4. Click Save and Close.

Adding Create Submission Function for Opportunity Submission

Add the Create Submission function to submit opportunities created in Oracle Sales Cloud to Oracle Banking Platform.

Perform the following steps to add the Create Submission function:

1. Sign in to Oracle Sales Cloud as an administrator.
2. Ensure that you are in an active sandbox.
3. Navigate to the Application Composer.
4. Select Sales check box.
5. Expand **Standard Objects > Opportunity** and click **Server Scripts**.
6. Navigate to the Object Functions tab and click **Actions > Add**.
7. On the Create Object Functions Page, provide the following information:
   - **Function Name**: CreateSubmission
   - **Return Type**: Void

   ✍️ **Note**: You can add any validations to the groovy script if required, during the opportunity submission.

   - In Edit Script, paste the following script and click **Save and Close**:
     ```groovy
def partyId = TargetPartyId
def OptyStatus = StatusCode;
def Desc = DescriptionText

//Validations for Submission Creation.
//Add more validations based on the requirement.
if (OptyStatus != 'OPEN')
{
    throw new oracle.jbo.ValidationException(adf.util.__ORAFS__getMessage('AFS_ORAFS_OPTY_STAT_OPEN'))
}

if (PartyType == 'GROUP')
{
    partyId = KeyContactId
    if (partyId == null)
    {
        throw new oracle.jbo.ValidationException(adf.util.__ORAFS__getMessage('AFS_ORAFS_PRIM_CONT_NT_FND'))
    }
}

if (__ORAFS__SubmissionID_c != null)
{
    throw new oracle.jbo.ValidationException(adf.util.__ORAFS__getMessage('AFS_ORAFS_SUBMIT_ONCE'))
}

if (partyId == null)
{
    throw new oracle.jbo.ValidationException(adf.util.__ORAFS__getMessage('AFS_ORAFS_CUST_NT_PRESENT'))
}

if (__ORAFS__IsPartyMastered_c == 'N')
{
    throw new oracle.jbo.ValidationException(adf.util.__ORAFS__getMessage('AFS_ORAFS_SERVICE_UNAVAILABLE'))
}

def childRev = ChildRevenue
if (!childRev.hasNext())
{
    throw new oracle.jbo.ValidationException(adf.util.__ORAFS__getMessage('AFS_ORAFS_REV_PRESENT'))
}

if (userID == null)
{
    throw new oracle.jbo.ValidationException(adf.util.__ORAFS__getMessage('AFS_ORAFS_USER_ID_PRESENT'))
}

//Set Product Status as In Progress
while (childRev.hasNext())
{
    // access the next row in the row iterator
}
```
def curRev = childRev.next()
curRev.setAttribute('__ORAFS__Status_c', 'ORA_AFS_IN_PROGRESS')
}

childRev.reset()
def ChildRevenues = [];
def voProduct = newView('Product');
def vcProduct = newViewCriteria(voProduct);
def vcrProduct = vcProduct.createRow();
def voProductGroup = newView('ProductGroupDetailBase');
def vcProductGroup = newViewCriteria(voProductGroup);
def vcrProductGroup = vcProductGroup.createRow();

//Iterate revenues to get the Product and Product Group details to be passed to OBP.
while (childRev.hasNext())
{
    // access the next row in the row iterator
    def curRev = childRev.next()

    // reference fields or object functions from the current row
    def revItemId = curRev.InventoryItemId;
def productGroupId = curRev.ProdGroupId
    def revAmount = curRev.RevnAmount
    def revID = curRev.RevnId
    def revRefId = curRev.__ORAFS__ProductRevenueReferenceID_c
    def ProductType
    def Revenue = []
def ProductIdRef
    def AllowableCurrencies

    //Get the Offer Id for the Item Id
    if (revItemId != null)
    {
        def vciProduct = vcrProduct.ensureCriteriaItem('InventoryItemID');
vciProduct.setOperator('=');
vciProduct.setValue(revItemId);
vcProduct.insertRow(vcrProduct);
voProduct.appendViewCriteria(vcProduct);
voProduct.executeQuery();
while (voProduct.hasNext())
{
    def ProductRec = voProduct.next();
    ProductIdRef = ProductRec.getAttribute('__ORAFS__OfferId_c')
    ProductType = ProductRec.getAttribute('__ORAFS__ProductType_c')

    //AllowableCurrencies Validation
    AllowableCurrencies = ProductRec.getAttribute('__ORAFS__AllowableCurrencies_c')
    if (AllowableCurrencies != null)
    {
        if (AllowableCurrencies.length() > 2)
        {
            if (! (AllowableCurrencies.substring(0, 3).equalsIgnoreCase('ALL')))
            {
                def curencies = AllowableCurrencies.split(';')
                if (CurrencyCode != null && !curencies.contains(CurrencyCode))
                {
                    def errmsg = adf.util.__ORAFS__getMessage('AFS_ORAFS_INVALID_CURRENCY')
def errmsg1 = errmsg.replaceFirst('\{PRODUCT_NAME\}', ProductRec.getAttribute('Name'))
def errmsg2 = errmsg1.replaceAll('\{OPPORTUNITY_CURRENCY\}', CurrencyCode)
throw new oracle.jbo.ValidationException(errmsg2)
                }
            }
        }
    }
}
//Pass only Products available in OBP.
if (ProductIdRef != null)
{
  //Product Details
  def revItems =
  [
    RevnId: revID,
    InventoryItemId: revItemID,
    __ORAFS__ProductType_c: ProductType,
    ProdGroupId: productGroupId,
    __ORAFS__ProductCrossReferenceID_c: ProductIdRef,
    __ORAFS__ProductRevenueReferenceID_c: revRefId,
    RevnAmount:
      [
        _text: revAmount
      ]
  ]
  ChildRevenues.add(revItems)
  else
  {
    curRev.setAttribute('__ORAFS__Status_c', '')
  }
  break;
}
else if (productGroupId != null)
{
  //Get the Product Group Id Reference
  def vciProductGroup = vcrProductGroup.ensureCriteriaItem('ProdGroupId1');
  vciProductGroup.setOperator('=');
  vciProductGroup.setValue(productGroupId);
  vcProductGroup.insertRow(vcrProductGroup);
  voProductGroup.appendViewCriteria(vcProductGroup);
  voProductGroup.executeQuery();
  while (voProductGroup.hasNext())
  {
    def PGRec = voProductGroup.next();
    ProductIdRef = PGRec.getAttribute('__ORAFS__ProductGroupReferenceId_c')
    ProductType = PGRec.getAttribute('__ORAFS__ProductType_c')
    //AllowableCurrencies Validation
    AllowableCurrencies = PGRec.getAttribute('__ORAFS__AllowableCurrencies_c')
    if (AllowableCurrencies != null)
    {
      if (AllowableCurrencies.length() > 2)
      {
        if (! (AllowableCurrencies.substring(0, 3).equalsIgnoreCase('ALL')))
        {
          def curencies = AllowableCurrencies.split(';')
          if (CurrencyCode != null && !curencies.contains(CurrencyCode))
          {
            def errmsg = adf.util.__ORAFS__getMessage('AFS_ORAFS_INVALID_CURRENCY_GRP')
            def errmsg1 = errmsg.replaceFirst('\{GROUP_NAME\}', curRev.ProdGroupName)
            def errmsg2 = errmsg1.replaceAll('\{OPPORTUNITY_CURRENCY\}', CurrencyCode)
            throw new oracle.jbo.ValidationException(errmsg2)
          }
        }
      }
    }
    if (ProductIdRef != null)
    {
      //Set Product Group Details.
def revItems =
[
  RevnId: 'PRDGRPLINEID01',
  InventoryItemID: revItemId,
  __ORAFS__ProductType_c: ProductType,
  ProdGroupId: productGroupId,
  __ORAFS__ProductCrossReferenceID_c: ProductIdRef,
  __ORAFS__ProductRevenueReferenceID_c: revRefId,
  RevnAmount:
  [ _text: revAmount ]
]
ChildRevenues.add(revItems)
else {
curRev.setAttribute('__ORAFS__Status_c', '')
break;
}

curRev.setAttribute('__ORAFS__ProductCrossReferenceID_c', ProductIdRef)
curRev.setAttribute('__ORAFS__ProductType_c', ProductType)
//Set the Opportunity attributes for submit Opportunity
def Opportunity =
[
  PrimaryOrganizationId: PrimaryOrganizationId,
  OptyId: OptyId,
  CurrencyCode: CurrencyCode,
  KeyContactId: KeyContactId,
  TargetPartyId: TargetPartyId,
  PartyType: PartyType,
  OptyLastUpdatedBy: userID,
  ChildRevenue: ChildRevenues
]
//Call ICS to submit opportunity in OBP
def list = adf.webServices.SubmitOpportunity.process(Opportunity)

//Set the Submission ID got from the response
if (list != null && list['__ORAFS__SubmissionID_c'] != null && list['__ORAFS__SubmissionID_c'].get(0) != '0') {
  setattr('__ORAFS__SubmissionID_c', list['__ORAFS__SubmissionID_c'].get(0))
  setattr('__ORAFS__SubmissionStatus_c', 'ORA_AFS_SUBMISSION_IN_PROGRESS')
}
//throw the error from OBP.
else {
  childRev.reset()
  while (childRev.hasNext()) {
    // access the next row in the row iterator
def curRev = childRev.next()
    curRev.setAttribute('__ORAFS__Status_c', '')
  }
}
if (list != null && list['DescriptionText'] != null)
{ 
    throw new oracle.jbo.ValidationException(list['DescriptionText'].get(0).toString()) 
} 

8. Publish the sandbox.

Creating Object Workflow to Send Email Notification

For sending email notifications to contacts using Oracle Sales Cloud for Financial Services when the opportunity status is updated on reverse publish from Oracle Banking Platform, you must create an object workflow and an email template.

Creating an Email Template

Following are the steps to create an email template:

1. Sign in to Oracle Sales Cloud as an administrator.
   Ensure that you are not in a sandbox.
2. Navigate to Application Composer.
3. Click EMail Templates in the Common Setup.
4. Click Add icon to create new email template.
   Create EMail Template window opens.
5. Select the object as Opportunity and enter a name for the email template.
6. To add any further details to the subject, optionally, in the EMail Subject section, you can select any option from the drop-down list and click Insert.
   Note that the email subject changes automatically based on the subject selected.
7. To add any further details to the description, in the EMail Body section, you can select any option from the drop-down list and click Insert.
   Note that the email body changes automatically based on the subject selected.
8. Following content is recommended for the email body:
   Add a message to notify that the opportunity status is updated and must update the reason for the status to be Win or Loss. You can specify the Opportunity Name, Status, and Reason in the message. For example, {$Name$} is {$StatusCodes$}. Update Win or Loss reason.
9. Click Save to close the Create EMail Template window.

Creating Object Workflow to Send Email Notification

Following are the steps to create an object workflow to send an email notification:

1. Sign in to Oracle Sales Cloud as an administrator.
2. Navigate to Application Composer.
3. Select Sales check box.
4. Click Object Workflows in Common Setup.
   Object Workflows page opens.
5. Click Create icon.
   Create Object Workflow page opens.
6. Specify **Object** as **Opportunity**.
7. Enter **Name** as **Update the Reason for Won or Lost**.
8. Enter **Description** as **Send email notification When Opportunity Status is Won or Lost and reason is not set**.
9. In the Event Point and Condition section, specify **Event Point** as **When a record is updated**.
10. Click **Groovy Builder** icon.
    Expression Builder dialog box opens.
11. Enter the following condition:
    ```java
    if(StatusCode == 'WON' && (ReasonWonLostCode == null || ReasonWonLostCode == ''))
    {
        return true
    }
    else
    {
        return false
    }
    ```
12. In the Actions section, click **Create** icon corresponding to the **EMail Notification**.
    EMail Notification on Update for Opportunity page opens.
13. For the **Name** field, enter **Enter Reason for Lost or Won Opportunity**.
14. In the EMail Details section, expand the drop-down list and select the email template that you created.
15. Click **To Address** section.
    Create Action : EMail Notification : To Address page opens.
16. Expand **Fields on Record**.
17. Select **Primary EMail** in the **Available Fields on Record** and move it to **Selected Fields on Record**.
18. Click **Save** in all the pages opened.

## Configuring Submission Tracker for Accounts and Contacts

Object functions must be added to configure the submission tracker for accounts and contacts.

To add object functions to configure submission tracker for accounts and contacts:

1. Sign in to Oracle Sales Cloud using the sales administrator role.
2. Ensure that you are in an active sandbox.
3. Navigate to the Application Composer.
4. Expand **Standard Objects** and select the object for which you are creating the object function and click **Server Scripts**.
5. Navigate to the Object Functions tab and click **Actions > Add**.
6. On the Create Object Functions Page, provide the following information for each object function:

<table>
<thead>
<tr>
<th>Object Name</th>
<th>Function Name</th>
<th>Script</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contact</td>
<td>getContactSubmissionTrackerURL</td>
<td></td>
</tr>
</tbody>
</table>

```java
    def ColorContrast = oracle.apps.fnd.applcore.common.ApplSessionUtil.getColorContrast();
    def FontSize = oracle.apps.fnd.applcore.common.ApplSessionUtil.getFontSize();
    def AccessibilityMode = oracle.apps.fnd.applcore.common.ApplSessionUtil.getAccessibilityMode();
    if(__ORAFS__CGUID_c==null||
    __ORAFS__CGUID_c==""){
```
7. Click **Save and Close** and publish the sandbox.

### Configuring Application Form and Application Tracker in Opportunities

Object functions must be configured to configure application form and application tracker for opportunities.

To add object functions to configure application form and application tracker for opportunities:

1. Sign in to Oracle Sales Cloud using the Sales Administrator role.
2. Ensure that you are in an active sandbox.
3. Navigate to the Application Composer.
4. Select Sales check box.
6. Navigate to the Object Functions tab and click Actions > Add.
7. On the Create Object Functions Page, provide the following information for each object function:

<table>
<thead>
<tr>
<th>Object Function</th>
<th>Function Name</th>
<th>Returns</th>
<th>Script</th>
</tr>
</thead>
</table>
| Application Form      | getApplicationFormURL  | String  | def ColorContrast = oracle.apps.fnd.applcore.common.AColorContrast;
def FontSize = oracle.apps.fnd.applcore.common.AFontSize;
def AccessibilityMode = oracle.apps.fnd.applcore.common.AAccessibilityMode;
return oracle.topologyManager.client.deployedInfo.DeployedInfoProvider.getEndPoint('OBP_Application_Form')+'?sid='+__ORAFS__SubmissionID_c+'&returnTarget=CX_OPTY&returnParam=isFuseTF%3DY&tabToOpen=%3D MOO_OPPTYMGMTOPPORTUNITIES_CRM%26cardToOpen=%3D MOO_OPPTYMGMTOPPORTUNITIES_CRM%26TF_subTabName=%3DSummary%26TF_skipToEditOptyId='+OptyId+'&accessibilityParam='+AccessibilityMode+'%23default%23'+ColorContrast+'%23'+FontSize |
| Application Tracker   | getApplicationTrackerURL | String  | def ColorContrast = oracle.apps.fnd.applcore.common.AColorContrast;
def FontSize = oracle.apps.fnd.applcore.common.AFontSize;
def AccessibilityMode = oracle.apps.fnd.applcore.common.AAccessibilityMode;
return oracle.topologyManager.client.deployedInfo.DeployedInfoProvider.getEndPoint('OBP_Application_Tracker')+'?sid='+__ORAFS__SubmissionID_c+'&returnTarget=CX_OPTY&returnParam=isFuseTF%3DY&tabToOpen=%3D MOO_OPPTYMGMTOPPORTUNITIES_CRM%26cardToOpen=%3D MOO_OPPTYMGMTOPPORTUNITIES_CRM%26TF_subTabName=%3DSummary%26TF_skipToEditOptyId='+OptyId+'&accessibilityParam='+AccessibilityMode+'%23default%23'+ColorContrast+'%23'+FontSize |

8. Click Save and Close and publish the sandbox.
Configuring Onboarding Customer

Object functions must be added to configure the onboarding customer for contacts and accounts.

To add object functions to configure onboarding customers for contacts and accounts:

1. Sign in to Oracle Sales Cloud using the Sales Administrator role.
2. Ensure that you are in an active sandbox.
3. Navigate to the Application Composer.
4. Expand **Standard Objects** and select the object for which you are creating the object function and click **Server Scripts**.
5. Navigate to the Object Functions tab and click **Actions > Add**.
6. On the Create Object Functions Page, provide the following information for each object function:

<table>
<thead>
<tr>
<th>Object Name</th>
<th>Function Name</th>
<th>Returns</th>
<th>Script</th>
</tr>
</thead>
</table>
| Company     | getOnboardCustomerURL | String | ```
def ColorContrast = oracle.apps.fnd.appcore.common.A
  def FontSize = oracle.apps.fnd.appcore.common.A
  def AccessibilityMode = oracle.apps.fnd.appcore.common.A
  return oracle.topologyManager.client.deployedInfo.DeployedInfoProvider.getEndPoint('OBP_Onboard_Customer')+
    "?epid=\"&retTarget=\"CX_CONTACT\"&extrefid="+
    "&OSC_PARAMS=isFuseTF%3DY%26tabToOpen%3DY%26cardToOpen%3DHZ_FOUNDATIONPARTIES_CONTACTS_CRM%26TF_subTabName%3DProfile%26TF_ContactPartyId"

Contact     | getOnboardCustomerURL | String | ```
def ColorContrast = oracle.apps.fnd.appcore.common.A
  def FontSize = oracle.apps.fnd.appcore.common.A
  def AccessibilityMode = oracle.apps.fnd.appcore.common.A
  return oracle.topologyManager.client.deployedInfo.DeployedInfoProvider.getEndPoint('OBP_Onboard_Customer')+
    "?epid=\"&retTarget=\"CX_CONTACT\"&extrefid="+
    "&OSC_PARAMS=isFuseTF%3DY%26tabToOpen%3DY%26cardToOpen%3DHZ_FOUNDATIONPARTIES_CONTACTS_CRM%26TF_subTabName%3DProfile%26TF_ContactPartyId"``` |
Object Name | Function Name | Returns | Script
------------|---------------|---------|--------------------------
| %3D"+PartyId +"&accessibilityParam="+AccessibilityMode +"%23default %23"+ColorContrast +"%23"+FontSize

7. Click **Save and Close** and publish the sandbox.

### Configuring Oracle Banking Platform Mashup in Financial Accounts

Configure Oracle Banking Platform in financial accounts by adding an object function.

To add an object function to configure Oracle Banking Platform in financial accounts:

1. Sign in to Oracle Sales Cloud using Sales Administrator role.
2. Ensure that you are in an active sandbox.
3. Navigate to the Application Composer.
4. Expand **Custom Objects > Financial Account** and click **Server Scripts**.
5. Navigate to the Object Functions tab and click **Actions > Add**.
6. On the Create Object Functions page, provide the following information:
   - Function Name: getOBPMashupURL
   - Returns: String
   - In Edit Script, paste the following script and click **Save and Close**:

```java
import oracle.topologyManager.client.deployedInfo.DeployedInfoProvider;

def ColorContrast = oracle.apps.fnd.applcore.common.ApplSessionUtil.getColorContrast();
def FontSize = oracle.apps.fnd.applcore.common.ApplSessionUtil.getFontSize();
def AccessibilityMode = oracle.apps.fnd.applcore.common.ApplSessionUtil.getAccessibilityMode();
return DeployedInfoProvider.getEndPoint('OBP_Servicing') + '?' + __ORAFS__ExternalAssetId_c + '&accessibilityParam=' + AccessibilityMode + '%' + FontSize + 'ColorContrast' + '%23' + FontSize
```

### Configuring Third Party Servicing System

For a product type, servicing system can be Oracle Sales Cloud, Oracle Banking Platform, or a third party. If the servicing system is Oracle Sales Cloud or Oracle Banking Platform, separate pages are set up for financial account transactions. But if the servicing system is a third party, then the user must set the third party URL and create a new layout for the financial account. Following sections contain the instructions for setting third party URL and third party financial account layout.

#### Adding Third Party URL to Manage Third Party Applications

If the user is using any third party application for product servicing for any of the product types, add the third party application URL to [Manage Third Party Applications](#).
To add third party URL to manage third party applications:

1. Sign in to Oracle Sales Cloud using the Application Implementation Consultant role.
2. In the Setup and Maintenance work area, go to the following:
   - Functional Area: Application Extensions
   - Task: Manage Setup Content
3. Select Manage Third Party Applications from the Topology Definition region.
4. Click Create to add third party applications in the following table.

<table>
<thead>
<tr>
<th>Application Name</th>
<th>Full URL</th>
<th>Partner Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>ThirdParty_Servicing</td>
<td>&lt;Third Party Product Servicing URL&gt;</td>
<td>Third Party</td>
</tr>
</tbody>
</table>

5. Click Save and Close.

**Hiding Transaction Subtab**

Hide the transaction subtab before adding the third party transaction subtab. To hide the Transaction subtab:

1. Sign in to Oracle Sales Cloud as an administrator.
2. Ensure that you are in an active sandbox.
3. Navigate to the Application Composer.
5. Click Pages.
6. Perform the following steps in the Details Page Layouts region:
   a. Select All Layouts from the Layouts Status field.
   b. Deselect the active check box for all the layouts other than the third party layout.
   c. Select the third party layout and set its role as Any.
   By selecting any role for the third party layout, you can only view the third party layout when you select a financial account record.
7. Navigate to the Settings and Action menu and select Customize Pages.....
8. On the Customize Pages dialog box, select Site and click OK.
10. Click Select and edit component properties.
11. In the Transaction subtab, click Edit Component.
12. Select Transaction from the Reorder the tabs region and select Hide this tab.
13. Click Apply > OK.
14. Repeat steps 6 to 13 for all the third party layouts listed in Configuring Page Layouts to Use Oracle Banking Platform and Third Party Servicing topic in this chapter.

**Adding Third Party Subtab**

Create a third party subtab to link the third party URLs to each of the third party layouts.

> **Note:** You must hide the Transaction subtab before adding the third party subtab.

To add a third party subtab:

1. Sign in to Oracle Sales Cloud as an administrator.
Chapter 4
Configuring Oracle Sales Cloud

2. Ensure that you are in an active sandbox.
3. Navigate to the Application Composer.
5. Click Pages.
6. Perform the following steps in the Details Page Layouts region:
   a. Select All Layouts from the Layouts Status field.
   b. Click the name of the third party layout you duplicated and perform the following steps in the Create Subtab page.
      i. Click Add.
      ii. Select Web content and click Next.
      iii. Provide a display name for the subtab and add use following example script to provide your own script in the Edit Script field.
         
         
         return
         
         oracle.topologyManager.client.deployedInfo.DeployedInfoProvider.getEndPoint('ThirdParty_Servicing')

         iv. On the Details Layout page, select the layouts on which the subtab must be added and click Save and Close.
7. Click Done.

For more information, see Configuring Page Layouts and Setting Up Product Servicing topics in this chapter.

Populating Value of Onboarding Status Field in Account and Contact

Existing account and contact records must be populated for the Onboarding Status field in Oracle Sales Cloud. Following are the steps:

- Generate import and export artifacts.
- Export data for the following objects:
  - Account
  - Contact
- Import data for the following objects:
  - Account
  - Contact

Generating Import and Export Artifacts

To generate import and export artifacts:

1. Sign in to Oracle Sales Cloud using the sales administrator role.
2. Navigate to the Application Composer.
3. Click Import and Export from the Overview region.
4. Click Generate.
Exporting Data

To export data:

1. Sign in to Oracle Sales Cloud using the sales administrator role.
2. In the Setup and Maintenance work area, go to the following:
   - Functional Area: Application Extensions
   - Task: Manage File Export Activities
3. Search for Manage File Export Activities and select it from the search result.
4. On the Manage Export Activities page, create an export process.
5. Depending on the object for which you are exporting the data, perform the following activities in the Enter Export Options train stop.
   - Account:
     - Parent Object: Account
   - Contact:
     - Parent Object: Contact
6. Navigate to the Configure Export Objects train stop and perform the following steps:
   - In the Export Objects region:
     - Deselect all the View objects for each of the objects:
       - Account: Account Profile
       - Contact: Contact Profile
   - In the Details region:
     - Deselect all the attributes except for:
       - PartyId
   - __ORAFS_OnboardingStatus_c
   - Click Define Criteria for select data to extract and perform the following in the Edit Filter Criteria dialog box:
     - Click Add Fields and select __ORAFS_OnboardingStatus_c.
     - Select ISBLANK for the __ORAFS_OnboardingStatus_c field.
     - Click OK.
7. Navigate to the Create Schedule train stop.
8. Select Immediate as the Schedule Type train stop and navigate to the Review train stop.
9. From Actions, click Activate.
10. Click Refresh and after export is completed, download the exported compressed file.

Note: Extract the CSV file with PartyId and __ORAFS_OnboardingStatus_c columns. Add the code value ORA_AFS_NOT_ONBOARDED for __ORAFS_OnboardingStatus_c column and the corresponding Party Id in the PartyId column in the exported CSV file and save the file.
Importing Data

To import data:

1. Sign in to Oracle Sales Cloud using the sales administrator role.
2. In the Setup and Maintenance work area, go to the following:
   - Offering: Sales
   - Data Import and Export
   - Task: Manage File Import Activities
3. Search for Manage File Import Activities and select it from the search result.
4. On the Manage Import Activities page, create an import process.
5. Depending on the object for which you are exporting the data, perform the following steps on the Enter Import Options train stop:
   a. Select the Object for which you are importing the data.
   b. Select Text file as the File Type.
   c. Select Desktop for the Upload From field and select the file which you imported and revised.
   d. Select Header Row Included.
6. Navigate to the Map Fields train stop and perform the following steps:
   a. In the Map Fields region, select the following in the Object column and the following corresponding attribute in the Attribute column for each of the objects you are importing data for:

<table>
<thead>
<tr>
<th>Object</th>
<th>Corresponding Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>OrganizationProfile</td>
<td>PartyId</td>
</tr>
<tr>
<td>OrganizationProfileExtn</td>
<td>_ _ ORAFS_ OnboardingStatus_ c</td>
</tr>
<tr>
<td>PersonProfile</td>
<td>PartyId</td>
</tr>
<tr>
<td>PersonProfileExtn</td>
<td>_ _ ORAFS_ OnboardingStatus_ c</td>
</tr>
</tbody>
</table>

7. Navigate to the Schedule train stop and select Immediate as the Schedule Type.
8. Navigate to the Review train stop and click Activate.
9. Click Refresh icon and wait for the import to complete.

The Onboarding Status field value is populated for Account and Contact in Oracle Sales Cloud. The value for the Onboarding Status field is Pending for all the existing account and contact records in Oracle Sales Cloud. Expand Standard Objects and select the object for which you are creating the object function and click Server Scripts.
Understanding Financial Account Roll Up for Households

The Financial account roll up functionality enables the ability to roll up all the financial accounts that are linked to a household through at least one direct member of the household.

Retail bankers and retail bank managers are interested in the information on financial account roll ups from the contacts associated with a household. The financial accounts must be rolled up based on individual members of households. Financial accounts are created for individuals and they get rolled up depending on whether the owner of the financial account is a member of the household or not.

There are two types of relationships in households:

- Direct
- Indirect

Roll up happens only for direct type of relationship in households. The roll up would not happen for the indirect relationship. Roll up happens only for Customer or Prospect type of contacts.

Scenarios on How the Financial Accounts Roll Up Works

Following are the scenarios:

- If a new financial account is created for an existing member of the household, it is rolled up if the contact is of the type Customer or Prospect and if the relationship is direct.
- If a new member is added to the household relationship directly and the member has existing financial accounts, these financial accounts are rolled up.
- If an existing member with direct relationship is deleted from household relationship, the financial accounts of the member are no longer shown in the household roll up (if the account is not joint).
- Financial accounts of a member are not rolled up if the member is of type Contact.

Joint Account Scenario

Financial accounts can have joint accounts and joint accounts get rolled up to the household as one single account. Although multiple members of the household can hold that account. A financial account must be shown in the household which has a relationship with the financial account through at least one member who is a customer or prospect and directly related to household.

Financial Account Ownership Change in Joint Accounts

Following are the two scenarios:

- When a joint account is converted to a normal single owner account, the updated single owner account rolls up to the household.
- When a financial account ownership is changed from one contact to another, the updated contact rolls up to the household.
Enabling the Financial Account Roll Up

To enable this functionality, ensure that the Application Composer security for Fin account household is done and the household layout Cx FINS Vertical Layout is enabled for all the required retail banking roles. A Financial account is rolled up to the household automatically if and only if all of the following conditions are fulfilled:

1. Financial account holder is a direct member of the household.
2. If the financial account has multiple holders, at least one holder is a direct member of household.
3. Holder is of type Customer or Prospect.

To add a financial account directly to household, a Fin Account Household record must be created which serves as an intersection or relationship object. While creating the Fin Account Household record, the following two values must be set:

1. ID of the financial account as __ORAFS__FinancialAccount_Id_c.
2. ID of the household as __ORAFS__Household_Id_c.
5 Configuring Siebel UCM

Prerequisites

There are certain prerequisites for configuring Siebel UCM.

Following are the required prerequisites:


2. Install the ACR-UCM-OBP-Integration Siebel objects using the instructions in the readme.txt file in the ACR-UCM-OBP-Integration compressed file. You must obtain the compressed file package from Siebel Business applications team.

   Note: Seed data populated using the data import command may duplicate your existing values. In such a case, see Additional Notes for Upgrading Siebel UCM topic in the Integrating Financial Services with Oracle Banking Platform and Siebel UCM guide.

3. If real-time publish from Siebel UCM is not configured, see Configuring Publish and Subscribe section of the Oracle Customer Hub (UCM) Master Data Management Reference Siebel Innovation Pack 2015 guide to configure real-time publish. Following are the high-level steps:
   
   a. Configure System Preferences
   
   b. Configure System Registration
   
   c. Configure Outbound Webservices

      Configure SyncOrganizationUCMJMSProducer and SyncPersonUCMJMSProducer outbound webservice address to point to AIA SyncOrganizationUCMJMSProducer and SyncPersonUCMJMSProducer composite URLs respectively.

   d. Configure Inbound Webservices


   e. Configure Real-time Publish Workflows:

      i. UCM Async Real-time Publish Workflow
      ii. UCM Organization Customer Profile Integration SOAP process
      iii. UCM Person Customer Profile Integration SOAP Process
      iv. UCM Financial Asset Customer Profile Integration SOAP Process

4. Test the real-time publish using the steps in the How To Test And Verify Siebel UCM Publishing Through A Middleware Integration available at My Oracle Support (Article Document ID 2199116.1).

   Note: AIA middleware system must be set up as a prerequisite before testing real-time publish from Siebel UCM.
Siebel UCM Configuration

For the Oracle Sales Cloud and Siebel UCM integration, you must add a new property to Siebel UCM business services such as Siebel UCM Publish or Siebel UCM Subscribe service.

Following are the steps to add the property:

1. Sign into the Siebel UCM Remote Desktop connection.
2. Open Siebel Tools.
4. Search for UCM Publish or UCM Subscribe business service.

Note: Ctrl+q is the shortcut key for searching.

6. Add a new entry with Name as ViewMode, Value as All.
7. Stop the Siebel Server service from the list of services.
8. Compile the UCM Publish or UCM Subscribe business service by clicking the business service and using the Compile option.
9. Start the Siebel Server service from the list of services.

Configuring System Preferences

Enable system preferences by setting the Enable Child Cross Referencing to True.

Following are the steps:

1. Sign into the Siebel UCM Remote Desktop connection.
2. Open Siebel Tools.
3. Go to Navigator > System Administration > System Preferences.
4. Search for the Enable Child Cross Referencing system preference name.
5. Set the system preference value to True.

Enabling Employees Import and Export as Part of Setting up the Enterprise Organization

To enable import and export of employees from Siebel UCM, following .sif files must be imported in Siebel Tools. These files are included as part of Integrating Financial Services with Oracle Banking Platform and Siebel UCM (Article Document ID 2293151.1) from My Oracle Support.

Import EmployeeImportExport.sif and compile the following objects:

- Employee BC
- Employee List Applet
- Organization List Applet
Import EmployeeImportExport_XREF.sif file and compile the following objects:

- Organization List Applet
- Position List Applet NB
- Import Object Organization
- Import Object Position

Configuring for Know Your Customer Status Synchronization

You must configure the Know Your Customer (KYC) status synchronization. Following are the steps to configure the KYC status synchronization:

1. Sign into the Siebel UCM Remote Desktop connection.
2. Open Siebel Tools.
3. In Object Explorer, lock UCM OBP Integration, VEAI UCM CIF, and Web Service Integration projects to edit. After editing, you must unlock the projects.
4. In Business Component, search for the UCM Account Due Diligence History record and lock it to edit.
5. Add a new field with the following parameters:
   - Name: Updated
   - Column: LAST_UPD
   - Type: DTYPE_UTCDATETIME
   - Text length: 30
6. In the business Component, add the following sort specification: Updated (DESCENDING).
7. Perform the steps 5 and 6 for UCM Contact Due Diligence History business component also.
8. In the Integration Object, search for CIFContact record.
9. In the Integration Component, search for the UCM Contact Due Diligence History record name.
10. In the Integration Component field, search for the external name record Updated and change the Type value to Data.
11. Perform the steps 8, 9, and 10 for the SwiPersonPublishIO integration object.
12. Stop the Siebel Server.
13. Compile the UCM OBP Integration, VEAI UCM CIF, and Web Service Integration projects using the Compile Project option.
14. Compile the UCM Contact Due Diligence History and UCM Account Due Diligence History business components.
15. Compile the CIFContact and SwiPersonPublishIO integration objects.
16. Start the Siebel Server from the list of services.
Configuring Accommodation Type Field

Add LOV for Address Type field to configure Accommodation Type field.

Following are the steps to add LOV for Address Type field:

1. Sign into Siebel UCM as Siebel Administrator.
2. From Navigator, go to Site Map > Administration - Data > List of Values.
3. Query for LOV Type: ADDRESS_TYPE.
4. Add new row, with LOV as ADDRESS_TYPE, Display Value as Other, LIC as CTH, and Translate as Checked.
5. Add another row with LOV as ADDRESS_TYPE, Display Value as Other for IND, LIC as ITH, and Translate as Checked.
6. Set field level properties for business components based on the following table. Compile the business components after the change.

<table>
<thead>
<tr>
<th>Business Component</th>
<th>Field</th>
<th>Field Property and Value to Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>CUT Address</td>
<td>Address Type</td>
<td>Post Default value: Expr: &quot;LookupValue(&quot;ADDRESS_TYPE&quot; **&quot;CTH&quot;)&quot;\n</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Personal Address</td>
<td>Contact Address Type</td>
<td>Post Default value: Expr: &quot;LookupValue(&quot;ADDRESS_TYPE&quot; **&quot;ITH&quot;)&quot;\n</td>
</tr>
</tbody>
</table>

Configuring Email Address Field

Configure the email address field to change the length of the email address field.

Following are the steps to change the length of the email address field:

1. Sign into the Siebel UCM Remote Desktop connection.
2. Open Siebel Tools.
3. In Object Explorer, lock Web Service Integration project.
4. In the Integration Object, search for the SwiPersonIO record and lock it.
5. In the Integration Component, search for Contact.
6. In the Integration Component Field, search for Email Address.
7. For the Field, change the values for columns Length and External Length to 100.
8. Repeat steps 4 to 7 for Integration Objects SwiPersonPublishIO, SwiOrganizationIO, and SwiOrganizationPublishIO also.
10. For the Integration Object, search and select Account Business Address as the integration component, and Email Address as the Integration Component Field.
11. For the Field, change the values for columns Length and External Length to 100.
12. Repeat steps 9 to 11 for the SwiOrganizationPublishIO integration object.
13. To deploy, select and click each modified integration object. Select **Deploy to Runtime Database** option.
15. Compile the **Web Service Integration** project.
16. Compile the **SwiPersonIO**, **SwiPersonPublishIO**, **SwiOrganizationIO**, and **SwiOrganizationPublishIO** integration objects.
17. Start the Siebel Server.
18. In the Siebel application, go to **Navigator > Site Map > Administration - Inbound Webservices** and click **Clear Cache**.

### Configuring for Financial Asset Synchronization

To configure for financial assets synchronization, you must configure the **AIA SyncFinancialAssetUCMJMSProducer** in Siebel UCM as a business service to publish.

Use the following url:

```
http://<aia_server>:<aia_port>/soa-infra/services/default/SyncFinancialAssetUCMJMSProducer/
SyncFinancialAssetUCMJMSProducer?WSDL
```

Following are the steps:

1. Export WSDL files for this web service using SOAP UI.
2. Copy the exported files into a location for which Siebel Tools have read and write permissions.
3. Lock **Web Service Integration** project.
4. From the File menu, select **New Object** to display the **New Object Wizards** dialog box. Click **EAI** tab and then double-click **Web Service**.
   
   a. Select the **Web Service Integration** project.
   b. Specify the WSDL exported.
   c. Use the default values for finalizing the file name and path for the file where you want to store the run-time data extracted from the WSDL document or accept the default values.
   d. Use the default values for finalizing the file name and path for the log file where you want errors, warnings, and other information related to the import process to be reported or accept the default values.
   e. Select the **Process Fault Schema** check box and do not specify the existing Fault Integration Object.
5. Click **Next**.
6. The summary of the import information together with errors, if any, is displayed.
   
   a. Do not select **Deploy the Integration Object** and **Proxy Business Service** check boxes.
   b. Click **Finish** to complete the process of importing the business service into the Siebel repository.
7. After WSDL import, modify the Business Service.
8. In Siebel Tools, from the **Object Explorer**, navigate to **Business Service**.
9. Select **SyncFinancialAssetUCMJMSProducerService** as the **Business Service**.
10. Select **SyncFinancialAsset** as the **Method**.
11. Select **Business Service Method Arg** as **SyncFinancialAssetReqMsg:ListOfSwiFinancialAssetPublishIO**.
   
   a. Update integration object to **SwiFinancialAssetPublishIO**.
12. Compile the business service. To verify the record from Navigator, go to **Administration > Webservices Outbound Web Services**.
13. Configure the business service on the Siebel UCM UI from Navigator, go to **Administration > UCM > System Registrations**.
   
   a. Drill down to AIA records.
b. Navigate to the Publish or Subscribe tab and configure values for the FINCORP account.
   
   i. Business Service: SyncFinancialAssetUCMJMSProducerService

Configuring for Account Synchronization

To configure for account synchronization, you must add Prospect value to CUT_ACCOUNT_TYPE LOV.

Following are the steps:

1. Sign into Siebel UCM as Siebel Administrator.
2. From Navigator, go to Site Map > Administration > Data > List of Values.
3. Query for the CUT_ACCOUNT_TYPE LOV type.
4. Add new row for Prospect.
5. Click Clear Cache.

Configuring LOV Type Alignment

Integrating with the Oracle Banking Platform and Oracle Sales Cloud requires certain pick list or list of value (LOV) changes.

Following is the list of required pick list or list of value changes:

- Configure the Country to State Hierarchy
- Configure the Communication User Type pick list
- Configure the Contact to Financial Asset Relationship pick list
- Configure the Account to Financial Asset Relationship pick list
- Configure Account Type LOV

Configuring the Country to State Hierarchy

You must configure country codes in Siebel UCM. You must set up a country to state hierarchy. You can also change the set of stored country and state codes as required.

锉 Note: Names suggested are recommendations only. You can select any alternate names.

To configure hierarchical pick lists, changes are required in Siebel Repository and Seed Data. Following are the high-level steps:

1. Create the country pick list.
2. Create the state pick list.
3. Configure the CUT Address business component for Country and State fields.
4. Configure the Personal Address business component for Country and State fields.
5. Configure the Account business component for Country fields.
6. Ensure that the S_ADDR_PER table COUNTRY and STATE column_lov_type has a display value that is compatible with the Language Independent Code pick list.

7. Ensure that the S_ORG_EXT table COUNTRY column_lov_type has a display value that is compatible with the Language Independent Code pick list.

8. Adjust seed data to match country and state business needs.

Creating the Country Pick List

Certain steps are required to create Country pick list.

Following are the steps to create Country pick list:

1. Sign into the Siebel UCM Remote Desktop connection.
2. Open Siebel Tools.
3. In Object Explorer, click the Pick List object to display the list of available records.
4. Select and click Record Pane and select New Record.
   An empty record is created in the record pane.
5. Enter the new record details based on the following table:

<table>
<thead>
<tr>
<th>Entity</th>
<th>Value</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Picklist OBP Custom Country</td>
<td>Any name of customer choice.</td>
</tr>
<tr>
<td>Project</td>
<td>OBP Project</td>
<td>Any project of customer choice. But preferably select the same project name which is used for the configured repository.</td>
</tr>
<tr>
<td>Bounded</td>
<td>TRUE</td>
<td>Verify and ensure that the value is as suggested.</td>
</tr>
<tr>
<td>Business Component</td>
<td>Picklist Hierarchical</td>
<td>Verify and ensure that the value is as suggested.</td>
</tr>
<tr>
<td>No Delete, No Insert, No Merge, No Update</td>
<td>TRUE</td>
<td>Verify and ensure that the value is as suggested.</td>
</tr>
<tr>
<td>Type Field</td>
<td>Type</td>
<td>Verify and ensure that the value is as suggested.</td>
</tr>
<tr>
<td>Type Value</td>
<td>COUNTRY_CODE</td>
<td>COUNTRY is the default LOV type that is used for this field. If necessary, you can switch to match your data requirements.</td>
</tr>
</tbody>
</table>

Alternatives include:

- COUNTRY_CODE - This LOV TYPE must be used for short form country codes.
- Or, create your own COUNTRY TYPE field. For example CUSTOM_COUNTRY_TYPE.
6. Save the record.

Similarly, create new pick list with name Picklist OBP Custom Country Short by providing the record details. Before creating the pick list, ensure that the user-defined LOV type CUSTOM_COUNTRY_SHORT is created, see Creating New LOV Types topic. Provide the new record details based on the following table:

<table>
<thead>
<tr>
<th>Entity</th>
<th>Value</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Picklist OBP Custom Country Short</td>
<td>Any name of customer choice.</td>
</tr>
<tr>
<td>Project</td>
<td>OBP Project</td>
<td>Any project of customer choice. But preferably provide the same project name which is used for the configured repository.</td>
</tr>
<tr>
<td>Bounded</td>
<td>TRUE</td>
<td>Verify and ensure that the value is as suggested.</td>
</tr>
<tr>
<td>Business Component</td>
<td>PickList Hierarchical</td>
<td>Verify and ensure that the value is as suggested.</td>
</tr>
<tr>
<td>No Delete, No Insert, No Merge, No Update</td>
<td>TRUE</td>
<td>Verify and ensure that the value is as suggested.</td>
</tr>
<tr>
<td>Type Value</td>
<td>CUSTOM_COUNTRY_SHORT</td>
<td>CUSTOM_COUNTRY_SHORT is the created LOV Type that is user-defined. For additional instructions on creating new LOV TYPES and adding additional LOV values, see Creating New LOV Types topic.</td>
</tr>
</tbody>
</table>

Creating the State Pick List

Certain steps are required to create State pick list.

Following are the steps to create State pick list:

1. Sign into the Siebel UCM Remote Desktop connection.
2. Open Siebel Tools.
3. In Object Explorer, click the Pick List object to display the list of available records.
4. Select and click Record Pane and select New Record.

An empty record is created in the record pane.
5. Enter the new record details based on the following table:
Configuring Siebel UCM

### Table

<table>
<thead>
<tr>
<th>Entity</th>
<th>Value</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Picklist OBP Custom State</td>
<td>Any name of customer choice.</td>
</tr>
<tr>
<td>Project</td>
<td>OBP Project</td>
<td>Any project of customer choice. But preferably provide the same project name which is used for the configured repository.</td>
</tr>
<tr>
<td>Bounded</td>
<td>TRUE</td>
<td>Verify and ensure that the value is as suggested.</td>
</tr>
<tr>
<td>Business Component</td>
<td>PickList Hierarchical Sub-Area</td>
<td>Verify and ensure that the value is as suggested.</td>
</tr>
<tr>
<td>No Delete, No Insert, No Merge, No Update</td>
<td>TRUE</td>
<td>Verify and ensure that the value is as suggested.</td>
</tr>
<tr>
<td>Type Field</td>
<td>Type</td>
<td>Verify and ensure that the value is as suggested.</td>
</tr>
<tr>
<td>Type Value</td>
<td>STATE_ABBREV</td>
<td>STATE_ABBREV is the default LOV type that is used for this field. If necessary, you can switch to match your data requirements. Alternatives include creating your own STATE TYPE field. For example, CUSTOM_STATE_TYPE. For additional instructions on creating new LOV TYPEs and adding additional LOV Values, see Creating New LOV Types topic.</td>
</tr>
</tbody>
</table>

6. Save the record.

7. Compile the OBP Project used for creating the pick list.

### Configuring the CUT Address Business Component and Country and State Fields

Certain steps are required to configure the CUT address business component and Country and State fields.

Following are the steps required to configure:

1. Sign into the Siebel UCM Remote Desktop connection.
2. Open Siebel Tools.
3. In Object Explorer, click the Business Component object to display the list of available records.
4. Query for the CUT Address business component.
5. Expand the Business Component in the Object Explorer, click Field entity.
7. Select the Country field and perform the following changes:
Chapter 5
Configuring Siebel UCM

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Value</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pick list</td>
<td>Picklist OBP Custom Country</td>
<td>This is the new pick list created.</td>
</tr>
<tr>
<td>Immediate Post Changes</td>
<td>TRUE</td>
<td>Set this property to True.</td>
</tr>
</tbody>
</table>

8. Expand the Field entity and select Pick Map. Create new records based on the following table.

<table>
<thead>
<tr>
<th>Name</th>
<th>Constraint</th>
<th>Pick List Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>Country</td>
<td>NULL</td>
<td>Value</td>
</tr>
<tr>
<td>State</td>
<td>NULL</td>
<td>NULL</td>
</tr>
</tbody>
</table>

9. Query for the field that represents State field.

10. Select State field and perform the following changes:

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Value</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pick list</td>
<td>Picklist OBP Custom State</td>
<td>This is the new pick list created.</td>
</tr>
<tr>
<td>Immediate Post Changes</td>
<td>TRUE</td>
<td>Set this property to True.</td>
</tr>
</tbody>
</table>

11. Expand the Field entity and select Pick Map. Create new records based on the following table:

<table>
<thead>
<tr>
<th>Name</th>
<th>Constraint</th>
<th>Pick List Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>Country</td>
<td>TRUE</td>
<td>Parent</td>
</tr>
<tr>
<td>State</td>
<td>NULL</td>
<td>Value</td>
</tr>
</tbody>
</table>

12. Compile the VERT CUT Address project for creating the pick list.

Configuring the Personal Address Business Component and and Country and State Fields

Certain steps are required to configure the Personal Address business component and Country and State fields.

Following are the steps to configure:

1. Sign into the Siebel UCM Remote Desktop connection.
2. Open Siebel Tools.
3. In Object Explorer, click the Business Component object to display the list of available records.
4. Query for the Personal Address business component.
5. Expand the Business Component in the Object Explorer, click Field entity.
6. Query for the **Country** field.
7. Select the **Country** field and perform the following changes:

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Value</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pick list</td>
<td>Picklist OBP Custom Country</td>
<td>This is the new pick list created.</td>
</tr>
<tr>
<td>Immediate Post Changes</td>
<td>TRUE</td>
<td>Set this property to True.</td>
</tr>
</tbody>
</table>

8. Expand the Field entity and select Pick Map. Create new records based on the following table:

<table>
<thead>
<tr>
<th>Name</th>
<th>Constraint</th>
<th>Pick List Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>Country</td>
<td>NULL</td>
<td>Value</td>
</tr>
<tr>
<td>State</td>
<td>NULL</td>
<td>NULL</td>
</tr>
</tbody>
</table>

9. Query for the **State** field.
10. Select **State** field and perform the following changes:

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Value</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pick list</td>
<td>Picklist OBP Custom State</td>
<td>This is the new pick list created.</td>
</tr>
<tr>
<td>Immediate Post Changes</td>
<td>TRUE</td>
<td>Set this property to True.</td>
</tr>
</tbody>
</table>

11. Expand the **Field** entity and select **Pick Map**. Create new records based on the following table:

<table>
<thead>
<tr>
<th>Name</th>
<th>Constraint</th>
<th>Pick List Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>Country</td>
<td>TRUE</td>
<td>Parent</td>
</tr>
<tr>
<td>State</td>
<td>NULL</td>
<td>Value</td>
</tr>
</tbody>
</table>

12. Compile the **Contact** project.

### Configuring the Account Business Component and Country Field

Certain steps are required for configuring the Account business component and Country field.

Following are the steps required for configuring:

1. Sign into the Siebel UCM Remote Desktop connection.
2. Open **Siebel Tools**.
3. In **Object Explorer**, click the **Business Component** object to display the list of available records.
4. Query for the **Account** business component.
5. Expand the **Business Component** in the Object Explorer, click **Field** entity.
6. Query for the **UCM Country of Incorporation** field.
7. Select the **UCM Country of Incorporation** field and perform the following changes:

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Value</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pick list</td>
<td>Picklist OBP Custom Country Short</td>
<td>This is the new pick list created.</td>
</tr>
<tr>
<td>Immediate Post Changes</td>
<td>TRUE</td>
<td>Set this property to True.</td>
</tr>
</tbody>
</table>

8. Query for the **Country** field.
9. Select the **Country** field and perform the following changes:

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Value</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pick list</td>
<td>Picklist OBP Custom Country</td>
<td>This is the new pick list created.</td>
</tr>
<tr>
<td>Immediate Post Changes</td>
<td>TRUE</td>
<td>Set this property to True.</td>
</tr>
</tbody>
</table>

10. Compile the **Account** project.

## Configuring the Contact Business Component and Country Field

Certain steps are required to configure the Contact business component and Country field.

Following are the required steps to configure:

1. Sign into the Siebel UCM Remote Desktop connection.
2. Open **Siebel Tools**.
3. In **Object Explorer**, click the **Business Component** object to display the list of available records.
4. Query for the **Contact** business component.
5. Expand the **Business Component** in the Object Explorer, click **Field** entity.
6. Query for the **Country** field.
7. Select the **Country** field and perform the following changes:

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Value</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pick list</td>
<td>Picklist OBP Custom Country</td>
<td>This is the new pick list created.</td>
</tr>
<tr>
<td>Immediate Post Changes</td>
<td>TRUE</td>
<td>Set this property to True.</td>
</tr>
</tbody>
</table>

8. Query for the **INS Personal Country** field.
9. Select the **INS Personal Country** field and perform the following changes:
Configuring the UCM HE Constituent Identification Business Component and Country Field

Certain steps are required to configure the UCM HE Constituent Identification business component and Country field. Following are the steps required to configure:

1. Sign into the Siebel UCM Remote Desktop connection.
2. Open Siebel Tools.
3. In Object Explorer, click Business Component object to display the list of available records.
4. Query for the UCM HE Constituent Identification business component.
5. Expand the Business Component in the Object Explorer, click Field entity.
7. Select the Country field and perform the following changes:

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Value</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pick list</td>
<td>Picklist OBP Custom Country</td>
<td>This is the new pick list created.</td>
</tr>
</tbody>
</table>

Immediate Post Changes | TRUE | Set this property to True. |

8. Compile the UCM Higher Education project.

Ensuring the Compatibility of S_ADDR_PER Table and COUNTRY and STATE Columns of Lov Type

Certain steps are required to ensure the compatibility of S_ADDR_PER table and COUNTRY and STATE columns of Lov type. Following are the steps required to ensure compatibility:

1. Sign into the Siebel UCM Remote Desktop connection.
2. Open Siebel Tools.
3. In Object Explorer, click the Table object to display the list of available records.
4. Query for the S_ADDR_PER table.
5. Expand the Table in the Object Explorer, click Column entity.
6. Query for the COUNTRY column.
7. Select the **COUNTRY** column and ensure that the **Lov Type** is using **LOV_TYPE** that has a display value matching with the Country pick list’s language independent code. The recommended value is **COUNTRY** since it has the same language dependent value and language independent code. **COUNTRY** is compatible with pick lists that use either **COUNTRY** or **COUNTRY_CODE**.

8. Save the record.

9. Query for the **STATE** column.

10. Select the **STATE** column and ensure that the **Lov Type** is using **LOV_TYPE** that has a display value matching with the State pick list’s language independent code.

11. Save the record.

12. Compile the **Table Person** project.

Ensuring the Compatibility of S_ORG_EXT Table and COUNTRY Column of Lov Type

Certain steps are required to ensure the compatibility of S_ORG_EXT table and COUNTRY column of Lov type.

Following are the steps required to ensure compatibility:

1. Sign into the Siebel UCM Remote Desktop connection.

2. Open **Siebel Tools**.

3. In **Object Explorer**, click **Table** object to display the list of available records.

4. Query for the **S_ORG_EXT** table.

5. Expand the **Table** in the object explorer, click **Column** entity.

6. Query for the **INCORP_COUNTRY** column.

7. Select the **INCORP_COUNTRY** column and ensure that the **Lov Type** is using **LOV_TYPE** that has a display value matching with the Country Pick List’s Language Independent Code. The recommended value is the user-defined LOV type created that is, **CUSTOM_COUNTRY_SHORT**.

8. Save the record.

9. Compile the **Table Organization** project.

Ensuring the Compatibility of S_CON_IDNTY_DOC Table and COUNTRY Column of Lov Type

Certain steps are required to ensure the compatibility of S_CON_IDNTY_DOC table and COUNTRY column of Lov type.

Following are the steps required to ensure compatibility:

1. Sign into the Siebel UCM Remote Desktop connection.

2. Open **Siebel Tools**.

3. In **Object Explorer**, click the **Table** object to display the list of available records.

4. Query for the **S_CON_IDNTY_DOC** table.

5. Expand the **Table** in the object explorer, click **Column** entity.

6. Query for the **ISS_COUNTRY_CD** column.

7. Select the **ISS_COUNTRY_CD** column and ensure that the **Lov Type** is using **LOV_TYPE** that has a display value matching with the Country Pick List’s Language Independent Code. The recommended value is the user-defined LOV type created that is, **CUSTOM_COUNTRY_SHORT**.
Adjusting Seed data to Match Country and State Business Needs

The steps for adjusting seed data to match country and state business needs are provided with the assumption that you have LOV types named CUSTOM_COUNTRY_TYPE, CUSTOM_STATE_TYPE, and CUSTOM_COUNTRY_SHORT in your repository. If they are available, modify the values in these LOV types.

In case these LOV types are not already available and you want to create these new LOV types, use the following steps:

Creating New LOV Types

Following are the steps to create new list of values:

1. Sign into the Siebel UCM Remote Desktop connection.
2. From navigator, go to Site Map > Administration > Data > List of Values.
3. In the displayed list applet, create new LOVs for COUNTRY based on the following table:

<table>
<thead>
<tr>
<th>Type</th>
<th>Display Value</th>
<th>Language Independent Code</th>
<th>Language Name</th>
<th>Parent LIC</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>CUSTOM_COUNTRY_TYPE</td>
<td>USA</td>
<td>USA</td>
<td>English-American</td>
<td>NULL</td>
<td>1</td>
</tr>
<tr>
<td>CUSTOM_COUNTRY_TYPE</td>
<td>India</td>
<td>India</td>
<td>English-American</td>
<td>NULL</td>
<td>2</td>
</tr>
</tbody>
</table>

Create new LOVs for CUSTOM_COUNTRY_SHORT based on the following table:

<table>
<thead>
<tr>
<th>Type</th>
<th>Display Value</th>
<th>Language Independent Code</th>
<th>Language Name</th>
<th>Parent LIC</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>CUSTOM_COUNTRY_SHORT</td>
<td>AU</td>
<td>AU</td>
<td>English-American</td>
<td>NULL</td>
<td>1</td>
</tr>
</tbody>
</table>

Create new LOVs for STATES based on the following table:

<table>
<thead>
<tr>
<th>Type</th>
<th>Display Value</th>
<th>Language Independent Code</th>
<th>Language Name</th>
<th>Parent LIC</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>CUSTOM_STATE_TYPE</td>
<td>Alabama</td>
<td>Alabama</td>
<td>English-American</td>
<td>USA</td>
<td>1</td>
</tr>
<tr>
<td>CUSTOM_STATE_TYPE</td>
<td>Alaska</td>
<td>Alaska</td>
<td>English-American</td>
<td>USA</td>
<td>2</td>
</tr>
<tr>
<td>Type</td>
<td>Display Value</td>
<td>Language Independent Code</td>
<td>Language Name</td>
<td>Parent LIC</td>
<td>Order</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------------</td>
<td>---------------------------</td>
<td>-------------------</td>
<td>------------</td>
<td>-------</td>
</tr>
<tr>
<td>CUSTOM_STATE_TYPE</td>
<td>Hawaii</td>
<td>Hawaii</td>
<td>English-American</td>
<td>USA</td>
<td>11</td>
</tr>
<tr>
<td>CUSTOM_STATE_TYPE</td>
<td>Washington</td>
<td>Washington</td>
<td>English-American</td>
<td>USA</td>
<td>49</td>
</tr>
<tr>
<td>CUSTOM_STATE_TYPE</td>
<td>Jammu &amp; Kashmir</td>
<td>Jammu &amp; Kashmir</td>
<td>English-American</td>
<td>India</td>
<td>1</td>
</tr>
<tr>
<td>CUSTOM_STATE_TYPE</td>
<td>Punjab</td>
<td>Punjab</td>
<td>English-American</td>
<td>India</td>
<td>2</td>
</tr>
<tr>
<td>CUSTOM_STATE_TYPE</td>
<td>Andhra Pradesh</td>
<td>Andhra Pradesh</td>
<td>English-American</td>
<td>India</td>
<td>3</td>
</tr>
<tr>
<td>CUSTOM_STATE_TYPE</td>
<td>Karnataka</td>
<td>Karnataka</td>
<td>English-American</td>
<td>India</td>
<td>12</td>
</tr>
</tbody>
</table>

**Note:** The list in the table is not final and comprehensive. You can add records based on your requirement. Parent Language Independent Code (LIC) value must be correctly entered.

To configure states for Australia, enter Display Value and Language-Independent Code based on the following table. LOV type used must be same as the one used for creating the state pick list.

<table>
<thead>
<tr>
<th>Display Value</th>
<th>Language Independent Code</th>
<th>Language Name</th>
<th>Parent LIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>NT</td>
<td>NT</td>
<td>English-American</td>
<td>Australia</td>
</tr>
<tr>
<td>ACT</td>
<td>ACT</td>
<td>English-American</td>
<td>Australia</td>
</tr>
<tr>
<td>NSW</td>
<td>NSW</td>
<td>English-American</td>
<td>Australia</td>
</tr>
<tr>
<td>QLD</td>
<td>QLD</td>
<td>English-American</td>
<td>Australia</td>
</tr>
<tr>
<td>SA</td>
<td>SA</td>
<td>English-American</td>
<td>Australia</td>
</tr>
<tr>
<td>TAS</td>
<td>TAS</td>
<td>English-American</td>
<td>Australia</td>
</tr>
<tr>
<td>VIC</td>
<td>VIC</td>
<td>English-American</td>
<td>Australia</td>
</tr>
<tr>
<td>WA</td>
<td>WA</td>
<td>English-American</td>
<td>Australia</td>
</tr>
</tbody>
</table>

After the LOV setup is completed, perform the steps in the Mapping Address LOV Field topic for LOV mapping in AIA.
Address LOV Field Mapping

Address LOV field mapping comprises of mapping for state and mapping for country.

Mapping for State

Identify the code for state that is used by the Oracle Banking Platform in Siebel UCM. If the configuration based on the Creating the State Pick List topic in the Integrating Financial Services with Oracle Banking Platform and Siebel UCM guide is completed, then use the Type Value used while creating the pick list as the LOV type. Following are the steps:

- Sign into Siebel UCM as Siebel Administrator.
- From Navigator, go to Site Map > Administration > Data > List of Values.
- Query for the LOV type value used in the Creating the State Pick List topic in the Integrating Financial Services with Oracle Banking Platform and Siebel UCM guide and for Display Value, select the state that you must map.
- Note the LIC for this LOV entry.

Map the state code in Application Integration Architecture using the following steps:

- Edit the dvm file to update the mapping for state type STATE.dvm
- Ensure that the UCM_01 value for the state that must be mapped, matches the value in Siebel UCM.
- Note the value for COMMON for the state.
- Deploy the file in MDS using the steps in the Updating MDS section of the Oracle Fusion Middleware Migration Guide for Oracle Application Integration Architecture.

⚠️ Note: If any of the values for UCM_01 and COMMON are already existing in the dvm, then edit the existing values for mappings.

Identify the code for country in Oracle Sales Cloud. Following are the steps:

- Sign into Oracle Sales Cloud.
- Navigate to Contacts.
- Click Create Contact.
- From the Create Contact page, note the value for the field state, if the field state is a drop-down list. If the field state is not a drop-down list, then you can provide values for the state that you want to map to a specific country. For the country United Kingdom, you can get the value from the County field since state field is not available.

Map state in Oracle Integration Cloud Service using the following steps:

- Sign into Oracle Integration Cloud Service application.
- Go to Menu > Designer > Lookups.
- Edit STATE_LOOKUP.
- Verify and ensure that the value of AIA in lookup for State is same as the mapping for COMMON in AIA. Edit if required and save.
- Verify and ensure that the value of Oracle Sales Cloud for State is the same value as in the Identify the code for country in Oracle Sales Cloud section. Edit if required and save.
Mapping for Country
Identify the code for state that is used by the Oracle Banking Platform in Siebel UCM. If the configuration based on the Creating the Country Pick List topic in the Integrating Financial Services with Oracle Banking Platform and Siebel UCM guide is completed, then use the Type Value used while creating the pick list as the LOV type. Following are the steps:

- Sign into Siebel UCM as Siebel Administrator.
- From Navigator, go to Site Map > Administration > Data > List of Values.
- Query for the LOV type value used in the Creating the Country Pick List topic in the Integrating Financial Services with Oracle Banking Platform and Siebel UCM guide and for Language Independent Code, select the country that you must map. Also confirm that the country exists in LOV type.

Map the country code in Application Integration Architecture using the following steps:

- Edit the dvm file to update the mapping for country type ADDRESS_COUNTRYID.dvm
- Ensure that the UCM_01 value for the country that must be mapped, matches the Language Independent Code value in Siebel UCM.
- Note the value for COMMON for the country.
- Deploy the file in MDS using the steps in the Updating MDS section of the Oracle Fusion Middleware Migration Guide for Oracle Application Integration Architecture.

Note: If any of the values for UCM_01 and COMMON are already existing in the dvm, then edit the existing values for mappings.

Identify the code for country in Oracle Sales Cloud. Following are the steps:

- Sign into Oracle Sales Cloud.
- In the Setup and Maintenance work area, go to the following:
  - Functional Area: Application Extensions
  - Task: Manage Geographies
  - Click Go to Task.
    The Manage Geographies page opens.
- Enter the country name such as Australia.
- Click Search.
- From the result, note the value for Country Code.
- For more details, see Setting Up Geography Reference Data for Territories and Addresses chapter of the Getting Started with Your Oracle Sales Cloud Implementation guide.

Map country in Oracle Integration Cloud Service using the following steps:

- Sign into Oracle Integration Cloud Service application.
• Go to **Menu > Designer > Lookups**.
• Edit **COUNTRY_LOOKUP**.
• Verify and ensure that the value of AIA in lookup for Country is same as the mapping for COMMON in AIA. Edit if required and save.
• Verify and ensure that the value of Oracle Sales Cloud for Country is the same value as in the Identify the code for country in Oracle Sales Cloud section. Edit if required and save.

**Note:** If any of the value for Oracle Sales Cloud and Application Integration Architecture is already present in the lookup, you must delete the existing values or edit the existing values. Also, if value for the Country is not configured in lookup, then address synchronization will not take place.

### Configuring the Communication Use Type Pick List

You must use an open communication use type picklist to adhere to the recommended Banking Platform policies.

Following are the required steps:

1. Sign into the Siebel UCM Remote Desktop connection.
2. Open **Siebel Tools**.
3. In **Object Explorer**, click the **Business Component** object to display the list of available records.
4. Query for the **Communication Address** business component.
5. Expand the business component in the Object Explorer and click **Field** entity.
6. Query for the **Use Type** field.
7. Select the **Use Type** field and perform the following changes. Also set **Immediate Post Changes** to **True**.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Value</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pick list</td>
<td>LOY PickList Comm All Type</td>
<td>Verify and ensure that the value is as suggested.</td>
</tr>
</tbody>
</table>

8. In the **Contact Business** component, search for the **Communication Address Use Type** field and change the pick list to **LOY PickList Comm All Type**. Set **Immediate Post Changes** to **True**.
9. Save the record.
10. Compile **Communication Address** and **Contact** business components.

### Configuring the Contact with Financial Asset Relationship Pick List

You must configure an alternative contact with the financial asset relationship pick list to adhere to the recommended banking platform policies.

Following are the steps:

1. Sign into the Siebel UCM Remote Desktop connection.
2. Open **Siebel Tools**.
3. In **Object Explorer**, click the **Pick List** object to display the list of available records.
4. Select and click **Record Pane** and select **New Record**.
   An empty record is created in the record pane.

5. Enter the new record details based on the following table:

<table>
<thead>
<tr>
<th>Entity</th>
<th>Value</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Picklist OBP Custom Contact Asset</td>
<td>Any name of customer choice.</td>
</tr>
<tr>
<td>Project</td>
<td>OBP Project</td>
<td>Any project of customer choice. But preferably provide the same project name which is used for the configured repository.</td>
</tr>
<tr>
<td>Bounded</td>
<td>TRUE</td>
<td>Verify and ensure that the value is as suggested.</td>
</tr>
<tr>
<td>Business Component</td>
<td>PickList Generic</td>
<td>Verify and ensure that the value is as suggested.</td>
</tr>
<tr>
<td>No Delete, No Insert, No Merge, No Update</td>
<td>TRUE</td>
<td>Verify and ensure that the value is as suggested.</td>
</tr>
<tr>
<td>Type Field</td>
<td>Type</td>
<td>Verify and ensure that the value is as suggested.</td>
</tr>
<tr>
<td>Type Value</td>
<td>AUTO_ASSET_CON_REL</td>
<td>Verify and ensure that the value is as suggested.</td>
</tr>
</tbody>
</table>

6. Save the record.

7. In **Object Explorer**, click the **Business Component** object to display the list of available records.

8. Query for the **FINCORP Account Contact** business component.

9. Expand the business component in the Object Explorer and click **Field entity**.

10. Query for the **Type** field.

11. Select the **Type** field and perform the following changes:

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Value</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pick list</td>
<td>Picklist OBP Custom Contact Asset</td>
<td>This is the new pick list created.</td>
</tr>
<tr>
<td>Immediate Post Changes</td>
<td>TRUE</td>
<td>Set this property to True.</td>
</tr>
</tbody>
</table>

12. Save the record.

13. Expand the **Field** entity and select **Pick Map**. If a record is not available, then create a new record based on the following table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Constraint</th>
<th>Pick List Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>False</td>
<td>Value</td>
</tr>
</tbody>
</table>

14. Save the record.
Configuring the Account with Financial Asset Relationship Pick List

It is recommended to configure an alternative account with the financial asset relationship pick list to adhere to the recommended banking platform policies. This pick list is useful only when you build a many to many association between assets and accounts.

The following optional steps are required to configure an alternative account with the financial asset relationship pick list:

1. Sign into the Siebel UCM Remote Desktop connection.
2. Open Siebel Tools.
3. In Object Explorer, click the Pick List object to display the list of available records.
4. Select and click Record Pane and select New Record.

An empty record is created in the record pane.
5. Enter the new record details based on the following table:

<table>
<thead>
<tr>
<th>Entity</th>
<th>Value</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Picklist OBP Custom Account Asset</td>
<td>Any name of customer choice.</td>
</tr>
<tr>
<td>Project</td>
<td>OBP Project</td>
<td>Any project of customer choice. But preferably provide the same project name which is used for the configured repository.</td>
</tr>
<tr>
<td>Bounded</td>
<td>TRUE</td>
<td>Verify and ensure that the value is as suggested.</td>
</tr>
<tr>
<td>Business Component</td>
<td>PickList Generic</td>
<td>Verify and ensure that the value is as suggested.</td>
</tr>
<tr>
<td>No Delete, No Insert, No Merge, No Update</td>
<td>TRUE</td>
<td>Verify and ensure that the value is as suggested.</td>
</tr>
<tr>
<td>Type Field</td>
<td>Type</td>
<td>Verify and ensure that the value is as suggested.</td>
</tr>
<tr>
<td>Type Value</td>
<td>AUTO_ ASSET_ ACCNT_REL</td>
<td>Verify and ensure that the value is as suggested.</td>
</tr>
</tbody>
</table>

6. Save the record.

If a many to many relationship is built, it will require construction of a business component based on the S_ASSET_ACCNT table. You can use the FINS Organization Asset as a starting point for the configuration. Use the pick list while configuring a new field in S_ASSET_ACCNT.REL_TYPE_CD.

Note: This configuration will also require integration object changes for SwiFinancialAssetIO, SwiFinancialAssetPublishIO, and related CIFs.
Configuring Account Type LOV

Certain steps are required to add *Prospect* value for configuring the Account Type LOV.

Following are the steps required to add *Prospect* value to CUT_ACCOUNT_TYPE LOV:

1. Sign into Siebel UCM as Siebel Administrator.
2. From Navigator, go to Site Map > Administration > Data > List of Values.
3. Query for the LOV Type: CUT_ACCOUNT_TYPE.
4. Add new row for *Prospect*.
5. Click Clear Cache.

Additional Notes for Upgrading Siebel UCM

The following section contains useful information for upgrading your Siebel UCM instance. These optional instructions are required only for upgrading your Siebel UCM instance.

Additional Notes for Applying Siebel Seed Data

Siebel UCM LOV seed data can overlap with the data already existing in your Siebel application instance. If this happens, following message (indicated in the table rows) is displayed:

<table>
<thead>
<tr>
<th>Date</th>
<th>Time</th>
<th>Total Rows with Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>2017-01-01</td>
<td>12:00:00</td>
<td>TOTAL ROWS PROCESSED: 79</td>
</tr>
<tr>
<td>2017-01-01</td>
<td>12:00:00</td>
<td>TOTAL ROWS FAILED: 13</td>
</tr>
<tr>
<td>2017-01-01</td>
<td>12:00:00</td>
<td>TOTAL ROWS SUCCEEDED: 66</td>
</tr>
</tbody>
</table>

In such a case, further inspection into the message log, will result in the display of messages such as:

```
2017-01-01 12:00:00 SQL Warning, SQL State 23000, 1, [tp][ODBC Oracle driver][Oracle]ORA-00001: unique constraint (ORATEST.S_LST_OF_VAL_U1) violated
```

To reconcile differences between imported seed data and your instance, either:

- It is recommended to delete the matching row in the Siebel database and rerun the import. This will replace the old value with the new value.
- Or, compare the value in the message log for the matching TYPE and NAME.

For reference, a typical seed data import command specifying an output log is as follows:

```
dataimp /u sadmin /p ***** /f C:\UCM-OBP-Integration\Seed\Seed_Locale_ENU.dat /l C:\lov_seed_2017_01_01.log /c "SSD default instance" /d oradb /h log /z y /w y
```
6 Setting Up the Product Master

Product Master: Overview

Being a one-stop solution for core banking operations, Oracle Banking Platform also serves as a master application for banking products. A user with an administrator role can import products to Oracle Sales Cloud from Oracle Banking Platform and can use these imported products data for creating opportunities.

Financial products in Oracle Banking Platform and Oracle Sales Cloud have one to one mapping. Product Catalog in Oracle Banking Platform has multiple levels and you must map them to corresponding levels in Oracle Sales Cloud.

For creating the product catalog in Oracle Sales Cloud, import file with columns from Oracle Banking Platform hierarchy. To import file, map fields in Oracle Banking Platform product levels to the product catalog in Oracle Sales Cloud. Import the product hierarchy in Oracle Banking Platform to Oracle Sales Cloud.

Mapping of Objects and Product Type between Oracle Sales Cloud and Oracle Banking Platform

The following table lists the required mappings of objects between Oracle Banking Platform and Oracle Sales Cloud for creating the product catalog in Oracle Sales Cloud.

<table>
<thead>
<tr>
<th>Oracle Banking Platform</th>
<th>Oracle Sales Cloud</th>
<th>Additional Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product Class</td>
<td>Product Type</td>
<td>Product class values are stored as list of values in Oracle Sales Cloud. The values are fixed list of values that identify whether the product is deposit, lending, or credit card, and so on. For example, CS (Current and Savings), TD (Term Deposit), and LN (Lending). To store these values in Oracle Sales Cloud, use the Fixed Choice List field available in both Product and Product Group.</td>
</tr>
<tr>
<td>(Line of Business)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Product Group</td>
<td>Product Group</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Offer</td>
<td>Product</td>
<td>Not applicable</td>
</tr>
</tbody>
</table>
Mapping Product Type in Oracle Sales Cloud to Product Class in Oracle Banking Platform

The Product Class in Oracle Banking Platform is a fixed choice list corresponding to the Product Type field in Oracle Sales Cloud. The Product Type is a fixed choice list and is added as a field in both Product and Product Group.

Following table lists Product Type values corresponding to the Product Class.

<table>
<thead>
<tr>
<th>Product Class Code in Oracle Banking Platform</th>
<th>Product Type in Oracle Sales Cloud</th>
<th>Additional Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS</td>
<td>ORA_AFS_CS</td>
<td>Current and Savings</td>
</tr>
<tr>
<td>TD</td>
<td>ORA_AFS_TD</td>
<td>Term Deposits</td>
</tr>
<tr>
<td>LN</td>
<td>ORA_AFS_LN</td>
<td>Lending</td>
</tr>
<tr>
<td>CC</td>
<td>ORA_AFS_CC</td>
<td>Credit Card</td>
</tr>
<tr>
<td>IA</td>
<td>ORA_AFS_IA</td>
<td>Investment</td>
</tr>
<tr>
<td>IN</td>
<td>ORA_AFS_IN</td>
<td>Insurance</td>
</tr>
</tbody>
</table>

Setting Up Product Catalog

Following are the steps for product catalog setup:

1. Exporting Product Catalog from Oracle Banking Platform.
3. Importing Product Groups in Oracle Sales Cloud.
4. Importing Products to Oracle Sales Cloud.
5. Importing Cross References for Products and Product Groups.

Exporting Product Catalog from Oracle Banking Platform

Oracle Banking Platform product data hand-off is designed as follows:

1. During the day, user creates or modifies offers and validates them.
2. During Oracle Banking Platform daily batch, a job runs to validate offers modified on that day and move those offers to staging tables of the Oracle Banking Platform.
3. As part of the Oracle Banking Platform batch job, an ODI job (Oracle Data Integration job) picks up all the offers that are not yet published to Oracle Sales Cloud and extracts a CSV file, and moves the file to a configured location.
4. Oracle Sales Cloud can update the catalog using the CSV file available at the configured location.
5. Batch job is an automated job configured by the Oracle Banking Platform user along with other Oracle Banking Platform batch jobs, there is no requirement for any configuration in Oracle Sales Cloud, except to uptake the changes and update catalog. For more details, see Oracle Banking Platform guide.
Following are the Oracle Banking Platform batch job details entered by the Oracle Banking Platform user:

- Category Name: Sales Offer Handoff
- Shell Name: Analytics Hand-Off Shell
- Execution Frequency: Daily

**Note:** Category name, shell name, and execution frequency are actual job details. The job is run on a daily schedule. The job is available as a part of the Oracle Banking Platform application to the user. User must modify the default location and set the shared location in Oracle Banking Platform from where data is exported to Oracle Sales Cloud.

The product catalog is exported from Oracle Banking Platform as a CSV file in the sample format shown in the following table.

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Example Value</th>
<th>Additional Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>OFFER_CODE</td>
<td>UOF100, LHBTB1, or USTDINVOF102</td>
<td>Not applicable</td>
</tr>
<tr>
<td>OFFER_NAME</td>
<td>Usaver, Back to Basics, or SEP IRA TD OFFER</td>
<td>Not applicable</td>
</tr>
<tr>
<td>OFFER_DESCRIPTION</td>
<td>Usave Offer, Back to Basics, or SEP IRA TD OFFER</td>
<td>Not applicable</td>
</tr>
<tr>
<td>PRODUCT_CLASS</td>
<td>CS, LN, or TD</td>
<td>Not applicable</td>
</tr>
<tr>
<td>PRODUCT_GROUP_CODE</td>
<td>CSGP1, LNHLN, or USINVTD</td>
<td>Not applicable</td>
</tr>
<tr>
<td>OFFER_START_DATE</td>
<td>31-Dec-15, 31-Dec-15, or 31-Dec-15</td>
<td>Not applicable</td>
</tr>
<tr>
<td>OFFER_EXPIRY_DATE</td>
<td>30-Dec-99, 30-Dec-99, or 30-Dec-99</td>
<td>Not applicable</td>
</tr>
<tr>
<td>ALLOWED_RATE_TYPE</td>
<td>V, F, or F</td>
<td>Not applicable</td>
</tr>
<tr>
<td>ALLOWED_CURRENCY_CODES</td>
<td>INR#AUD, AUD, or USD</td>
<td>Not applicable</td>
</tr>
</tbody>
</table>

Following table consists of fields exported from Oracle Banking Platform for the import of product catalogs to Oracle Sales Cloud.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Additional Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>OFFER_CODE</td>
<td>Used for cross reference (Offer ID) in Oracle Sales Cloud.</td>
</tr>
<tr>
<td>OFFER_NAME</td>
<td>Mapped to Item Description in Oracle Sales Cloud.</td>
</tr>
<tr>
<td>OFFER_DESCRIPTION</td>
<td>Mapped to Item Long Description in Oracle Sales Cloud.</td>
</tr>
<tr>
<td>PRODUCT_GROUP_CODE</td>
<td>The Product Group ID to which the Offer is linked.</td>
</tr>
</tbody>
</table>
The exported data is available at a shared location based on a schedule. Shared location is configured in Oracle Banking Platform database. Clients can configure using Oracle Banking Platform UI as part of day zero configuration.

Creating a Matching Product Group Structure in Oracle Sales Cloud

You must identify and create the product group for the product in Oracle Sales Cloud, that maps to the product group of the offer in Oracle Banking Platform.

Creating Root Product Group

To use a Product Group created in an opportunity, the Product Group must be part of a Root Product Group. Therefore, while creating a Product Group, you must specify its Root Product Group. If there is already one, you can use it. Otherwise, create a new Product Group as Root and use the same.

Following are the steps to create a root product group:

1. Sign in to Oracle Sales Cloud as an administrator.
2. In the Setup and Maintenance work area, go to the following:
   - Functional Area: Application Extensions
   - Task: Manage Product Groups
3. In the Manage Product Groups pane, search for Root product group.
4. If there is an existing Root product group, use it.
5. If there is no existing Root product group, then in the Manage Product Groups pane, click Create icon.

   Create Product Group dialog box opens.

6. Provide Name and Display details.
7. Select Root check box.
8. Click Save and Close.

Importing Product Groups in Oracle Sales Cloud

Importing product groups in Oracle Sales Cloud comprises of preparing files for the product group import and importing product groups in Oracle Sales Cloud.
Preparing Files for the Product Group Import

Prepare CSV file for the Product Group Import using the format specified in the following table.

The **Download Template** button in the Import Mapping used for Product Group provides the sample CSV file. Map only the fields indicated in the following table. Ensure that other field values are blank.

<table>
<thead>
<tr>
<th>Column Name in .csv File to be Imported</th>
<th>Field value in the file exported or from Oracle Banking Platform or Default Values</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>PGHeader Reference Number</td>
<td>&lt;Product Group Name from Oracle Banking Platform&gt;</td>
<td>NA</td>
</tr>
<tr>
<td>PGHeader Internal Name</td>
<td>&lt;Product Group Name from Oracle Banking Platform&gt;</td>
<td>NA</td>
</tr>
<tr>
<td>PGHeader Language</td>
<td>US</td>
<td>NA</td>
</tr>
<tr>
<td>PGHeader Source Language Flag</td>
<td>Y</td>
<td>NA</td>
</tr>
<tr>
<td>PGHeader Product Group Display Name</td>
<td>&lt;Product Group Name from Oracle Banking Platform&gt;</td>
<td>NA</td>
</tr>
<tr>
<td>PGHeader Allow Duplicate Children Flag</td>
<td>N</td>
<td>NA</td>
</tr>
<tr>
<td>PGHeader Root Catalog Flag</td>
<td>N</td>
<td>NA</td>
</tr>
<tr>
<td>PGHeader Source Object Type</td>
<td>Sales Catalog</td>
<td>NA</td>
</tr>
<tr>
<td>PGHeader Product Type</td>
<td>Product Type value in Oracle Sales Cloud corresponding to the PRODUCT_CLASS</td>
<td>NA</td>
</tr>
<tr>
<td>PGHeader Product Group ReferenceID</td>
<td>PRODUCT_.GROUP_.CODE</td>
<td>NA</td>
</tr>
<tr>
<td>PGRel Parent PG Reference Number</td>
<td>&lt;Reference Number of your root Product Group&gt;</td>
<td>NA</td>
</tr>
<tr>
<td>PGRel Child PG Reference Number</td>
<td>&lt;Product Group Name from Oracle Banking Platform&gt;</td>
<td>NA</td>
</tr>
<tr>
<td>PGRel Relation Type Code</td>
<td>PRODUCT_.GROUP_.HIERARCHY</td>
<td>NA</td>
</tr>
<tr>
<td>PGRel Language</td>
<td>US</td>
<td>NA</td>
</tr>
<tr>
<td>PGRel Source Language Flag</td>
<td>Y</td>
<td>NA</td>
</tr>
<tr>
<td>PGRel Active Flag</td>
<td>Y</td>
<td>NA</td>
</tr>
</tbody>
</table>
Use the productgroup_import_sample.csv spreadsheet available at Collaterals for Integrating Financial Services with Oracle Banking Platform and Siebel UCM (My Oracle Support Article Doc ID 2293151.1). Enter the root product group reference number that is, PGRel Parent PG Reference Number in your import file to link the product groups you are importing to the root. Use the Manage Product Groups task to find the reference number.

Importing Product Groups

Following are the steps to import product groups:

1. Sign in to Oracle Sales Cloud using Administrator role.
2. In the Setup and Maintenance work area, go to the following:
   - Functional Area: Application Extensions
   - Task: Manage File Import Mappings
3. In Search, select Product Group as Object and select Seeded check box. Click Search.
5. Click Copy Mapping.
6. Change the Import Mapping Name to Product Group Mapping for OBP Import.
7. Click the Add icon in column mapping.

Add the following three rows and specify the sequence number in the Sequence column:

<table>
<thead>
<tr>
<th>Column Name in .csv File to be Imported</th>
<th>Field value in the file exported or from Oracle Banking Platform or Default Values</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>PGHeader Product Group Allowable Currencies</td>
<td>ALLOWED__CURRENCY__CODES</td>
<td>Note: When we export the Product Catalog from Oracle Banking Platform, the ALLOWED__CURRENCY__CODES comes with # as a separator. When we import the Product and Product Groups from Manage File Import Activities page in Oracle Sales Cloud, we provide the Allowable Currencies values with separator as ; instead of #.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Column Header</th>
<th>Product Type Example Value</th>
<th>Object</th>
<th>Attribute</th>
<th>Display Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>PGHeader Product Type</td>
<td>ORAFS_CS</td>
<td>ProductGroupBulkImport or Product Group</td>
<td>__ ORAFS__ _ ProductType_c</td>
<td>Product Type</td>
</tr>
<tr>
<td>PGHeader Product Group ReferenceID</td>
<td>CSGP1</td>
<td>ProductGroupBulkImport or Product Group</td>
<td>__ ORAFS__ _ ProductGroupRefId_c</td>
<td>Product Group reference Id</td>
</tr>
<tr>
<td>PGHeader Product Group AllowableCurrencies</td>
<td>CSGP1</td>
<td>ProductGroupBulkImport or Product Group</td>
<td>__ ORAFS__ _ AllowableCurrencies_c</td>
<td>Allowable Currencies</td>
</tr>
</tbody>
</table>

Note: Sequence Number: When we add mappings, they are added in incremental order. While Importing, the values specified are taken in the same order as specified in the mapping.
8. Save the Mapping file.
9. Use the Manage File Import Activities task to import your file. Following are the steps:
   a. In the Setup and Maintenance work area, go to the following:
      - Offering: Sales
      - Functional Area: Data Import and Export
      - Task: Manage File Import Activities
   b. Search for Manage File Import Activities task.
   c. In the Search Results, click Manage File Import Activities link.
      Manage Import Activities page opens.
   d. Click Create icon.
      Create Import Activity page opens.
   e. Enter the information for Name and Description fields. Select Product Group from the Object list. Select Desktop option for Upload from in the Source File section.
   f. Click Browse and select the CSV file prepared in the Step of the Preparing Files for the Product Group Import section.
      Select the Product Group Mapping for OBP Import Import Mapping file. Click Next.
   g. Click Next.
   h. Click Next.
      Edit Import: Review and Activate page opens.
   i. Click Activate. Wait for the process to complete and verify that the status of the import job is changed to Completed.

Related Topics
- Collaterals for Integrating Financial Services with Oracle Banking Platform and Siebel UCM

Importing Products to Oracle Sales Cloud

The Offers exported from the Oracle Banking Platform are imported to Oracle Sales Cloud as Sales Products.

Following table contains the fields required for importing a product in Oracle Sales Cloud. For more details on how to export orders, see Oracle Banking Platform guide.

<table>
<thead>
<tr>
<th>Required Fields</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item Number</td>
<td>Unique value that identifies the product being created.</td>
</tr>
<tr>
<td>Name</td>
<td>Name of the product.</td>
</tr>
<tr>
<td>Item Status Code</td>
<td>Status of the product.</td>
</tr>
<tr>
<td>Primary UOM Code</td>
<td>UOM of the product.</td>
</tr>
</tbody>
</table>
Preparing Files for Import

Prepare CSV file for product import using the format specified in the following table. Use the product_import_sample.csv spreadsheet available at Collaterals for Integrating Financial Services with Oracle Banking Platform and Siebel UCM (My Oracle Support Article Doc ID 2293151.1).

<table>
<thead>
<tr>
<th>CSV Column Header</th>
<th>Field Value in the file exported or from Oracle Banking Platform or Default Values</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number</td>
<td>OFFER_CODE</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Name</td>
<td>OFFER_NAME</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Description</td>
<td>OFFER_DESCRIPTION</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Offer Id</td>
<td>OFFER_CODE</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Product Type</td>
<td>Not applicable</td>
<td>See table in the Mapping Product Type in Oracle Sales Cloud to Product Class in Oracle Banking Platform section.</td>
</tr>
<tr>
<td>Primary UOM Code</td>
<td>&lt;Default UOM Code in Sales Product&gt;</td>
<td>Select a default UOM code needed</td>
</tr>
<tr>
<td>Language</td>
<td>US</td>
<td>US</td>
</tr>
<tr>
<td>Source Language</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>Item Status Code</td>
<td>Active</td>
<td>Active</td>
</tr>
</tbody>
</table>
## Allowable Currencies

<table>
<thead>
<tr>
<th>CSV Column Header</th>
<th>Field Value in the file exported or from Oracle Banking Platform or Default Values</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allowable Currencies</td>
<td>ALLOWED.Currency.CODES</td>
<td></td>
</tr>
</tbody>
</table>

**Note:** When we export the Product Catalog from Oracle Banking Platform, the ALLOWED.Currency.CODES come with # as a separator. When we import the Product and Product Groups from Manage File Import Activities page in Oracle Sales Cloud, we provide the Allowable Currencies values with separator as ; instead of #.

---

### Importing Products

To import products:

1. In the Setup and Maintenance work area, go to the following:
   - Functional Area: Application Extensions
   - Task: Manage File Import Mappings
2. Search for **Manage File Import Mappings** task.
3. In the Search Results, click **Manage File Import Mappings** link. Manage File Import Mappings page opens.
4. Click **Create** icon. Create Import Mapping page opens.
5. Specify **Import Mapping** value as **Product Mapping for OBP**. **Object** as **Product**. **Description** as **Import mapping file for product**.
6. Click **Save and Close**.
7. Search for the mapping **Product Mapping for OBP**. Drill down to the mapping file.
8. Click **Add** icon in the Column Mappings section.
9. Add the Column headers and their mappings as specified in the following table.

<table>
<thead>
<tr>
<th>Column Header</th>
<th>Example Value</th>
<th>Object</th>
<th>Attribute</th>
<th>Display Name</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>Offer Id</td>
<td>CBBEA1</td>
<td>ProductBulkImport or Product</td>
<td>OfferId.c</td>
<td>Offer Id</td>
<td>N</td>
</tr>
<tr>
<td>Name</td>
<td>Usave Offer</td>
<td>ProductBulkImport or Product</td>
<td>ItemDescription</td>
<td>Name</td>
<td>N</td>
</tr>
<tr>
<td>Primary UOM Code</td>
<td>zzw</td>
<td>ProductBulkImport or Product</td>
<td>PrimaryUomCode</td>
<td>Default UOM Code</td>
<td>N</td>
</tr>
<tr>
<td>Language</td>
<td>US</td>
<td>ProductBulkImport or Product</td>
<td>Language</td>
<td>Language</td>
<td>N</td>
</tr>
<tr>
<td>Source Language Flag</td>
<td>Y</td>
<td>ProductBulkImport or Product</td>
<td>SourceLangFlag</td>
<td>Source Language</td>
<td>N</td>
</tr>
<tr>
<td>Column Header</td>
<td>Example Value</td>
<td>Object</td>
<td>Attribute</td>
<td>Display Name</td>
<td>Required</td>
</tr>
<tr>
<td>------------------------</td>
<td>------------------------</td>
<td>-------------------------------</td>
<td>-----------------------------</td>
<td>--------------------</td>
<td>----------</td>
</tr>
<tr>
<td>Description</td>
<td>Usave Offer description</td>
<td>Product or Product</td>
<td>ItemLongDescription</td>
<td>Description</td>
<td>N</td>
</tr>
<tr>
<td>Item Status Code</td>
<td>Active</td>
<td>Product or Product</td>
<td>ItemStatusCode</td>
<td>Status</td>
<td>N</td>
</tr>
<tr>
<td>Product Type</td>
<td>ORAFS_CS</td>
<td>Product or Product</td>
<td>_ _ ORAFS_ _ProductType_c</td>
<td>Product Type</td>
<td>N</td>
</tr>
<tr>
<td>Number</td>
<td>Usave Offer</td>
<td>Product or Product</td>
<td>ItemNumber</td>
<td>Product Number</td>
<td>N</td>
</tr>
<tr>
<td>Allowable Currencies</td>
<td>INR;USD;AUD</td>
<td>Product or Product</td>
<td>_ _ ORAFS_ _AllowableCurrencies_c</td>
<td>Allowable Currencies</td>
<td>N</td>
</tr>
</tbody>
</table>

10. Save the Mapping file.
11. In the Setup and Maintenance work area, go to the following:
   - Offering: Sales
   - Functional Area: Data Import and Export
   - Task: Manage File Import Activities
12. Search for Manage File Import Activities task. Navigate to that task.
13. In the Search Results, click Manage File Import Activities link.

Manage File Import Activities page opens.
14. Click Create icon.

Create Import Activity page opens.
15. Enter the information for Name and Description fields. Select Product from the Object list. Select Desktop option for Upload from in the Source File section.
16. Click Browse and select the CSV file. Check the Header Row included check box. Select the new mapping file Product Mapping for OBP.
17. Click Next.

Create Import Activity: Review and Activate page opens.
18. Click Activate.

Wait for the process to complete and confirm that the status of the import job is changed to Completed.
19. After completing the import, navigate to Products page and verify that the product is created.

*Note:* In case of changes to existing products also, these instructions are valid since the Offer Code of OBP maps to a unique Item Number. For all existing products if the item number is same as of the one that is being imported, then the record is merged.
Importing Products in Appropriate Product Groups

After the product groups and products are imported, we must import the product in the appropriate Product Group using the following steps:

1. Proceed with the same steps as used for importing product groups in Importing Product Groups in Oracle Sales Cloud topic.
2. Use the seeded mapping file saved during Manage File Import Activities task in step 11 of the Importing Product Groups in Oracle Sales Cloud section of the Importing Product Groups in Oracle Sales Cloud topic.
3. Update and use the sample CSV file producttoproductgroup_sample.csv with values for importing the product in the product group.

Map only the fields indicated in the following table. Ensure that other field values are blank.

<table>
<thead>
<tr>
<th>CSV column header</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>PGHeader Reference Number</td>
<td>Parent Product Group Reference Number</td>
</tr>
<tr>
<td>PGHeader Internal Name</td>
<td>Parent Product Group Internal Name</td>
</tr>
<tr>
<td>PGHeader Product Group Display Name</td>
<td>Parent Product Group Display Name</td>
</tr>
<tr>
<td>PGHeader Source Object Type</td>
<td>Sales Catalog</td>
</tr>
<tr>
<td>PGItem Product Number</td>
<td>OFFER_CODE</td>
</tr>
<tr>
<td>PGItem Active Flag</td>
<td>Y</td>
</tr>
</tbody>
</table>

If multiple products are in the same product group, then enter a row with all the details available in the table in this section and fill only the **PGItem Product Number** value in the following row. See the sample CSV file producttoproductgroup_sample.csv.

Related Topics

- Integrating Oracle Sales Cloud with Siebel CRM
- Collaterals for Integrating Financial Services with Oracle Banking Platform and Siebel UCM

Importing Cross References for Products and Product Groups

Store the ID of the Product and Product Group in Oracle Sales Cloud cross reference object __ORAFS__XREF_c corresponding to the Offer and Product Group in Oracle Banking Platform. Use the sample crossref_mapping.csv spreadsheet available at Collaterals for Integrating Financial Services with Oracle Banking and Siebel UCM (My Oracle Support Article Doc ID 2293151.1).
Following are the steps:

1. In the Setup and Maintenance work area, go to the following:
   a. Offering: Sales
   b. Functional Area: Data Import and Export
   c. Task: Manage File Activities

2. Search for Manage File Import Activities task.

3. In the Search Results, click Manage File Import Activities link.

4. Click Create icon.

5. Enter the information for Name and Description fields. Select Common Custom Object from the Object list. Select Desktop option for Upload from in the Source File section.

6. Click Browse and select the CSV file. Use the crossref_mapping.csv spreadsheet available at Collaterals for Integrating Financial Services with Oracle Banking Platform and Siebel UCM (My Oracle Support Article Doc ID 2293151.1).

   Select the Header Row included check box.

   Click Next. CSV file is attached.

7. Map the fields as specified in the table.

8. Click Next.

9. Click Activate.

   Wait for the process to complete and confirm that the status of the import job is changed to Completed.
Related Topics

- Collaterals for Integrating Financial Services with Oracle Banking Platform and Siebel UCM
Configuring Oracle Integration Cloud Service: Overview

Oracle Sales Cloud and Siebel UCM bidirectional integration uses Oracle Integration Cloud Service (ICS) as one of the integration component. Oracle Integration Cloud Service uses a series of pre-built integrations that are packaged and deployed on your ICS instance. Once these integrations are activated, the data between Oracle Sales Cloud, Oracle SOA suite, and Siebel UCM starts synchronizing.

The following tasks must be performed to set up the secure integration between applications:

- Create the Integration Cloud Service Integration User. See Setting up Integration Users and Roles.
- Import certificates from Oracle Sales Cloud and Siebel UCM to ICS.
- Create Integration Cloud Service credential key in Oracle Sales Cloud. See Creating a CSF Key for Event Subscriptions.
- Import the integration package and configure integration endpoints.
- Configure Integration Connections.
- Clone and activate integrations.
- Set up Error Notifications.

Related Topics
- Setting Up Integration Users and Roles
- Creating a CSF Key for Event Subscriptions

Importing Certificates to Integration Cloud Service: Overview

You must import certificates from Oracle Sales Cloud and Siebel UCM to Integration Cloud Service to validate outbound SSL connections to Oracle Sales Cloud and Siebel UCM and enable connections to the external services.

Before you begin, export certificates from Oracle Sales Cloud and Siebel UCM.

**Note:** The instructions for exporting certificates, using https URL in the web browser are dependent on the client web browser and the version. Use the relevant documentation provided by your browser service provider to export the certificates.

Import both the certificates to ICS using instructions provided in the Uploading an SSL Certificate in ICS topic listed in Related Topics section. Use the following information while importing:

- **Certificate Type:** Trust Certificate
• Certificate Alias: Oracle Sales Cloud and Siebel UCM

Note: You must ensure that Oracle Sales Cloud instance is aware of Integration Cloud Service instance. This can be setup when you receive the instances. If you see any security issues, file a service request.

Related Topics
• Uploading an SSL Certificate in ICS

Importing the Integration Package: Highlights

You can import a package of integrations into Oracle Integration Cloud Service from the Packages page. The Packages page enables you to import packages that you or other users have created.

Downloading the Package
Download the Oracle Integration Cloud Service (ICS) integration artifacts to your local computer from `<cxfins.zip>/ics/packages`.

- Use cxfins.par package file available at Integrating Financial Services with Oracle Banking Platform and Siebel UCM (My Oracle Support Article Doc ID 2293151.1) on My Oracle Support at https://support.oracle.com

Importing the Package
- Import the integration package to ICS using instructions provided by the Integration Cloud Service documentation. See: Importing a Package in ICS.

Verifying the Integration Artifacts
You can verify these by navigating to Integrations, Lookups, and Connections pages from the Designer menu.

- After the import is successful, verify the integration artifacts listed in Integration Artifacts Reference topic are imported within the package. See Integration Artifacts Reference for Integration Cloud Service in the Related Topics section.

Configuring Integration Service Endpoints: Explained

If your integration user name is other than FUSION_APPS_ICS_APPID, you must perform the following additional configuration for the purposes of echo suppression for Oracle Sales Cloud to Siebel UCM integration flows.

1. Sign in to Oracle Integration Cloud service.
2. Navigate to Designer > Integrations.
3. For the integrations in listed in the following table, From the Actions menu, click Edit.

<table>
<thead>
<tr>
<th>Integration</th>
<th>Trigger name</th>
<th>Filter Expression</th>
<th>Expression</th>
</tr>
</thead>
</table>
| I_OSC_AIA_Account_Create_Cxfins | CreateAccountOSC | Filter Expr for Account created Event | `<xpathExpr xmlns:ns_1='http://`
<table>
<thead>
<tr>
<th>Integration</th>
<th>Trigger name</th>
<th>Filter Expression</th>
</tr>
</thead>
<tbody>
<tr>
<td>I_OSC_AIA_Account_</td>
<td>updateAccountOSC</td>
<td>xmlns:ns_5='<a href="http://xmlns.oracle.com/apps/crmCommon/salesParties/accountService/">http://xmlns.oracle.com/apps/crmCommon/salesParties/accountService/</a>'</td>
</tr>
<tr>
<td>Update_Cxfins</td>
<td></td>
<td>xmlns:ns_0='<a href="http://xmlns.oracle.com/adf/svc/types/">http://xmlns.oracle.com/adf/svc/types/</a>'</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$eventPayload/ns_1:result/ns_0:Value/ns_5:LastUpdatedBy != 'Integration-user-name'</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&lt;xpathExpr xmlns:ns_1='<a href="http://xmlns.oracle.com/apps/crmCommon/salesParties/">http://xmlns.oracle.com/apps/crmCommon/salesParties/</a>'</td>
</tr>
<tr>
<td></td>
<td></td>
<td>accountService/types/' xmlns:ns_0='<a href="http://xmlns.oracle.com/adf/svc/types/">http://xmlns.oracle.com/adf/svc/types/</a>'</td>
</tr>
<tr>
<td></td>
<td></td>
<td>xmlns:ns_5='<a href="http://xmlns.oracle.com/apps/crmCommon/salesParties/accountService/">http://xmlns.oracle.com/apps/crmCommon/salesParties/accountService/</a>'</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$eventPayload/ns_1:result/ns_0:Value/ns_5:LastUpdatedBy != 'Integration-user-name'</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Integration</th>
<th>Trigger name</th>
<th>Filter Expression</th>
</tr>
</thead>
<tbody>
<tr>
<td>I_OSC_AIA_Contact_</td>
<td>createContactOSC</td>
<td>xmlns:ns_5='<a href="http://xmlns.oracle.com/apps/crmCommon/salesParties/contactservice/">http://xmlns.oracle.com/apps/crmCommon/salesParties/contactservice/</a>'</td>
</tr>
<tr>
<td>Create_Cxfins</td>
<td></td>
<td>xmlns:ns_0='<a href="http://xmlns.oracle.com/adf/svc/types/">http://xmlns.oracle.com/adf/svc/types/</a>'</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$eventPayload/ns_1:result/ns_0:Value/ns_5:LastUpdatedBy != 'Integration-user-name'</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&lt;xpathExpr xmlns:ns_1='<a href="http://xmlns.oracle.com/apps/crmCommon/salesParties/">http://xmlns.oracle.com/apps/crmCommon/salesParties/</a>'</td>
</tr>
<tr>
<td></td>
<td></td>
<td>contactservice/types/' xmlns:ns_0='<a href="http://xmlns.oracle.com/adf/svc/types/">http://xmlns.oracle.com/adf/svc/types/</a>'</td>
</tr>
<tr>
<td></td>
<td></td>
<td>xmlns:ns_5='<a href="http://xmlns.oracle.com/apps/crmCommon/salesParties/contactservice/">http://xmlns.oracle.com/apps/crmCommon/salesParties/contactservice/</a>'</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$eventPayload/ns_1:result/ns_0:Value/ns_5:LastUpdatedBy != 'Integration-user-name'</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Integration</th>
<th>Trigger name</th>
<th>Filter Expression</th>
</tr>
</thead>
<tbody>
<tr>
<td>I_OSC_AIA_Contact_</td>
<td>updateContactOSC</td>
<td>xmlns:ns_5='<a href="http://xmlns.oracle.com/apps/crmCommon/salesParties/contactservice/">http://xmlns.oracle.com/apps/crmCommon/salesParties/contactservice/</a>'</td>
</tr>
<tr>
<td>Update_Cxfins</td>
<td></td>
<td>xmlns:ns_0='<a href="http://xmlns.oracle.com/adf/svc/types/">http://xmlns.oracle.com/adf/svc/types/</a>'</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$eventPayload/ns_1:result/ns_0:Value/ns_5:LastUpdatedBy != 'Integration-user-name'</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&lt;xpathExpr xmlns:ns_1='<a href="http://xmlns.oracle.com/apps/crmCommon/salesParties/">http://xmlns.oracle.com/apps/crmCommon/salesParties/</a>'</td>
</tr>
<tr>
<td></td>
<td></td>
<td>contactservice/types/' xmlns:ns_0='<a href="http://xmlns.oracle.com/adf/svc/types/">http://xmlns.oracle.com/adf/svc/types/</a>'</td>
</tr>
<tr>
<td></td>
<td></td>
<td>xmlns:ns_5='<a href="http://xmlns.oracle.com/apps/crmCommon/salesParties/contactservice/">http://xmlns.oracle.com/apps/crmCommon/salesParties/contactservice/</a>'</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$eventPayload/ns_1:result/ns_0:Value/ns_5:LastUpdatedBy != 'Integration-user-name'</td>
</tr>
</tbody>
</table>

---

**Note:** The table above contains filter expressions used for integrating Oracle Integration Cloud Service with Oracle Banking Platform and Siebel UCM. Each row represents a specific trigger and its corresponding filter expression to handle different events related to account and contact updates.
4. Open the trigger names listed in the table in edit mode.
5. Click **Request**.
6. In the Configure the Integration Service Endpoint to Receive Requests from Oracle Sales Cloud Application region, update the **Expression** using the table and for the **Filter Expression** listed in the same table.
7. Provide the expression using the preceding table, at **Filter Expression** after replacing the value of Integration-username in the **Expression** column of the table.
8. Click **Save** to save the trigger.
9. Click **Save** on the integration page to same the integration.

### Configuring Integration Connections: Highlights

You must configure connectivity and security for the connections imported from the integration package. Use the connections and related configuration information provided in the following table to configure integration connections.

**Note:** You must refer to relevant sections in ICS documentation to configure the integration connections listed in the following table. Refer to the Related Documentation section of this topic to find the references to all the required documentation.

<table>
<thead>
<tr>
<th>Connection Name</th>
<th>Connection Adapter</th>
<th>Connection Properties</th>
<th>Connection Security Policies</th>
<th>Agent Groups</th>
</tr>
</thead>
<tbody>
<tr>
<td>CUSTOMERPARTYEBM</td>
<td>SOAP</td>
<td>• Use the &lt;cxfins.zip&gt;/ics/connections/ SyncCustomerPart file available at Integrating Financial Services with Oracle Banking Platform and Siebel UCM (2293151.1) on My Oracle Support at <a href="https://support.oracle.com">https://support.oracle.com</a>. • Open the .wsdl file and replace the values of AIA host and Port with actual values for your Oracle AIA instance and save. • Upload the modified file as WSDL URL. • Suppress insertion of</td>
<td>NA</td>
<td>NA</td>
</tr>
</tbody>
</table>

---

Integration | Trigger name | Filter Expression | Expression |
------------|--------------|-------------------|------------|
<p>|             |              | ns_5:LastUpdatedBy != 'Integration-user-name' xPathExpr |            |</p>
<table>
<thead>
<tr>
<th>Connection Name</th>
<th>Connection Adapter</th>
<th>Connection Properties</th>
<th>Connection Security Policies</th>
<th>Agent Groups</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Provide user name and password for AIA weblogic administration.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Provide user name and password for AIA weblogic administration.</td>
<td></td>
</tr>
<tr>
<td>C_AIA_Cxpt</td>
<td>SOAP</td>
<td>• Use the &lt;cxfins.zip&gt;/ics/connections/SyncFinancialAccount file available at Integrating Financial Services with Oracle Banking Platform and Siebel UCM (2293151.1) on My Oracle Support at <a href="https://support.oracle.com">https://support.oracle.com</a>.</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Open the .wsdl file and replace the values of AIA host and Port with actual values for your Oracle AIA instance and save.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Upload the modified file as WSDL URL.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Suppress insertion of timestamp into the request: Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CustomerPartyResponseEBM</td>
<td>SOAP</td>
<td>Use the &lt;cxfins.zip&gt;/ics/connections/SyncCustomerPart file available at Integrating Financial Services with Oracle</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Connection Name</td>
<td>Connection Adapter</td>
<td>Connection Properties</td>
<td>Security Policy</td>
<td>Agent Groups</td>
</tr>
<tr>
<td>-----------------</td>
<td>--------------------</td>
<td>-----------------------</td>
<td>----------------</td>
<td>--------------</td>
</tr>
<tr>
<td>C_OSC</td>
<td>Oracle Sales Cloud</td>
<td>You must obtain a mandatory Oracle Sales Cloud Adapter service catalog service WSDL (for accessing business objects) and optionally an event catalog URL (for accessing event subscriptions) or interface catalog URL (for accessing Oracle Fusion Applications REST API resources). Specify the required WSDL URLs in the Connection Properties dialog.</td>
<td>Security Policy: Username and Password Token</td>
<td>NA</td>
</tr>
<tr>
<td>C_OBP_SUBMISSION</td>
<td>SOAP</td>
<td>WSDL URL: http://&lt;OBP HOST&gt;/com.ofss.fc.webservice/services/origination/SubmissionBasicDraftCreation</td>
<td>Security Policy: Username and Password Token</td>
<td>You must download the agent installer from Oracle Integration Cloud Service and run the installer to install the on-premises agent in your local environment. During installation, you associate the agent with the agent group identifier you generated when creating an agent group in Oracle Integration.</td>
</tr>
</tbody>
</table>
Chapter 7
Configuring Oracle Integration Cloud Service

Cloud Service. To configure agent groups:

1. Create an agent group in ICS.
2. Download and run On-Premise Agent Installer.
3. Configure this agent group by editing the connection.

**Note:** For detailed instructions on configuring agent group, see Related Documentation Section.

<table>
<thead>
<tr>
<th>Connection Name</th>
<th>Connection Adapter</th>
<th>Connection Properties</th>
<th>Connection Security Policies</th>
<th>Agent Groups</th>
</tr>
</thead>
</table>

Provide user name and password for Oracle Banking Platform.

**Related Documentation**

For steps to edit connections to configure connectivity and security, refer to the following documentation provided by Integration Cloud Service:

- **General ICS related documentation:**
  - See: Editing a Connection in the Integration Cloud Service.
  - See: Creating an Agent Group in Integration Cloud Service.
  - See: Downloading and Running the On-Premises Agent Installer.

- **Oracle Sales Cloud Adapter related documentation:**
  - See: Obtaining the Oracle Sales Cloud Adapter Service Catalog Service WSDL, Event Catalog URL, or Interface Catalog URL.
  - See: Configuring Connection Properties for Oracle Sales Cloud Adapter.
  - See: Configuring Connection Security for Oracle Sales Cloud.

- **SOAP Adapter related documentation:**
Cloning and Activating Integrations

You must not edit or delete the prebuilt integrations that are included in this integration package. Instead clone the integration, with new name and identifier and activate the cloned integration. Cloning an integration creates a new copy with identical connections and data mappings. You can reconfigure the clone after you create it. Once you create or clone an integration and the progress indicator shows 100 percent, you can activate that integration to the runtime environment.

Note: Use the Integrations table in the Integration Artifacts Reference for Oracle Integration Cloud Service topic available in Related Topics section to find more details about all the integrations.

After you complete the configuration of connections, clone each of the integrations listed in Integrations table in Integration Artifacts Reference for Integration Cloud Service from Related Topics, using New Integration Name and New Identifier columns. For more information on steps to clone an integration, see Cloning an Integration topic in Oracle Integration Cloud service documentation.

After you clone an integration, you can activate that integration to the runtime environment. See Activating an Integration topic in Oracle Integration Cloud service documentation.

Update AIAConfigurationProperties.xml with ICS WSDL End Points

If the new identifier name of the cloned integration is different from what is listed in the Integrations table, note down the ICS WSDL URL displayed over the integration, after you activate it. You must update this value in the AIA configuration property .xml file.

Use the following instructions to update the AIAConfigurationProperties.xml file.

Note: The following steps are to be performed only when ICS integrations are cloned using an identifier different from the one listed in the New Identifier column.

1. Connect to AIA host and edit AIAConfigurationProperties.xml file with details of the new end points.
   - The property file is deployed by default with ICS WSDL end points of the cloned New Identifier. Change the ICS WSDL name accordingly. WSDL end points are to be obtained from the service port SOAP address location available at the end in the ICS WSDL. For example, WSDL URL for I_OSC_AIA_ACCOUNT_CREATE is https://<ICS_IP>:<ICS_PORT>/integration/flowsvc/soap/I_OSC_AIA_ACCOUNT_CREATE/v01/?wsdl. If the identifier of the cloned integration is different then update the WSDL URL as: https://<ICS_IP>:<ICS_PORT>/integration/flowsvc/soap/<new-integration-Identifier>/v01/?wsdl.
   - Update the ICS end points with the new identifiers used while cloning the integration and save the file.

2. Deploy the AIAConfigurationProperties.xml file on AIA server.
**Note:** For more information on working with AIAConfigurationProperties.xml file, see How to work with AIAConfigurationProperties.xml in $AIA_HOME/aia_instances/$INSTANCE_NAME/AIAMetaData/config in Oracle Fusion Middleware Developer’s Guide for Oracle Application Integration Architecture Foundation Pack guide.

**Related Topics**

- How to work with AIAConfigurationProperties.xml in $AIA_HOME/aia_instances/$INSTANCE_NAME/AIAMetaData/config

---

### Configuring Multiple Email IDs in Integration Service Mail Notification

To configure multiple email IDs with greater character length, you must edit the Oracle Integration Cloud Service and add the email IDs.

If there are only one or two email IDs to configure, you can opt to configure the lookups. See the Adding Emails to the Lookup for Integration Notifications: Explained topic in the Integrating Financial Services with Oracle Banking Platform and Siebel UCM guide.

Following are the steps for editing the Oracle Integration Cloud Service to configure multiple email IDs:

1. Sign into Integration Cloud Service (ICS).
2. Navigate to **Integrations**.
3. Open the integrations listed in the following table in **Edit** mode.

   **Note:** You must deactivate the integration to open in **Edit** mode.

4. Open the Mail Configurations listed in the following table in **Edit** mode.
5. Enter **To Address** in the expression input box.
   - Enter comma separated addresses in the case to send the notification to multiple recipients.
   - Enter the whole set of email ID in single quotes.
6. Click **Validate and Close**.
7. Repeat steps 2 to 5 for all the integrations listed in the following table.

<table>
<thead>
<tr>
<th>Integration</th>
<th>Mail Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>I_OSC_AIA_Account_Create_Cxfins</td>
<td>MailConf</td>
</tr>
<tr>
<td>I_OSC_AIA_Account_Update_Cxfins</td>
<td>MailCnf</td>
</tr>
<tr>
<td></td>
<td>MailSkippedContactDetails</td>
</tr>
<tr>
<td>I_OSC_AIA_Cxfs_Customer_Update</td>
<td>MailConf</td>
</tr>
<tr>
<td></td>
<td>MailSkippedAccountDetails</td>
</tr>
</tbody>
</table>

---

103
Adding Emails to the Lookup for Integration Notifications: Explained

This integration comes with preconfigured lookup that you can use for sending notifications. One example of such notification is an email that is sent when Account - Contact Association synchronization fails and the association is skipped from the synchronization. This information is sent to the integration administrators over notification emails.

To add values to this lookup:

1. Sign into Integration Cloud Service (ICS).
2. Navigate to Designer > Lookups.
3. Click the lookup NOTIFICATIONMAIL_LOOKUP.
4. Update the email addresses USER_EMAIL_ID_FROM and USER_EMAIL_ID_TO. To add multiple emails, use comma separated list.
5. Click Save.

Setting Up Error Notifications: Explained

Use this topic to set up email notifications to notify users by email with hourly or daily reports about total messages received, total messages processed, successful messages, failed messages, and successful message rate; with service failure alerts; or with a detailed periodic report when an integration failure occurs.

To configure error notification in ICS, see Sending Service Failure Alerts, System Status Reports, and Integration Error Reports by Notification Emails section in Administering Oracle Integration Cloud Service guide.

Related Topics

- Sending Service Failure Alerts, System Status Reports, and Integration Error Reports by Notification Emails in ICS
- Tracing and Managing Synchronization Errors in Integration Cloud Service
Integration Artifacts Reference for Oracle Integration Cloud Service

This topic lists the artifacts that are included in the Integration Cloud Service package.

Integrations

Following integrations are included in the ICS integration package:

<table>
<thead>
<tr>
<th>Name</th>
<th>Identifier</th>
<th>New Integration Name</th>
<th>New Identifier</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>I_ OSC_ AIA_ Account_ Create_Cxfins</td>
<td>I_ OSC_ AIA_ ACCOUNT_CREATE_CXFIN</td>
<td>I_ OSC_ AIA_ Account_ Create</td>
<td>I_ OSC_ AIA_ ACCOUNT_CREATE</td>
<td>To synchronize account create from Oracle Sales Cloud to Siebel UCM.</td>
</tr>
<tr>
<td>I_ OSC_ AIA_ Account_ Update_Cxfins</td>
<td>I_ OSC_ AIA_ ACCOUNT_UPDATE_CXFIN</td>
<td>I_ OSC_ AIA_ Account_ Update</td>
<td>I_ OSC_ AIA_ ACCOUN_UPDATE</td>
<td>To synchronize account update from Oracle Sales Cloud to Siebel UCM.</td>
</tr>
<tr>
<td>I_ OSC_ AIA_ Contact_ Create_Cxfins</td>
<td>I_ OSC_ AIA_ CONTACT_CREATE_CXFIN</td>
<td>I_ OSC_ AIA_ Contact_ Create</td>
<td>I_ OSC_ AIA_ CONTACT_CREATE</td>
<td>To synchronize contact create from Oracle Sales Cloud to Siebel UCM.</td>
</tr>
<tr>
<td>I_ OSC_ AIA_ Contact_ Update_Cxfins</td>
<td>I_ OSC_ AIA_ CONTACT_UPDATE_CXFIN</td>
<td>I_ OSC_ AIA_ Contact_ Update</td>
<td>I_ OSC_ AIA_ CONTACT_UPDATE</td>
<td>To synchronize contact update from Oracle Sales Cloud to Siebel UCM.</td>
</tr>
<tr>
<td>I_ OSC_ AIA_ AccountResponse_ Cxfins</td>
<td>I_ OSC_ AIA_ ACCOUNTR_CXFIN</td>
<td>I_ OSC_ AIA_ AccountResponse</td>
<td>I_ OSC_ AIA_ACCOUNTR</td>
<td>Account response flow to create original system reference in Oracle Sales Cloud.</td>
</tr>
<tr>
<td>I_ OSC_ AIA_ ContactResponse_ Cxfins</td>
<td>I_ OSC_ AIA_ CONTACTR_CXFIN</td>
<td>I_ OSC_ AIA_ ContactResponse</td>
<td>I_ OSC_ AIA_CONTACTR</td>
<td>Contact response flow to create original system reference in Oracle Sales Cloud.</td>
</tr>
<tr>
<td>I_ AIA_ OSC_ Account_ Sync_Cxfins</td>
<td>I_ AIA_ OSC_ ACCOUN_SYNC_CXFIN</td>
<td>I_ AIA_ OSC_ Account_Sync</td>
<td>I_ AIA_ OSC_ACCOUN_SYNC</td>
<td>Synchronize account from Siebel UCM to Oracle Sales Cloud.</td>
</tr>
<tr>
<td>I_ AIA_ OSC_ Contact_ Sync_Cxfins</td>
<td>I_ AIA_ OSC_ CONTAC_SYNC_CXFIN</td>
<td>I_ AIA_ OSC_ Contact_Sync</td>
<td>I_ AIA_ OSC_CONTAC_SYNC</td>
<td>Synchronize contact from Siebel UCM to Oracle Sales Cloud.</td>
</tr>
<tr>
<td>I_ OSC_ OBP_ OPPORT_SUBMISSION_Cxfins</td>
<td>I_ OSC_ OBP_ OPPORT_SUBMISSION_CXFIN</td>
<td>I_ OSC_ OBP_ OPPORT_SUBMISSION</td>
<td>I_ OSC_ OBP_OPPORT_SUBMIS</td>
<td>Submission creation in Oracle Banking Platform.</td>
</tr>
<tr>
<td>Name</td>
<td>Identifier</td>
<td>New Integration Name</td>
<td>New Identifier</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>------------</td>
<td>----------------------</td>
<td>----------------</td>
<td>-------------</td>
</tr>
<tr>
<td>I_OBP_OSC_UPDATE_OPPORTUNITY_Cxfins</td>
<td>I_OBP_OSC_UPDATE_OPPORT_CXRIN</td>
<td>I_OBP_OSC_UPDATE_OPPORTUNITY</td>
<td>I_OBP_OSC_UPDATE_OPPORT</td>
<td>Opportunity update from Oracle Banking Platform.</td>
</tr>
</tbody>
</table>

### Lookups

Following lookups are included in the ICS integration package:

<table>
<thead>
<tr>
<th>New Lookup Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>COUNTRY_LOOKUP</td>
<td>Country Lookup</td>
</tr>
<tr>
<td>STATE_LOOKUP</td>
<td>State Lookup</td>
</tr>
<tr>
<td>ACCOUNT_TYPE_LOOKUP</td>
<td>Account Type Lookup</td>
</tr>
<tr>
<td>ONBOARDINGSTATUS_LOOKUP</td>
<td>Onboarding Status Lookup</td>
</tr>
<tr>
<td>KYCSTATUS_LOOKUP</td>
<td>KYC Status Lookup</td>
</tr>
<tr>
<td>PARTYTYPE_LOOKUP</td>
<td>Party Type Lookup</td>
</tr>
<tr>
<td>FINACCOUNT_ACCOUNTTYPE</td>
<td>Financial Account Type Lookup</td>
</tr>
<tr>
<td>FINACCOUNT_STATUS</td>
<td>Financial Account Status Lookup</td>
</tr>
<tr>
<td>SUBMISSIONSTATUS_LOOKUP</td>
<td>Submission Status Lookup</td>
</tr>
<tr>
<td>PRODUCTSTATUS_LOOKUP</td>
<td>Product Status Lookup</td>
</tr>
<tr>
<td>CONTACTPOINT_PHONETYPE_LOOKUP</td>
<td>Contact Point Phone Type Lookup</td>
</tr>
<tr>
<td>CONTACTPOINT_PURPOSE_LOOKUP</td>
<td>Contact Point Purpose Lookup</td>
</tr>
<tr>
<td>OSC_RELTYPE_LOOKUP</td>
<td>Relationship Type Lookup</td>
</tr>
<tr>
<td>NOTIFICATIONMAIL_LOOKUP</td>
<td>Notification Email Lookup</td>
</tr>
<tr>
<td>OSC_RELCODE_LOOKUP</td>
<td>Relationship Code Lookup</td>
</tr>
</tbody>
</table>
## Connections

Following connections are included in the ICS integration package:

<table>
<thead>
<tr>
<th>New Connection Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>C_OSC</td>
<td>Oracle Sales Cloud adapter connection</td>
</tr>
<tr>
<td>CUSTOMERPARTYEBM</td>
<td>SOAP adapter connection</td>
</tr>
<tr>
<td>C_AIA_ACCOUNT</td>
<td>SOAP adapter connection</td>
</tr>
<tr>
<td>C_AIA_Contact</td>
<td>SOAP adapter connection</td>
</tr>
<tr>
<td>CustomerPartyResponseEBM</td>
<td>SOAP adapter connection</td>
</tr>
<tr>
<td>C_AIA_ASSET</td>
<td>SOAP adapter connection</td>
</tr>
<tr>
<td>C_OBP_SUBMISSION</td>
<td>SOAP adapter connection</td>
</tr>
<tr>
<td>C_OBP_APPLICATION</td>
<td>SOAP adapter connection</td>
</tr>
</tbody>
</table>
8 Configuring Application Integration Architecture

Configuration and Deployment of Integration: Overview

The following are the major steps involved in overall configuration and deployment of AIA (Application Integration Architecture):

1. Install and deploy Application Integration Architecture Foundation Pack 11g R1 (11.1.7.0).
   For detailed instructions, see Installation and Upgrade Guide in Oracle Fusion Middleware Online Documentation Library 11g Release 1 (11.1.1.7) at https://docs.oracle.com/cd/E28280_01/nav/aia.htm
   For detailed instructions, see Installing Pre-Built Integrations in Application Integration Architecture Installation and Upgrade Guide for Pre-Built Integrations at https://docs.oracle.com/cd/E58077_01/doc.116/e55200/toc.htm
3. Copy and Deploy the integration package.
   For detailed instructions, see Installing Pre-Built Integrations in Application Integration Architecture Installation and Upgrade Guide for Pre-Built Integrations at https://docs.oracle.com/cd/E58077_01/doc.116/e55200/toc.htm

Note:

1. The integration services deployment instructions are specific to Linux based platforms.
2. Several components of Customer MDM Base Pack are used in integration flows. You must ensure that proper backups of existing configurations are obtained as installing backpack with overwrite existing configurations. For more information, see https://docs.oracle.com/cd/E58077_01/doc.116/e55200/installpips.htm#PIINU193
3. You must undeploy any existing AIA integration package, before deploying a new package.

Copying Integration Artifacts

Copy the integration artifacts manually from the <cxfins.zip>/aia directory in the collateral Integrating Financial Services with Oracle Banking Platform and Siebel UCM (2293151.1) on My Oracle Support at https://support.oracle.com.

Note: You must backup the existing AIA files as they are overwritten when you deploy the integration.

Copy the contents of the following directories to AIA host machine:

<table>
<thead>
<tr>
<th>Source Directory</th>
<th>Destination</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;cxfins.zip&gt;/aia/ AIAMetaData</td>
<td>&lt;AIA_HOME&gt;/AIAMetaData</td>
</tr>
</tbody>
</table>
Deploying Integration Artifacts Using AIA Installer

Use instructions in this topic to deploy the integration artifacts using AIA installer.

**Note:** You must undeploy any existing integrations before performing the steps in this topic.

To deploy integration artifacts:

1. Edit `<AIA_HOME>/aia_instances/<INSTANCE_NAME>/config/AIAInstallProperties.xml` to add target ICS server details:
   
   a. If an entry for `ics` exists in `participatingapplications` element, verify that the details are correctly added.  
   b. If an entry for `ics` does not exist in `participatingapplications` element, add the following section in `participatingapplications` element:

   ```xml
   <ics>
     <internal>
     <id>ICS_SERVER_ID</id>
     </internal>
     <host>ICS_HOST</host>
     <port>ICS_PORT</port>
     <version>ICS_VERSION</version>
     <InternetProtocol>ICS_PROTOCOL</InternetProtocol>
     </server>
   </db>
   </ics>
   ```

   Replace the following:
   
   - **ICS_SERVER_ID** - Unique name representing ICS server.  
   - **ICS_HOST** - Complete host name of the ICS server on which ICS integrations will be deployed.  
   - **ICS_PORT** - Port value for ICS integration service WSDL URLs.  
   - **ICS_VERSION** - Version of ICS used in integration.  
   - **ICS_PROTOCOL** - Protocol to be used for ICS integration service WSDL URLs. Possible values: `http://` and `https://`.

2. Open a new terminal and access the `<AIA_HOME>/aia_instances/<INSTANCE_NAME>/bin` folder.
3. Source `aiaenv.sh` by executing the following command, to set the environment variables:
source aiaenv.sh

4. Run the following command to deploy the integrations:

```bash
sh </pips/BankingAndWealthManagement/CXFINS/DeploymentPlans/deployCXFINS.sh>
```

**Note:**
- Replace `<AIA_HOME>` in instructions with actual path to installed AIA Foundation Pack Home directory.
- Replace `<INSTANCE_NAME>` in instructions with actual AIA Instance Name used in AIA installation.

---

### Deploying AIA Metadata File Manually: Procedure

Use the instructions provided in this topic to manually deploy AIA metadata files.

Perform the following steps to deploy the files manually:

1. Access `$AIA_HOME/aia_instances/$INSTANCE_NAME/bin` folder.
2. Run the following command:

```bash
source aiaenv.sh
```

4. Open the deployment plan file, `UpdateMetaDataDP.xml`.
5. Update the file by inserting `include` tags for each resource group to be added to the MDS:

   a. To upload all files in `AIAMetadata`, add the following:
   ```xml
   <include name='AIAMetadata/**'/>
   ```
   
   b. To upload AOL objects to MDS, for example, to upload SEBL AOL objects in the `AIAComponents/ApplicationObjectLibrary/SEBL/schemas` folder, add the following:
   ```xml
   <include name='AIAComponents/ApplicationObjectLibrary/SEBL/schemas/**'/>
   ```

   **Note:** In the include tag, the folder path must be relative to the folder `AIAMetadata`.

6. Access the `AIA_HOME/Infrastructure/Install/config` folder.
7. Execute the `UpdateMetaData.xml` script using the following command:

```bash
ant -f UpdateMetaData.xml
```
Verifying the Deployment

Once you have deployed your integrations on AIA, you must verify it and also validate security policies.

Verifying the Deployment

To verify the integration services deployment:

1. Open the log files from the following location and look for warnings and error messages: `<AIA_INSTANCE>/logs/CXFINS-Deploy.log`
2. Confirm that the integration services have been installed.
   a. Navigate to the Oracle Enterprise Manager Console (EM Console) using http://<server name>/em/.
   b. Sign in using the server admin user name.
   c. Expand Farm_soa_domain > SOA > soa-infra > Default and look for the following items:
      • BankingAndWealthManagementFinancialAccountEBSV1
      • BankingAndWealthManagementFinancialAccountResponseEBSV1
      • CustomerPartyEBSV2
      • CustomerPartyResponseEBSV2
      • SyncCustomerPartyListICSProvABCSImp1
      • SyncCustomerPartyListResponseICSProvABCSImp1
      • SyncFinancialAccountListICSProvABCSImp1
      • SyncCustomerPartyListICSReqABCSImp1
      • SyncCustomerPartyListResponseUCMProvABCSImp1
      • SyncCustomerPartyListUCMProvABCSImp1
      • SyncOrganizationUCMReqABCSImp1
      • SyncPersonUCMReqABCSImp1
      • SyncFinancialAssetUCMJMSConsumer
      • SyncFinancialAssetUCMJMSProducer
      • SyncFinancialAccountListResponseUCMProvABCSImp1
      • SyncFinancialAssetUCMReqABCSImp1

Validating Security Policies

Individual services for this integration have locally attached security policies, which must be validated.

To validate locally attached security policies:

1. Sign into Oracle Enterprise Manager Fusion Middleware Control (EM Console).
2. Navigate to WebLogic Domain > soa_domain > Web Services > Policies.
3. Verify the Service Policy attachment.
   a. Find the service policy in the list of policies.
   b. Click the number in Attachment Count column.
   c. This opens Usage Analysis screen.
   d. Change the Subject Type list box to SOA Service.
e. Validate that all the composites are listed with local attachment to this service policy.

4. Verify Client Policy attachment.
   
a. Navigate back to Policies screen and find the client policy.
b. Click the number in Attachment Count column.

   This opens Usage Analysis screen.

c. Change the Subject Type list box to **SOA Reference**.
d. Using the following tables, Validate that all the composites are listed with local attachment to this client policy and attached to the correct references.

No Authentication Service Policy Attachments for the integration services:

<table>
<thead>
<tr>
<th>Composite</th>
<th>Service Policy</th>
</tr>
</thead>
<tbody>
<tr>
<td>SyncFinancialAssetUCMJMSProducer</td>
<td>oracle/no_authentication_service_policy</td>
</tr>
</tbody>
</table>

Client Policy Attachments for integration services:

<table>
<thead>
<tr>
<th>Composite</th>
<th>Reference</th>
<th>Client Policy</th>
<th>CSF Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>SyncCustomerPartyListICSProvAE CustomerPartyAccount</td>
<td>oracle/wss_username_token_over_ssl_client_policy</td>
<td>ics_key</td>
<td></td>
</tr>
<tr>
<td>SyncCustomerPartyListICSProvAE CustomerPartyContact</td>
<td>oracle/wss_username_token_over_ssl_client_policy</td>
<td>ics_key</td>
<td></td>
</tr>
<tr>
<td>SyncCustomerPartyListResponseICSProvABCSImpl CustomerPartyAccountResponse</td>
<td>oracle/wss_username_token_over_ssl_client_policy</td>
<td>ics_key</td>
<td></td>
</tr>
<tr>
<td>SyncCustomerPartyListResponseICSProvABCSImpl CustomerPartyContactResponse</td>
<td>oracle/wss_username_token_over_ssl_client_policy</td>
<td>ics_key</td>
<td></td>
</tr>
<tr>
<td>SyncFinancialAccountListICSProvABCSImpl FinancialAccountEBMPort</td>
<td>oracle/wss_username_token_over_ssl_client_policy</td>
<td>ics_key</td>
<td></td>
</tr>
</tbody>
</table>

**Undeploying Integration Package**

Use the instructions in this topic to undeploy all the Metadata and SOA composites used in this integration.

1. Open a new terminal and access the `<aia_instances>/bin` directory.
2. Source `aiaenv.sh` by executing the following command, to set the environment variables: `source aiaenv`.
3. Run following command to undeploy the integration: `sh <AIA_HOME>/pips/BankingAndWealthManagement/CXFINS/DeploymentPlans/undeployCXFINS.sh`
Importing ICS Certificate to AIA

Oracle Integration Cloud Service (ICS) Integration services are SSL enabled, so you must import ICS certificate for communication between AIA and ICS.

Before you begin, export certificates from ICS. The instructions for exporting certificates, using https URL in the web browser are dependent on the client web browser and the version. Use the relevant documentation provided by your browser service provider to export the certificates.

To import ICS certificate,

1. Copy the certificate to AIA.
2. Run the following command:

   keytool -importcert -trustcacerts -keystore DemoTrust.jks -alias <ICS-host-name> -file <ICS-certificate-name>.crt

   Provide the password when prompted.
3. Enter yes when prompted about trusting the certificate.
4. Restart AIA server.

Creating CSF Map

You must create the CSF map and configure the CSF key in Oracle Enterprise Manager Fusion Middleware Control before you can manage and monitor the ICS SOAP Adapter in an application during runtime. The same CSF key configuration details that you specify at design time in the Adapter Configuration Wizard must be specified in the credential store of the Oracle Weblogic Server domain on which to deploy SOA composite application or Oracle Service Bus business service.

Use the following instructions to create a CSF key:

1. Sign into Oracle Enterprise Manager for the CRM Domain.
2. Expand the Farm_CRMDomain node.
3. Expand WebLogic Domain and right-click CRMDomain.
4. Click Security > Credentials.
5. Click Create Map.
6. If the Map does not already exist, enter oracle.wsm.security as Map Name.
7. On Create Key dialog box, enter the following:
   - Select Map: oracle.wsm.security
   - Key: ics-key
   - Type: Password
   - User Name: User name, which is used to access Integration Cloud Service.
   - Password: Password, which is used to access Integration Cloud Service.
8. Click OK.
Registering ICS as a System in AIA

You must register ICS as a system to be integrated with AIA. This is used by services to obtain system specific information at run-time.

1. Sign into the AIA Home Page using \texttt{http://<host>:<port>/AIA}.
2. In the Setup area, click \textit{Go}.
4. Click \textbf{Create} to register ICS as a new system.
5. Enter values as listed in the following table:

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internal ID</td>
<td>ICS</td>
</tr>
<tr>
<td>System Code</td>
<td>ICS_01</td>
</tr>
<tr>
<td>System Description</td>
<td>Long description of the requester or provider system instance identified in the System Code field.</td>
</tr>
<tr>
<td>IP Address</td>
<td>IP address of the participating application by which you can access the participating application endpoint.</td>
</tr>
<tr>
<td>URL</td>
<td>This is the hostname and port combination specified in the uniform resource locator (URL) format by which one can access the participating application. Typically, this will be of the form http://&lt;hostname&gt;:&lt;port&gt;/</td>
</tr>
<tr>
<td>System Type</td>
<td>ICS</td>
</tr>
<tr>
<td>Application Type</td>
<td>ICS</td>
</tr>
<tr>
<td>Version</td>
<td>ICS version number</td>
</tr>
<tr>
<td>Contact Name</td>
<td>Name of the contact responsible for the system.</td>
</tr>
<tr>
<td>Contact Phone</td>
<td>Phone number of the contact responsible for the system.</td>
</tr>
<tr>
<td>Contact Email</td>
<td>Email address of the contact responsible for the system.</td>
</tr>
</tbody>
</table>

6. Click \textbf{Save}.

Setting Up Error Notifications in AIA

You can configure error notification emails to receive information about synchronization failures. You can also modify the subject line and body text of emails issued by error notification functionality.
To display error message seen in the AIA failed instance in the notification email body, AIAEHNotification.xml file can be modified to add the following line to body tag: Error seen is #XPATH.default:FaultMessage/default:Text##

**Note:** For more information about setting up and configuring error notifications in AIA, see Related Topics section.

**Related Topics**

- Setting Up Error Handling in AIA
- Customizing Error Notification Emails

### Tuning AIA

You must tune AIA to avoid failures due to lock time-outs in long transactions.

It is recommended to use a DISTRIBUTED_LOCK_TIMEOUT that is long enough to account for likely integration delay at load. Additionally, it is recommended that the EJB Timeout, JAT Timeout, and syncMaxWaitTime parameters be aligned such that DISTRIBUTED_LOCK_TIMEOUT > EJB Timeout > JTA Timeout > syncMaxWaitTime.

The the following values as example:

- distributed_lock_timeout: 360
- EJB Timeout: 300
- JTA Timeout: 90
- syncMaxWaitTime: 45

If these values are too low or improperly set with reference to each other, you may see the following error: oracle.tip.xref.exception.RepositoryException: Unable to access Cross Reference Values from Database. The SQL Exception is: ORA-02049: timeout: distributed transaction waiting for lock.

### Steps to Resolve Concurrency Issues During On-boarding

To resolve any concurrency issue during on-boarding, increase or decrease the wait time by using the following steps:

1. Open the project SyncPersonUCMReqABCSImpl.
2. Edit file SyncPersonUCMReqABCSImplProcess.bpel
3. Search for Switch tag with name as Switch1.
   
   Edit this switch wait time by modifying the for parameter value for wait element with name as Wait.
   
   For example, to set wait time as 20 seconds, edit the line as: <wait name="Wait1" for="PT20S"/>
4. Save the file and build and deploy the composite in MDS.

Similarly, make changes for SyncOrganizationUCMReqABCSImpl and SyncOrganizationUCMReqABCSImplProcess.bpel files.
# Integration Artifacts Reference for AIA

## Artifacts Available as Collateral

This topic describes the artifacts that are available as a part of collateral directory `<cxfins.zip>/aia`.

### AIA Metadata Files

The following table lists the existing MDS (Meta Data Services) repository related artifacts present in MDM basepack that are modified and newly added files.

<table>
<thead>
<tr>
<th>Existing files that are modified</th>
<th>New files</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/dvm/TARGETIdvm</code></td>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/dvm/CUSTOMERPARTYONBOARDINGSTATUS dvm</code></td>
</tr>
<tr>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/dvm/SOURCEIdvm</code></td>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/dvm/CUSTOMERPARTYRELATIONSHIPCODE dvm</code></td>
</tr>
<tr>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/dvm/STATEdvm</code></td>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/dvm/CUSTOMERPARTYCOMMUNICATION PURPOSE.dvm</code></td>
</tr>
<tr>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/dvm/ADDRESSCOUNTRYIdvm</code></td>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/dvm/ FINANCIALACCOUNTORG HOLDER_ RELATIONSHIP_ CODE.dvm</code></td>
</tr>
<tr>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/dvm/CUSTOMERPARTYACCOUNTTYPECODE dvm</code></td>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/dvm/ FINANCIALACCOUNTPERSON HOLDER_ RELATIONSHIP_ CODE.dvm</code></td>
</tr>
<tr>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/dvm/CUSTOMERPARTYACCOUNTCLASSCODE dvm</code></td>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/dvm/ FINANCIALACCOUNTSTATUSCODE dvm</code></td>
</tr>
<tr>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/xref/CUSTOMERPARTYACCOUNTIDxref</code></td>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/dvm/ FINANCIALACCOUNTTYPECODE dvm</code></td>
</tr>
<tr>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/xref/CUSTOMERPARTYACCOUNTTYPEIDxref</code></td>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/dvm/ FINANCIALACCOUNTACCOUNTTYPECODE dvm</code></td>
</tr>
<tr>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/xref/CUSTOMERPARTYRELATIONSHIPIDxref</code></td>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/dvm/ FINANCIALACCOUNTBILLINGCODE dvm</code></td>
</tr>
<tr>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/xref/CUSTOMERPARTYLOCATIONIDxref</code></td>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/dvm/ FINANCIALACCOUNTRELATIONSHIPPCODE dvm</code></td>
</tr>
<tr>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/xref/CUSTOMERPARTYADDRESSIDxref</code></td>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/dvm/ FINANCIALACCOUNTHOLDERIDxref</code></td>
</tr>
<tr>
<td>Existing files that are modified</td>
<td>New files</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>-----------</td>
</tr>
<tr>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/ xref/ CUSTOMERPARTYCONTACT EMAILCOMMID. xref</code></td>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/ xref/ FINANCIALACCOUNTIDxref</code></td>
</tr>
<tr>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/ xref/ CUSTOMERPARTYCONTACT FAXCOMMID. xref</code></td>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/ xref/ CUSTOMERPARTYCONTACT ALT_EMAILCOMMID. xref</code></td>
</tr>
<tr>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/ xref/ CUSTOMERPARTYCONTACT PHONECOMMID. xref</code></td>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/ xref/ CUSTOMERPARTYCONTACT ALT_PHONECOMMID. xref</code></td>
</tr>
<tr>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/ xref/ ORGANIZATIONIDxref</code></td>
<td>Not applicable</td>
</tr>
<tr>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/ xref/ CUSTOMERPARTYACCOUNT FAXCOMMID. xref</code></td>
<td>Not applicable</td>
</tr>
<tr>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/ xref/ CUSTOMERPARTYACCOUNT PHONECOMMID. xref</code></td>
<td>Not applicable</td>
</tr>
<tr>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/ xref/ CUSTOMERPARTYACCOUNT WEBCOMMID. xref</code></td>
<td>Not applicable</td>
</tr>
<tr>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/ xref/ CUSTOMERPARTYPARTYCONTACTID xref</code></td>
<td>Not applicable</td>
</tr>
<tr>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/ AIAComponents/ ApplicationObjectLibrary/ UCM/ SIA810/ schemas/ ListOfSwiOrganizationIOxsd</code></td>
<td>Not applicable</td>
</tr>
<tr>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/ AIAComponents/ ApplicationObjectLibrary/ UCM/ SIA810/ schemas/ ListOfSwiOrganizationPublishIOxsd</code></td>
<td>Not applicable</td>
</tr>
<tr>
<td><code>&lt;AIA_HOME&gt;/ AIAMetaData/ AIAComponents/ ApplicationObjectLibrary/ UCM/ SIA810/ schemas/ ListOfSwiPersonIOxsd</code></td>
<td>Not applicable</td>
</tr>
</tbody>
</table>
## AIA Projects Files

The following table lists the modified and newly added AIA projects directories, that contains SOA composites related files, deployed on the server.

<table>
<thead>
<tr>
<th>New files</th>
<th>Existing files that are modified</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;AIA_ HOME&gt;/services/industry \BankingAndWealthManagement \UCM/AdapterServices \SyncFinancialAssetUCM/JMSProducer</td>
<td>&lt;AIA_ HOME&gt;/services/ core/ UCM/ ProviderABCS/ SyncOrganizationUCMReqABCSImpl</td>
</tr>
<tr>
<td>&lt;AIA_ HOME&gt;/services/industry \BankingAndWealthManagement \UCM/AdapterServices \SyncFinancialAssetUCM/JMConsumer</td>
<td>&lt;AIA_ HOME&gt;/services/ core/ UCM/ RequesterABCS/ SyncPersonUCMReqABCSImpl</td>
</tr>
<tr>
<td>&lt;AIA_ HOME&gt;/services/industry \BankingAndWealthManagement \UCM/RequesterABCS \SyncFinancialAssetUCMReqABCSImpl</td>
<td>&lt;AIA_ HOME&gt;/services/ core/ UCM/ ProviderABCS/ SyncCustomerPartyListUCMProvABCSImpl</td>
</tr>
<tr>
<td>&lt;AIA_ HOME&gt;/pips \BankingAndWealthManagement \CXFIN/EBSS/FinancialAccount \BankingAndWealthManagementFinancialAccount</td>
<td>&lt;AIA_ HOME&gt;/pips/ MDMCustomer/EBS/CustomerPartyEBSV2</td>
</tr>
<tr>
<td>&lt;AIA_ HOME&gt;/pips \BankingAndWealthManagement \CXFIN/EBSS/FinancialAccount \BankingAndWealthManagementFinancialAccount</td>
<td>&lt;AIA_ HOME&gt;/pips/ MDMCustomer/EBS/CustomerPartyEBSV2</td>
</tr>
</tbody>
</table>
New files | Existing files that are modified
---|---
<AIA_HOME>/services/\core/ICS/ProviderABCS/\SyncFinancialAccountListICSProvABCSImpl | <AIA_HOME>/pips/MDMCustomerEBSS/\CustomerPartyResponseEBSV2

 | <AIA_HOME>/services/industry/\BankingAndWealthManagement/\UCM/\ProviderABCS/\SyncFinancialAccountListResponseUCMProvABCSImpl

 | Not applicable

 | Not applicable

 | Not applicable

Information Related to Artifact Changes

This topic describes the artifacts that are changed as a part of collateral directory <cxfins.zip>/aia.

Business Mappings

The following table provides all the changes in business mappings:

<table>
<thead>
<tr>
<th>AIA Project</th>
<th>File Name</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>SyncOrganizationUCMReqABCSImpl</td>
<td>1. xsl/ListOfSwiOrganizationPublishIO</td>
<td>Transformations related to field mapping for the integration.</td>
</tr>
<tr>
<td></td>
<td>2. xsl/ListOfSwiOrganizationPublishIO_Custom</td>
<td>xsl</td>
</tr>
<tr>
<td>SyncOrganizationUCMReqABCSImpl</td>
<td>xsl</td>
<td>A switch is introduced to add wait, if DueDiligenceStatus is not empty.</td>
</tr>
<tr>
<td>SyncPersonUCMReqABCSImpl</td>
<td>1. xsl/XformUCMPersonToSyncCustomerPa</td>
<td>Transformations related to field mapping for the integration.</td>
</tr>
<tr>
<td></td>
<td>2. xsl/XformUCMPersonToSyncCustomerPa:</td>
<td></td>
</tr>
<tr>
<td>SyncPersonUCMReqABCSImpl</td>
<td>SyncPersonUCMReqABCSImplProcess.bpm</td>
<td>A switch is introduced to add wait, if DueDiligenceStatus is not empty.</td>
</tr>
<tr>
<td>SyncCustomerPartyListUCMProvABCSImpl</td>
<td>1. xsl/XfromUCMPersonToSyncCustomerPa:</td>
<td>Transformations related to field mapping for the integration.</td>
</tr>
</tbody>
</table>
ICS Routing

BankingAndWealthManagementFinancialAccountEBSV1 and BankingAndWealthManagementFinancialAccountResponseEBSV1 are the new EBS (Enterprise Business Services) created to route FinancialAccount to ICS.

Changes for routing the existing EBS to ICS are listed as follows.

<table>
<thead>
<tr>
<th>AIA Project</th>
<th>File Name</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CustomerPartyEBSV2</td>
<td>CustomerPartyEBSV2. componentType</td>
<td>A reference for SyncCustomerPartyListICSProvABCSImpl has been added.</td>
</tr>
<tr>
<td>CustomerPartyEBSV2</td>
<td>composite.xml</td>
<td>A reference for ICS provider SyncCustomerPartyListICSProvABCSImpl has been added.</td>
</tr>
<tr>
<td>CustomerPartyResponseEBSV2</td>
<td>CustomerPartyResponseEBSV2. componentType</td>
<td>A reference for SyncCustomerPartyListResponseICSProvABCSImpl has been added.</td>
</tr>
<tr>
<td>CustomerPartyResponseEBSV2</td>
<td>composite.xml</td>
<td>A reference for Response ICS provider SyncCustomerPartyListResponseICSProvABCSImpl has been added.</td>
</tr>
</tbody>
</table>

Siebel UCM Inbound Security

The following table lists changes in Siebel UCM Inbound Security.

<table>
<thead>
<tr>
<th>AIA Project</th>
<th>Files</th>
<th>What Has Changed</th>
</tr>
</thead>
<tbody>
<tr>
<td>SyncCustomerPartyListUCMProvABCSImpl</td>
<td>SyncCustomerPartyListUCMProvABCSImplProcess.bpel</td>
<td>An Assign block AssignSecurityCredentials is added before invoking Siebel UCM PersonService and OrganizationService to assign the target endpoint user name and password.</td>
</tr>
</tbody>
</table>
AIA Project                  | Files                                | What Has Changed                                                                 |
---                           | ---                                  | ---                                                                            |
SyncCustomerPartyListUCMProvABCSImpl  | OrganizationServiceRef.wsdl           | XSD files to create security header in the invocation to Siebel UCM are imported. |
SyncCustomerPartyListUCMProvABCSImpl  | AIAServiceConfigurationProperties.xml | The EndpointURI of UCM_01 has been changed from eai_ to eai_anon.                |
SyncCustomerPartyListUCMProvABCSImpl  | XSD Files                            | XSD directory to add security header in the invocation to Siebel UCM is added.   |
SyncCustomerPartyListResponseUCMProvABCSImpl  | XSD Files                            | XSD directory to add security header in the invocation to Siebel UCM is added. |
SyncCustomerPartyListResponseUCMProvABCSImpl  | OrganizationCrossReferenceServiceRef.wsdl | Imported XSD files to create security header in the invocation to Siebel UCM. |
SyncCustomerPartyListResponseUCMProvABCSImpl  | AIAServiceConfigurationProperties.xml | The EndpointURI of UCM_01 has been changed from eai_ to eai_anon.                |
SyncCustomerPartyListResponseUCMProvABCSImpl  | XSD Files                            | XSD directory to add security header in the invocation to Siebel UCM is added.   |

**Changes in AIAMETADATA**

The DVM files given in the following table have been edited to update the values for the column UCM_01 and COMMON.

<table>
<thead>
<tr>
<th>File Name</th>
<th>COMMON</th>
<th>UCM_01</th>
</tr>
</thead>
<tbody>
<tr>
<td>TARGET_ID.dvm</td>
<td>ICS_01</td>
<td>ICS01</td>
</tr>
<tr>
<td>SOURCE_ID.dvm</td>
<td>ICS_01</td>
<td>ICS_01</td>
</tr>
<tr>
<td>STATE.dvm</td>
<td>Alabama</td>
<td>Al</td>
</tr>
<tr>
<td>STATE.dvm</td>
<td>Northwest</td>
<td>NT</td>
</tr>
<tr>
<td>File Name</td>
<td>COMMON</td>
<td>UCM_01</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>------------</td>
<td>----------</td>
</tr>
<tr>
<td>STATE.dvm</td>
<td>Queensland</td>
<td>QLD</td>
</tr>
<tr>
<td>STATE.dvm</td>
<td>South Australia</td>
<td>SA</td>
</tr>
<tr>
<td>STATE.dvm</td>
<td>Western Australia</td>
<td>WA</td>
</tr>
<tr>
<td>STATE.dvm</td>
<td>TAS</td>
<td>TAS</td>
</tr>
<tr>
<td>STATE.dvm</td>
<td>ACT</td>
<td>ACT</td>
</tr>
<tr>
<td>STATE.dvm</td>
<td>NSW</td>
<td>NSW</td>
</tr>
<tr>
<td>STATE.dvm</td>
<td>VIC</td>
<td>VIC</td>
</tr>
<tr>
<td>ADDRESS_COUNTRYID.dvm</td>
<td>Viet Nam</td>
<td>VN</td>
</tr>
<tr>
<td>ADDRESS_COUNTRYID.dvm</td>
<td>United Kingdom</td>
<td>UK</td>
</tr>
<tr>
<td>ADDRESS_COUNTRYID.dvm</td>
<td>Singapore</td>
<td>SG</td>
</tr>
<tr>
<td>ADDRESS_COUNTRYID.dvm</td>
<td>New Zealand</td>
<td>NZ</td>
</tr>
<tr>
<td>ADDRESS_COUNTRYID.dvm</td>
<td>Japan</td>
<td>JP</td>
</tr>
<tr>
<td>ADDRESS_COUNTRYID.dvm</td>
<td>Indonesia</td>
<td>ID</td>
</tr>
<tr>
<td>ADDRESS_COUNTRYID.dvm</td>
<td>India</td>
<td>IN</td>
</tr>
<tr>
<td>ADDRESS_COUNTRYID.dvm</td>
<td>Hong Kong</td>
<td>HK</td>
</tr>
<tr>
<td>ADDRESS_COUNTRYID.dvm</td>
<td>China</td>
<td>CN</td>
</tr>
<tr>
<td>ADDRESS_COUNTRYID.dvm</td>
<td>Australia</td>
<td>AU</td>
</tr>
<tr>
<td>CUSTOMERPARTY_ACCOUNTTYPECODE.dvm</td>
<td>TRUST</td>
<td>TRU</td>
</tr>
<tr>
<td>CUSTOMERPARTY_ACCOUNTTYPECODE.dvm</td>
<td>LEGAL_ENTITY</td>
<td>LEG</td>
</tr>
<tr>
<td>CUSTOMERPARTY_ACCOUNTCLASSCODE.dvm</td>
<td>PROSPECT</td>
<td>Prospect</td>
</tr>
</tbody>
</table>
XREF Changes

A new column ICS_01 has been added in the following XREF files.

1. CUSTOMERPARTY_ACCOUNTID.xref
2. CUSTOMERPARTY_PARTYID.xref
3. CUSTOMERPARTY_CONTACTID.xref
4. CUSTOMERPARTY_PARTYLOCATIONID.xref
5. CUSTOMERPARTY_LOCATIONREFID.xref
6. CUSTOMERPARTY_ADDRESSID.xref
7. CUSTOMERPARTY_CONTACT_EMAILCOMMID.xref
8. CUSTOMERPARTY_CONTACT_FAXCOMMID.xref
9. CUSTOMERPARTY_CONTACT_PHONECOMMID.xref
10. ORGANIZATION_ID.xref
11. CUSTOMERPARTY_ACCOUNT_FAXCOMMID.xref
12. CUSTOMERPARTY_ACCOUNT_PHONECOMMID.xref
13. CUSTOMERPARTY_ACCOUNT_WEBCOMMID.xref
14. CUSTOMERPARTY_PARTYCONTACTID.xref
15. CUSTOMERPARTY_UCM_ADDRESSID.xref

ApplicationObjectLibrary

The following XSD files that are provided as a part of the integration contain the latest schemas extracted from Siebel UCM. For any additional changes manually merge the change in the delivered schemas.

1. UCM/SIA810/schemas/ListOfSwiFinancialAssetPublishIO.xsd
2. UCM/SIA810/schemas/ListOfSwiOrganizationIO.xsd
3. UCM/SIA810/schemas/ListOfSwiOrganizationPublishIO.xsd
4. UCM/SIA810/schemas/ListOfSwiPersonIO.xsd
5. UCM/SIA810/schemas/ListOfSwiPersonPublishIO.xsd

EnterpriseObjectLibrary

The following changes has been made in CustomCustomerPartyEBO.xsd file by adding new variables for owner and DueDiligenceDetail mapping.

1. New elements CustomCustomerPartyOwnerType and ListOfDueDiligenceDetailType are added to the existing complexType, CustomCustomerPartyEBOType.

   <xsd:complexType name="CustomCustomerPartyEBOType">
   <xsd:sequence>
      <xsd:element name="CustomerPartyOwner" type="CustomCustomerPartyOwnerType"/>
      <xsd:element name="ListOfDueDiligenceDetail" type="ListOfDueDiligenceDetailType" minOccurs="0"/>
   </xsd:sequence>
   </xsd:complexType>

2. The definitions for the new elements CustomCustomerPartyOwnerType and ListOfDueDiligenceDetailType have been added.

   <xsd:complexType name="CustomCustomerPartyOwnerType">
   <xsd:sequence>
      <xsd:element ref="corecom:Identification"/>
   </xsd:sequence>
   </xsd:complexType>

   <xsd:complexType name="ListOfDueDiligenceDetailType"
<xsd:complexType name="DueDiligenceHistoryDetailType">
  <xsd:sequence>
    <xsd:element name="DueDiligenceStatus" type="corecom:TextType" minOccurs="0"/>
    <xsd:element name="serialNumber" type="corecom:TextType" minOccurs="0"/>
    <xsd:element name="UpdatedDateTime" type="corecom:DateTimeType" minOccurs="0"/>
    <xsd:element name="IdentityVerificationHistoryDetail" type="IdentityVerificationHistoryDetailType" minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="IdentityVerificationHistoryDetailType">
  <xsd:sequence>
    <xsd:element name="DueDiligenceType" type="corecom:TextType" minOccurs="0"/>
    <xsd:element name="IdentityVerificationStatus" type="corecom:TextType" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
9 Tracing and Managing Synchronization Errors

Tracing and Managing Synchronization Errors in Integration Cloud Service

This topic covers different ways to trace integration related failures in Integration Cloud Service. Integration Cloud Service offers various error monitoring capabilities to trace your failures.

There are multiple ways to track synchronization failures in Integration Cloud Service. You can track failures by:

- Errors Page
- Tracking Page
- Using Monitoring Dashboards
- Setting Email notifications

Note: For complete information on tracking failures in Integration Cloud Service, see Managing Errors in Getting Started with Integration Cloud Service guide.

Tracking Failures Using Errors Page

You can display errors by integration name, instance identifier, error location, or the time of occurrence over a specific time period. This provides you with a more granular view of integration failure details.

2. Click Errors.
3. For each instance, you can view details like integration name, identifier, and so on.
4. Click View Error icon to view the complete error.
5. Click Primary Identifier to access a graphical view of the integration instance.

Tracking Failures Using Error Tracking Page

You can use Tracking page to check for any instance of failure or success. This page lists all the instances. For each instance you can view details like identifiers, integration name, and so on. To check the status of a particular instance:

1. Navigate to Monitoring > Tracking from the Navigation Menu.
2. Identify the instance using the identifier names and check the status of the instance.
3. Click the Primary Identifier to drill down into the instance.
4. You can view all the integration steps here. Click Actions to view errors, activity stream, audit trail, and other details.

Tracking Failures Using Monitoring Dashboards

Use the dashboards to find total number of failures or view Activity Stream.
You can view monitoring dashboards by navigating to **Monitoring Dashboards** from the Navigation Menu.

You can drill down into the `<strong>Failures</strong>` to get the complete list of errors or click **Activity Stream** to view all the messages that are logged for each failure or success. For each message in the activity stream you can view details like integration name, integration step, and instance ID.

### Using Email Notifications for Synchronization Errors

You can notify users by email with hourly or daily reports about total messages received, total messages processed, successful messages, failed messages, and successful message rate with service failure alerts or with detailed report every five minutes when an integration failure occurs. For more information about configuring these email notifications, see Sending Service Failure Alerts, System Status Reports, and Integration Error Reports by Notification Emails in the Getting Started with Oracle Integration Cloud Service guide. If notification mail is configured for failures, an email is sent to all the users that are configured. The email contains the instance details for the time period configured. You can also track failed instances using these email notifications.

### Resolving Synchronization Errors

After you track the failure using one of these methods, you can resolve the failure based on the cause of the failure. If the failure is due to application error, correct the data and try to synchronize again. If the failure is due to one of the integrating applications being down, you can try synchronization again after all the applications are up.

**Related Topics**

- Sending Service Failure Alerts, System Status Reports, and Integration Error Reports by Notification Emails.
- Managing Errors in Integration Cloud Service

### Tracing and Managing Synchronization Errors in Oracle AIA

Use this topic to understand how you can trace synchronization errors in Oracle Application Integration Architecture.

In case of partner link errors, for remote systems such as Oracle Sales Cloud or Siebel UCM, it is logged as a remote business fault and thrown by a throw activity. You can check these faults from the Oracle AIA EM console, as a remote fault for the instance.

In case of non-partner link errors such as **xls xform** error, the **AIAsyncErrorHandlingBPELprocess** is called from the catch block. You can view the instance in **AIAsyncErrorHandlingBPELprocess** composite.

**Note:**

- For further understanding of different fault types in Oracle AIA, see Introduction to Oracle AIA Error Handling in Fusion Middleware Infrastructure Components and Utilities User’s Guide for Oracle Application Integration Architecture Foundation Pack.
- Oracle AIA offers pre-built error handling capabilities, however you may require additional configuration to use it effectively. For detailed configuration of error handling in Oracle AIA, see Setting Up Error Handling in Oracle AIA, in For detailed understanding of different fault types in Oracle AIA, see Introduction to Oracle AIA Error Handling in Fusion Middleware Infrastructure Components and Utilities User’s Guide for Oracle Application Integration Architecture Foundation Pack.

You can either use the pre-configured email notifications or EM console to trace synchronization errors.
Using Email Notifications for Tracing Synchronization Errors

You can configure Oracle AIA, to receive email notifications for synchronization failures. It is also possible to modify the contents of the email notifications so that you can receive complete error message as a part of the email. For more information on setting up email notifications see the topic Setting Up Error Notifications in AIA listed in the Related Topics section in this topic.

Email notifications, contains the details of the AIA composite that failed, failure error message, URL to the instance in the EM, and other details. Use the following instructions to trace faults using email notifications:

- Use the failure error message sent in the email to identify the issue and take the corrective action. Click the instance URL sent in the email notification. This opens the instance in the EM. This displays all the faults for that instance.
- Click Fault Location column to display detailed trace of that fault. In the Trace region you can view all the components and steps in AIA.
- Click the component to see details about the steps executed and check for any issue.

Using Oracle AIA EM Console for Tracing Synchronization Errors

You can find the details of AIA composite that failed, failure error message, and link to the instance in EM console along with other details, from the email notifications that you receive. Use the error message mentioned in the email to identify the issue and take corrective action. Use the following instructions to check for error in instance in AIA EM Console:

1. Sign into the AIA EM console (http://<host>::<port>/em).
2. Navigate to soa-infra. Use one of the following ways to find faulty instances:
   - On the Dashboard tab, check Faulted Instances column in Deployed Composites section.
   - On the Deployed Composites tab and check Faulted Instance column.
   - On the Instance tab, search using instance Name, using format %jane%. You can also use other filters such as Instance ID, Start Time From, and so on.
   - On the Faults and Rejected Messages tab, search using Composite Instance ID, Composite Name, Error Message Contains, and so on. You can find Composite Name from the error message notification email. If the email notification text is modified to send error message text, you can also use part of the error message text, in Error Message Contains field.
   - Click the composite name in soa-infra to view dashboard and faults for that composite.
3. On the faults list, click Error Message column to display complete error message.
4. Once you have filtered out the faults using one of the methods, view detailed Trace of the fault, click Instance ID column on the Instances tab or Composite Instance ID column on the Faults and Rejected Messages tab.
5. In the Trace region you can view all the components and steps in AIA. Click the component to see details about the steps executed and check for any issue.

After you track the failure using one of these methods, you can resolve the failure based on the cause of the failure. If the failure is due to application error, correct the data and try to synchronize again. If the failure is due to one of the integrating applications being down, you can try synchronization again after all the applications are up.

Related Topics

- Introduction to Oracle AIA Error Handling
- Setting Up Error Handling in Oracle AIA
- Customizing Error Notification Emails in Oracle AIA
• Using Error Notifications in Oracle AIA

• Setting Up Error Notifications in AIA
### 10 Performing Bulk Import and Export of Employee Data

#### Importing and Exporting Employee Data for Integration

This topic provides an overview of the import tasks you must perform for integrating Oracle Sales Cloud with Siebel UCM and Oracle Banking Platform.

Employee data from all the three applications: Oracle Banking Platform, Siebel UCM, and Oracle Sales Cloud must be in synchronization. To achieve this, you must import the data from one application to another wherever applicable.

To import data to different applications:

1. Export data from Oracle Sales Cloud, Oracle Banking Platform, and Siebel UCM.
2. Create appropriate enterprise structure in all the applications.
3. Import data from other applications using templates provided in the collateral .zip file.
4. Perform post import configurations, if any.
5. Export Cross References for resources and business units.
6. Import Cross References in Oracle Sales Cloud and Siebel UCM.

### Related Topics
- Understanding Terminologies Across Applications

#### Importing and Exporting Employee Data in Siebel UCM

### Exporting Employee Data from Siebel UCM

Before you export data from Siebel UCM, you must import system preferences from the och_system_preference_parameters.csv file with the collateral and import and compile required objects from EmployeeImportExport.sif.

#### Importing System Preferences

To import system preferences.

1. Sign in to Siebel UCM with a user that has administrative privileges.
2. Select Site Map from the Navigator.
3. On the next page, click Administration - Application > System Preferences.
4. Click Menu > Import....
5. Click Browse, select och_system_preference_parameters.csv file that is provided with the collateral.

**Note:** You must update the .csv file according to your environment settings before using it.

6. Select Auto Mapping as Input Source and Overwrite Existing Record as Conflict Resolution, and click Next.
7. Ensure the correct field mappings and click Next.
8. Query Oracle Sales Cloud records on the Administration - Application > System Preferences screen to verify that the data has been updated.

Exporting Employee

To export employee data:

1. Sign in to Siebel UCM with a user that has administrative privileges.
2. From the Navigator menu, select Site Map.
3. On the next page, click Administration - User > Employees.
4. Click Menu > Export...
5. On the Export dialog box, select the following:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rows to Export</td>
<td>All Rows In Current Query</td>
</tr>
<tr>
<td>Columns to Export</td>
<td>Visible Columns</td>
</tr>
<tr>
<td>Output Format</td>
<td>Tab Delimited Text File</td>
</tr>
</tbody>
</table>

6. Click Next and save the output .csv file.

Use this exported .csv file to fill in OCH_EMPLOYEES worksheet of the template file employees_template_for_import_into_osc.xlsm.

Exporting Organization Data

You must associate organizations to employees that are to be imported to Oracle Sales Cloud and Oracle Banking Platform. To export organization details of employees, use the following query against Siebel UCM database and export the results to a .csv file.

```sql
select U.login,
       LISTAGG(P.Name, ',') WITHIN GROUP (ORDER BY P.Name) AS OrganizationName
from S_CONTACT_BU E,
     S_party P,
     S_USER U
where E.BU_ID=P.ROW_ID
  and E.CONTACT_ID=U.ROW_ID
GROUP BY U.login;
```

This lists all the organizations that the employees have access to and helps in creating Resource Organization structure in Oracle Sales Cloud. Use this information to fill in OCH_EMP_ORG_ASSOCIATION worksheet of template file employees_template_for_import_into_osc.xlsm.

To export organization details from Siebel UCM user interface.

1. Sign in to Siebel UCM with a user that has administrative privileges.
2. From the Navigator menu, select Site Map.
3. On the next page, click Administration - Group > Organization.
4. Click Menu > Export...
5. On the Export dialog box, select the following:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rows to Export</td>
<td>All Rows In Current Query</td>
</tr>
</tbody>
</table>
### Field | Value
---|---
Columns to Export | Visible Columns
Output Format | Tab Delimited Text File

6. Click **Next** and save the output .csv file.

Use the exported .csv file to fill in `OCH_ORGANIZATIONS` worksheet of template `employees_template_for_import_into_och.xlsm`.

**Related Topics**
- Understanding Terminologies Across Applications

### Creating Enterprise Structure in Siebel UCM: Highlights

You must define your enterprise structure which includes defining organizations, divisions, positions, and responsibilities before you create or import employee records in Siebel UCM. Use the information provided in the following section to define the necessary enterprise structure.

**Compiling Required Objects**
- Before you create or import employee data, you must import and compile required objects using `employee_import_export.sif` file. See Compiling Objects to Enable Import and Export of Employees from Related Topics section.

**Creating Organizations**
- Each Marketing Entity and Business Unit in Oracle Banking Platform and Business Unit in Oracle Sales Cloud must have a corresponding Organization created in Siebel UCM. You cannot delete an organization once the record is created and saved. See Setting Up Organizations and Organization Skills in Related Topics section.

**Creating Divisions**
- Divisions are used to record addresses and maintain default currencies. A division is automatically created for each organization with the same name as that of the organization. Any additional divisions for the organizations need to be created using following instructions. Divisions are not considered for import to Oracle Sales Cloud and Oracle Banking Platform. See Setting Up Divisions in Related Topics section.

**Creating Positions**
- Positions determine which records users with a particular position can access. You must create a position for the organization that you have created. An employee must be assigned a position to be able to login to Siebel UCM. See Setting Up Positions and Position Skills in Related Topics section.

**Defining Responsibilities**
- Responsibilities determine which views users can access. Define responsibilities that correspond to the major job functions in your organization, either by creating a new responsibility or copying an existing responsibility. See Defining Responsibilities in Related Topics section.

**Related Topics**
- Defining Responsibilities
Creating Employees in Siebel UCM: Procedure

You must create an enterprise structure before creating an employee. This involves creating organizations and defining positions and responsibilities for the organizations. An organization in Siebel UCM is mapped to Marketing Entity and Business Unit in Oracle Banking Platform and Business Unit in Oracle Sales Cloud.

Setting Up Database Users

Before you can set up an application user in Siebel UCM, ensure that the user has a database account user name. Add the database accounts to the appropriate database and add these accounts to the group `SSE_ROLE`. Database authentication is one of the several possible methods of CRM authentication. To create database users:

1. Connect to database as a user with necessary privileges.
2. Modify the following example script according to your requirements to create new users and commit the transaction.

   ```sql
   CREATE USER <user> IDENTIFIED BY <password> DEFAULT TABLESPACE <default_tablespace_name> TEMPORARY TABLESPACE <temporary_tablespace_name>;
   GRANT "CONNECT" TO <user>;
   GRANT "SSE_ROLE" TO <user>;
   ALTER USER <user> DEFAULT ROLE "SSE_ROLE";
   COMMIT;
   ```

   **Caution:** The steps for adding users and placing them in this role group depend on the database software your organization is using. Work with your database administrator to set up developers as database users in the `SSE_ROLE` group.

Creating Employees

You must create an employee in Siebel UCM for each resource in Oracle Sales Cloud or user in Oracle Banking Platform.

For more information, see Setting Up an Employee in the related links section.

Importing Employees Data to Siebel UCM: Points to Consider

This topic provides information and steps to import employee data from Oracle Sales Cloud and Oracle Banking Platform to Siebel UCM.

Prerequisites

**Note:** For steps to import data to Siebel UCM, see Importing Data to Siebel UCM: Procedure.

Ensure that you have met the following prerequisite criteria before importing employee data to Siebel UCM.

1. You must create an enterprise structure before importing employee data. This involves creating organizations and defining positions and responsibilities for the organizations.
2. Employee work phone number must have minimum 10 digits.

3. When employees with multiple positions are imported to Siebel UCM, the primary position must be mentioned first, to enable mapping of Primary Organization ID mapping.

4. To import an employee who is associated to multiple organization, you must have multiple rows for the employees with differing values for Organization column and other fields such as Position, Responsibility, and so on.

5. Provide the ID of the organization which is required to be primary organization for the employee in the Primary Organization ID column.

6. Update the Primary Organization of an employee after the import process.
   a. Sign in to Siebel UCM with a user that has administrative privileges.
   b. From the Navigator menu, select Site Map.
   c. On the next page, click Administration - User > Employees.
   d. Update the Organization field in the look up dialog box by setting it as Primary.

7. Create a List of Values entry for MR_MS field in Siebel UCM. This List of Values entry must contain all the values that exist in Oracle Sales Cloud and Oracle Banking Platform to avoid any failed or partial import of data. To create MR_MS List of Values item:
   a. Sign in to Siebel UCM with a user that has administrative privileges.
   b. From the Navigator menu, select Site Map.
   c. On the next page, click Administration - Data > List of Values.
   d. Click New and enter MR_MS as Type.
   e. Enter appropriate values for Display Value and Language-Independent Code.
   f. Click Clear Cache to clear List of Values cache.

## Importing Data from Oracle Sales Cloud

Ensure that you have met the following prerequisite criteria before you import ensure the following prerequisites.

- Export both Employee Resource and Person(for Title field) from Oracle Sales Cloud and also Organizations(for Primary Organization ID column) from Siebel UCM.
- Ensure that enterprise structure is properly set up in Siebel UCM. For more information, see Creating Enterprise Structure in Siebel UCM: Highlights.
- All business units in Oracle Sales Cloud have been set up properly and can be found in Manage Business Unit task.

Refer to the Understanding Column Mappings section to understand the column mappings between Siebel UCM and Oracle Sales Cloud.

Use template file employees_template_for_import_into_och.xlsm to consolidate the export data. Fill in the worksheets listed in the following table and click Generate Data to get consolidated data for import. The resulting worksheet OSC_EMPLOYEES_CONSOLIDATED will have consolidated data.
Performing Bulk Import and Export of Employee Data

### Worksheet Name | Descriptions
--- | ---
OSCEmployees | This worksheet contains employees exported from Oracle Sales Cloud through Scheduled Export Process task for Employee Resource Object. Copy the contents of the exported .csv file into this worksheet. The employee details like First Name, Last Name, User name, and so on are obtained from this worksheet.

OSCPersons | This worksheet contains employees exported from Oracle Sales Cloud through Scheduled Export Process task for Person Object. Copy the contents of the exported .csv file into this worksheet. Title field value is obtained from this worksheet and used to find corresponding MR_MS field value of Siebel UCM from OCH OSC TITLE_LOOKUPS worksheet.

OCH OSC TITLE_LOOKUPS | This worksheet contains the lookups for Title field between Oracle Sales Cloud and Siebel UCM. You can create lookups if required and update this worksheet.

OCH ORGANIZATIONS | Organization details exported from Customer Hub UI. This is necessary to fill Primary Organization Id field needed for import in Siebel UCM. The Primary Organization ID is obtained from this worksheet. This value is by default set with Organization value of the employee and can be edited as required to change the primary organization of the employee.

OSC RESORG BU ASSOCIATIONS | This worksheet contains Resource Organization Business Units association details exported from Oracle Sales Cloud database. It lists all the business units a resource organization is associated with.

### Importing Data from Oracle Banking Platform

Before you import employee data from Oracle Banking Platform to Siebel UCM, ensure the following.

- Export employee details from Oracle Internet Directory as .ldif file.
- Ensure that enterprise structure is set up in Siebel UCM. For more information, see Creating Enterprise Structure in Siebel UCM: Highlights.
- All Marketing Entities and Business Units in Oracle Banking Platform can be found in ME991 screen for Market Entity - Business Unit Definition.

Refer to the Understanding Column Mappings section to understand the column mappings between Siebel UCM and Oracle Banking Platform.

### Understanding Column Mappings

<table>
<thead>
<tr>
<th>Column names in Siebel UCM</th>
<th>Column names from the file exported from Oracle Sales Cloud</th>
<th>Column names from the file exported from Oracle Banking Platform</th>
<th>Additional Information</th>
</tr>
</thead>
</table>
| Organization | BusinessUnit | businessunit | • This column should list organizations required to be associated with an employee.  
• accessiblebusinessunits field in Oracle Banking Platform lists all the business units the user has access to. Associate all the business units listed in this column to the |
### Column names in Siebel UCM

<table>
<thead>
<tr>
<th>Column names in Siebel UCM</th>
<th>Column names from the file exported from Oracle Sales Cloud</th>
<th>Column names from the file exported from Oracle Banking Platform</th>
<th>Additional Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>NA</td>
<td>LegalEntity</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>First Name</td>
<td>First Name</td>
<td>givenname</td>
<td>NA</td>
</tr>
<tr>
<td>Last Name</td>
<td>Last Name</td>
<td>sn</td>
<td>NA</td>
</tr>
<tr>
<td>Email addr</td>
<td>Primary EMail</td>
<td>mail</td>
<td>NA</td>
</tr>
<tr>
<td>NA</td>
<td>Party ID</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Work Phone #</td>
<td>Primary Phone</td>
<td>telephonenumber</td>
<td>NA</td>
</tr>
<tr>
<td>User ID</td>
<td>User Name</td>
<td>uid</td>
<td>NA</td>
</tr>
<tr>
<td>Responsibility</td>
<td>RoleCode</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>NA</td>
<td>OrganizationName</td>
<td>NA</td>
<td>Identify all the business units associated with this resource organization in Oracle Sales Cloud. Import records with these as organizations in Siebel UCM.</td>
</tr>
<tr>
<td>NA</td>
<td>Primary Organization Id</td>
<td>NA</td>
<td>This information is obtained from export of organizations from Siebel UCM. For more information, see Exporting Data from Siebel UCM.</td>
</tr>
<tr>
<td>Mr/Ms</td>
<td>Title</td>
<td>title</td>
<td>You must change or create title values in Oracle Sales Cloud or Oracle Banking Platform to match the corresponding values in Siebel UCM.</td>
</tr>
<tr>
<td>NA</td>
<td>Business Units Associated</td>
<td>NA</td>
<td>This column contains the list of all the business units in Oracle Sales Cloud associated with the employee. This includes both business units and market entities that are created as</td>
</tr>
</tbody>
</table>
Performing Bulk Import and Export of Employee Data

<table>
<thead>
<tr>
<th>Column names in Siebel UCM</th>
<th>Column names from the file exported from Oracle Sales Cloud</th>
<th>Column names from the file exported from Oracle Banking Platform</th>
<th>Additional Information</th>
</tr>
</thead>
</table>

- business units. Associate the employees in Siebel UCM, with all the organizations, corresponding to these business units, but not with the organizations corresponding to marketing entities.

- Find this information from the Oracle Sales Cloud Data export. For more information, see Exporting Data from Oracle Sales Cloud.

**Related Topics**

- Understanding Terminologies Across Applications

### Importing Employees Data to Siebel UCM: Procedure

Perform the following steps in Siebel UCM to import employees from other applications.

1. Sign in to Siebel UCM with a user that has administrative privileges.
2. From the **Navigator** menu, select **Site Map**.
3. On the next page, click **Administration - User > Employees**.
4. Click **Menu > Import...**.
5. On the next dialog box, click **Browse**, select and open the file to be imported.
6. Select **Auto mapping** as **Input Source**, **Overwrite Existing Record** as **Conflict Resolution**, and click **Next**.
7. On the next page, update the column name mapping if required and click **Next**. The import result is displayed.
8. After the file has been imported, create database user login for these users.

**Note:** For more information on creating database users, see Setting Up Database Users section in Creating Employees in Siebel UCM.

**Related Topics**

- Understanding Terminologies Across Applications

### Importing and Exporting Employees Data in Oracle Sales Cloud
Exporting Employees Data from Oracle Sales Cloud

This topic explains the procedure to export resources and business units from Oracle Sales Cloud.

Exporting Resources

To export employee data from Oracle Sales Cloud:

1. Sign in to Oracle Sales Cloud using administrator role.
2. In the Setup and Maintenance work area, go to the following:
   - Offering: Service
   - Functional Area: Data Import and Export
   - Task: Schedule Export Process
3. On the Overview page, click Create icon.
4. On the Enter Basic Information step, provide appropriate Name and click Next.
5. In the Export Objects region, on the Configure Export Objects step, click Create icon.
6. On the Manage Export Objects dialog box, select Employee Resource and Person, and click Done.

Caution: Employee Resource and Person are used to fill worksheets of the template employees_template_for_Import_into_och.xlsm.
- Keep only the columns mentioned in OSC_EMPLOYEES worksheet of the template for the Employee Resource object.
- Keep only the Party ID and Title fields for export, while exporting Person object.

7. On the Configure Export Objects step click Next.
8. On the Create Schedule step, click Next.
10. On the Overview page, click Refresh icon until you verify the export process is done.

Exporting Business Units

You must associate business units with employees that are to be imported to Siebel UCM and to Oracle Banking Platform. To export the list of business units associated with Resource Organizations, use the following query against Sales Cloud Database and export the results to a .csv file. This information should be filled in OSC_RESORG_BU_ASSOCIATIONS worksheet of the template employees_template_for_Import_into_och.xlsm.

```sql
select B.NAME AS RESOURCE_ORGANIZATION, 
   LISTAGG(C.Name, ',') WITHIN GROUP (ORDER BY C.Name) AS BUSINESS_UNITS 
from JTF_RS_RESOURCE_ORG_BU A, 
   hr_all_organization_units B, 
   hr_all_organization_units C 
where A.ORGANIZATION_ID=B.ORGANIZATION_ID 
  and A.BUSINESS_UNIT_ID=C.ORGANIZATION_ID 
GROUP BY B.NAME;
```

Related Topics

- Understanding Terminologies Across Applications
Creating Legal Entities and Business Units in Oracle Sales Cloud

Use this topic to know how to setup legal entities and business units for each marketing entity and business unit in Oracle Banking Platform or an organization in Siebel UCM.

Creating Legal Entities

You must setup enterprise and legal entity as a part of initial customer setup.

A legal entity is a recognized party with rights and responsibilities given by legislation. Legal entities must comply with the regulations of jurisdictions, in which they register. For example, Europe now allows for companies to register in one member country and do business in all member countries, and the US allows for companies to register in one state and do business in all states.

To support local reporting requirements, legal reporting units are created and registered. A legal entity can represent all or part of your enterprise’s management framework. For example, if you operate in a large country such as the United Kingdom or Germany, you might incorporate each division in the country as a separate legal entity. In a smaller country, for example Austria, you might use a single legal entity to host all of your business operations across divisions.

A legal employer is a legal entity that employs workers. Payroll statutory units are legal entities that are responsible for paying workers, including the payment of payroll tax and social insurance. A payroll statutory unit can pay and report on payroll tax and social insurance on behalf of one or many legal entities, depending on the structure of your enterprise.

For more information about creating legal entities, refer to Implementing Sales guide.

Creating Business Units

A business unit is a unit of an enterprise that performs one or many business functions that can be rolled up in a management hierarchy. A business unit can process transactions on behalf of many legal entities.

Use a business unit as a securing mechanism for transactions. For example, if you run your export business separately from your domestic sales business, then secure the export business data to prevent access by the domestic sales employees. To accomplish this security, set up the export business and domestic sales business as two separate business units.

You must create one business unit for each marketing entity and business unit in Oracle Banking Platform or an organization Siebel UCM.

Note: In Oracle Sales Cloud, there is no hierarchy defined between Marketing Entity and its Business Units in Oracle Banking Platform. When employees are created and the business unit is added to the resource organization, the parent business unit (market entity in Oracle Banking Platform) associated with the business unit is also added to maintain the relationship.

To create business units in Oracle Sales Cloud:

1. Sign in to Oracle Sales Cloud using administrator role.
2. In the Setup and Maintenance work area, go to the following:
   - Offering: Service
   - Functional Area: Accounts and Contacts
   - Task: Manage Business Unit
3. On the Manage Business Units page, in Search Results region, click Create icon.
4. On the Create Business Unit page, specify the following and select Active check box.
   - Name: Valid Name for the Business Unit
   - Default Set: COMMON
   - Select Active check box.

5. Click Save and Close.

Related Topics
- Understanding Terminologies Across Applications
- Sales Cloud Getting Started with Your Implementation

Setting Up Resources in Oracle Sales Cloud
Use this topic to understand how to set up resource organizations in Oracle Sales Cloud after the data import.

Note: To understand how different entities are mapped for Oracle Sales Cloud, Siebel UCM, and Oracle Banking Platform, see Understanding Entity Mappings.

Creating Resource Organizations
Resource Organizations are organizations whose members are resources. Resource Organizations are used to implement Sales Organization, Partner Organizations, and so on.

Note: Create resource organization manually, only if you are not importing data from Siebel UCM and Oracle Banking Platform.

To create resources:
1. Sign in to Oracle Sales Cloud using Administrator role.
2. In the Setup and Maintenance work area, go to the following:
   - Offering: Service
   - Functional Area: Users and Security
   - Task: Manage Internal Resource Organizations
3. On the Manage Internal Resource Organizations page, in the Search Results region, click Create icon.
5. Click Next.
6. On Create Organization: Enter Basic Information step, enter a valid organization name.
7. On the Organization Usages region, click Add Row icon.
8. Select Sales Organization from the Usage list.
9. Click Finish.

Adding Resource Organizations in Root Organization
Add the imported resource organizations that you created in the root organization using following steps:
1. Sign in to Oracle Sales Cloud using Administrator role.
3. Click View Organization Hierarchies from Tasks > Resource Organizations.
4. On the View Organizations Hierarchies page, in the Search using the default search options.
5. In the Search Results, select Internal Resource Organization Hierarchy.
6. Click Actions > Edit.
8. Click Create icon.
9. On the Search Node dialog box, search and select each of newly created organizations to be added and click OK.
10. Click Save and Close.
11. Click Done.

Adding Business Units to Resource Organizations

After the resources have been imported, add required business units to the resource organization manually in the resource directory.

Perform the following procedure for all the resource organizations that you have created.

1. Sign in to Oracle Sales Cloud using Administrator role.
3. In Search region, select Organizations from the Search drop-down list, enter the organization and click Search icon.
4. Click the organization from the Search Results.
5. Navigate to Business Units tab.
6. Click Add Row icon and select required business unit from the Business Unit drop-down list.
7. Click Save and Close.

Creating Resource Roles

To setup resource roles in Oracle Sales Cloud:

1. Sign in to Oracle Sales Cloud using Administrator role.
2. In the Setup and Maintenance work area, go to the following:
   - Offering: Service
   - Functional Area: Users and Security
   - Task: Manage Resource Roles
3. On the Manage Resource Roles page, click Create icon in the Search Results region.
4. On the Create Role page, specify the following:
   - Role Name: Enter a new custom Integration role name in Oracle Sales Cloud.
   - Role Code: Enter responsibility value from Siebel application as a role code.
   - Role Type: Sales
     For example, Siebel Responsibility of Channel Executive could be a Role Code, with Role Name as Siebel Integration Channel Executive in Oracle Sales Cloud.
5. Ensure that only the Manage check box is selected in the Create Role region for a manager role and Member is selected for a sales representative role.
6. Click Save and Close.

Creating Role Mappings

Roles give users access to data and functions. To provision a role to users, you define a relationship, called a role mapping, between the role and some conditions.
Use the following procedure to create role mapping.

1. Sign in to Oracle Sales Cloud using Administrator role.
2. In the Setup and Maintenance work area, go to the following:
   - Offering: Service
   - Functional Area: Users and Security
   - Task: Manage HCM Role Provisioning Rules
3. On the Manage Role Mappings page, click Create icon in Search Results region.
4. On the Create Role Mappings page, specify Mapping Name, which will be used to identify the mapping.
5. In the Conditions region:
   a. Select the resource role that you want to provision from the Resource Role drop-down list.
   b. Select active from the HR Assignment Status drop-down list. This ensures that the provisioned roles are automatically removed if the user is terminated in Global Human Resources.
6. In the Associated Roles region, click Add Row icon to add the job role to be provisioned.
   a. In the Role Name column, search for the job roles and select the role.
   b. Select Autoprovision check box for all the roles.
7. Click Save and Close.

**Related Topics**
- Resource Organizations and Organization Usage: Explained
- Primary Resource Organization by Usage: Explained
- Why am I unable to view the resource hierarchy for my organization?
- Autoprovisioning: Explained
- Role Mappings: Explained

**Importing Employees Data to Oracle Sales Cloud: Points to Consider**

This topic explains the points that you must consider before you begin your import of data to Oracle Sales Cloud.

**Prerequisites**

The following prerequisites must be completed before importing data to Oracle Sales Cloud.

1. Setup necessary Legal Entities and Business Units according to Organizations in Siebel UCM or Marketing Entities and Business Units in Oracle Banking Platform.
2. To create required titles to match the Title field in Siebel UCM.
   a. Sign in to Oracle Sales Cloud using Administrator role.
b. In the Setup and Maintenance work area, go to the following:
   - Functional Area: Application Extensions
   - Task: Manage Common Lookups

c. Search for Lookup Type TITLE.

d. Add Lookup Codes for the required titles in Siebel UCM and Oracle Banking Platform.

e. Click Save and Close.

3. Setup necessary jobs in Oracle Sales Cloud to map Job Title field in Siebel UCM. Jobs and positions represent roles that enable you to distinguish between tasks and the individuals who perform those tasks.

Use the following procedure to create jobs:

   a. Sign in to Oracle Sales Cloud using Administrator role.
   b. In the Setup and Maintenance work area, go to the following:
       - Functional Area: Application Extensions
       - Task: Manage Jobs
   c. On the Manage Jobs page, click Create.
   d. On the Basic Details step, enter the Name and Code for the job and click Next.
   e. On the Details step, ensure that Active is selected in the Status drop-down list.
   f. Click Submit.

4. Define resource roles for Oracle Sales Cloud. For more information, see Setting Up Resources in Oracle Sales Cloud.

5. Setup roles and users.

6. Autoprovision job roles to sales users.

For more information, see Setting Up Resources in Oracle Sales Cloud.

✏️ Note: For steps to import data to Siebel UCM, see Importing Employees in Oracle Sales Cloud: Procedure.

Importing Data from Siebel UCM

Before you import data from Siebel UCM, ensure the following.

- Export the employees from Siebel UCM.
- Create all the business units in Oracle Sales Cloud to map organizations in Siebel UCM.
- Ensure that required role codes are created. Autoprovision the job roles for Siebel Responsibilities with role code as Siebel Responsibility Value.
- Create required Jobs for Job Title and Titles for MR MS field in Siebel UCM.
- The primary organization of the employee is Siebel UCM is mapped to Business Unit of the resource in Oracle Sales Cloud. In addition to this, the business units corresponding to all the organizations of the employee in Siebel UCM are added to the resource organization of the resource in the Oracle Sales Cloud.
Note: Resource Organization structure in Oracle Sales Cloud depends on the combination of organizations that resources need access to. A resource organization is created for each set of business units or marketing entity that are staffed by a group of people in Oracle Banking Platform. For example, consider an organization structure with two business units in Oracle Sales Cloud: Unit1 and Unit2. So there could be a maximum of three resource organizations:

- **Unit1_Org**: For all the users who only require Unit1
- **Unit2_Org**: For all the users who only require Unit2
- **Unit1_Unit2_Org**: For all the users who require both Unit1 and Unit2

Refer to the Understanding Column Mappings section understand the column mappings between Oracle Sales Cloud and Siebel UCM.

Use template file `employees_template_for_import_into_osc.xlsm` to consolidate the export data. Fill in the worksheets listed in the following table and click Generate Data to get consolidated data for import. The resulting worksheet `OCH_EMPLOYEES_CONSOLIDATED` will have consolidated data.

<table>
<thead>
<tr>
<th>Worksheet Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OCH_EMPLOYEES</td>
<td>This worksheet contains employees exported from Siebel UCM. Copy the contents of the exported .csv file into this worksheet. The employee details like First Name, Last Name, User name, and so on are obtained from this worksheet.</td>
</tr>
<tr>
<td>OCH_EMP_ORG_ASSOCIATIONS</td>
<td>This worksheet contains employee organization association details exported from Siebel UCM. Copy the contents of the exported .csv file into this worksheet. The list of organizations associated with each employee is obtained from this worksheet, which can be used to structure the Resource Organization.</td>
</tr>
<tr>
<td>OCH_OSC_TITLE_LOOKUPS</td>
<td>This worksheet contains the lookups for Title field between Oracle Sales Cloud and Siebel UCM. Create lookups if required, and update the worksheet.</td>
</tr>
</tbody>
</table>

### Importing Data from Oracle Banking Platform

Before you import data from Oracle Banking Platform, ensure the following.

- Export the employees from Oracle Banking Platform as `.ldif` file.
- Resource Organization structure in Oracle Sales Cloud depend on the combination of business units that employees need access to, obtained from `accessibleBusinessUnits` field of Oracle Banking Platform.
- Create a transaction branch field in resource pages corresponding to the home branch in Oracle Banking Platform.

Refer to the Understanding Column Mappings section to understand the column mappings between Oracle Sales Cloud and Oracle Banking Platform.

### Understanding Column Mappings

<table>
<thead>
<tr>
<th>Column Names from the file exported from Siebel UCM</th>
<th>Column Names from the file exported from Oracle Banking Platform</th>
<th>Column names in Oracle Sales Cloud Import file</th>
<th>Additional Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Last Name</td>
<td>sn</td>
<td>Last Name</td>
<td>NA</td>
</tr>
<tr>
<td>Column Names from the file exported from Siebel UCM</td>
<td>Column Names from the file exported from Oracle Banking Platform</td>
<td>Column names in Oracle Sales Cloud Import file</td>
<td>Additional Information</td>
</tr>
<tr>
<td>-----------------------------------------------------</td>
<td>-----------------------------------------------------------------</td>
<td>---------------------------------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>First Name</td>
<td>givenname</td>
<td>First Name</td>
<td>NA</td>
</tr>
<tr>
<td>Job Title</td>
<td>NA</td>
<td>Job Title</td>
<td>Create necessary jobs before importing.</td>
</tr>
<tr>
<td>User ID</td>
<td>uid</td>
<td>User ID</td>
<td>NA</td>
</tr>
<tr>
<td>Responsibility</td>
<td>NA</td>
<td>RoleCode</td>
<td>Create resource roles and provide mapping.</td>
</tr>
<tr>
<td>Position</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Division</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Organization</td>
<td>businessunit</td>
<td>OSC BusinessUnit</td>
<td>NA</td>
</tr>
<tr>
<td>Employee Type</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Email Addr</td>
<td>mail</td>
<td>Email</td>
<td>NA</td>
</tr>
<tr>
<td>Cluster Name</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Integration User Flag</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Position Id</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Position Integration Id</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Employee</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Employee Id</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>OSC Tree Code</td>
<td>NA</td>
<td>OSC Tree Code</td>
<td>NA</td>
</tr>
<tr>
<td>OSC Parent Tree Code</td>
<td>NA</td>
<td>OSC Parent Tree Code</td>
<td>NA</td>
</tr>
<tr>
<td>OSC Organization Usage</td>
<td>NA</td>
<td>OSC Organization Usage</td>
<td>NA</td>
</tr>
<tr>
<td>OSC Legal Entity</td>
<td>Legal Entity of the businessunit</td>
<td>OSC Legal Entity</td>
<td>NA</td>
</tr>
<tr>
<td>OSC Person Type</td>
<td>NA</td>
<td>OSC Person Type</td>
<td>NA</td>
</tr>
<tr>
<td>OSC Create Login Account</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
</tbody>
</table>
## Importing Employees Data in Oracle Sales Cloud: Procedure

Use this topic to find out necessary procedure to import data in Oracle Sales Cloud from Siebel UCM and Oracle Banking Platform.

> **Note:** After the import is successful, you must add Business Units to Resource Organization for all Resource Organizations created using the file import process. For more information, See Setting Up Resources in Oracle Sales Cloud.

<table>
<thead>
<tr>
<th>Column Names from the file exported from Siebel UCM</th>
<th>Column Names from the file exported from Oracle Banking Platform</th>
<th>Column names in Oracle Sales Cloud Import file</th>
<th>Additional Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSC Role Type</td>
<td>NA</td>
<td>OSC Role Type</td>
<td>NA</td>
</tr>
<tr>
<td>Mr/Ms</td>
<td>NA</td>
<td>NA</td>
<td>Create any values if required and provide the corresponding Title Code for the Title in Siebel UCM.</td>
</tr>
<tr>
<td>Work Phone #</td>
<td>telephonenumber</td>
<td>Work Phone</td>
<td>NA</td>
</tr>
<tr>
<td>Organization Names</td>
<td>NA</td>
<td>NA</td>
<td>This is a list of all organizations associated with the employee in Siebel UCM, obtained from data exported from Siebel UCM. The business units corresponding to these organizations should be associated to the user's resource organization.</td>
</tr>
<tr>
<td>Title</td>
<td>title</td>
<td>Title</td>
<td>NA</td>
</tr>
<tr>
<td>Manager Email</td>
<td>NA</td>
<td>Manager Email</td>
<td>Provide the manager email address, depending on the Resource Org structure.</td>
</tr>
<tr>
<td>Resource Organization</td>
<td>NA</td>
<td>Resource Organization</td>
<td>Resource Organization structure in Oracle Sales Cloud depends on the combination of business units that employees need access to.</td>
</tr>
<tr>
<td>Parent Resource Organization</td>
<td>NA</td>
<td>Parent Resource Organization</td>
<td>Provide the parent organization name.</td>
</tr>
<tr>
<td>NA</td>
<td>homebranch</td>
<td>Transaction Branch</td>
<td>Code value of the homebranch in Oracle Banking Platform.</td>
</tr>
</tbody>
</table>

---

147
Importing Employees Using File-based Import

To import employees using file-based import:

1. Sign in to Oracle Sales Cloud using Administrator role.
2. In the Setup and Maintenance work area, go to the following:
   - Offering: Sales
   - Functional Area: Data Import and Export
   - Task: Manage File Import Activities
3. On Manage Import Activities page, click Create.
4. On the Enter Import Options step, perform the following and click Next.
   a. Enter a valid Name for the import.
   b. Select Employee Resource from Object drop-down list.
   c. Select Header row included check box.
   d. Select Desktop as Uploaded From.
   e. Click Browse and select .csv file to upload.
5. On the Map Fields page, use the following table to map the .csv file.
6. Click Next til you reach Review and Activate step.
7. Review the details and click Activate to submit the import immediately.
8. Add Business Units to Resource Organization for all the Resource Organizations created using the file import process.

**Note:** You must order employee records in the file such that, for any organization, the manager of the organization is imported first followed by his team members.

<table>
<thead>
<tr>
<th>Column Header (Source)</th>
<th>Ignore</th>
<th>Target Object (Target)</th>
<th>Target Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Last Name</td>
<td>N</td>
<td>WorkerProfile</td>
<td>LastName</td>
<td>Not applicable</td>
</tr>
<tr>
<td>First Name</td>
<td>N</td>
<td>WorkerProfile</td>
<td>FirstName</td>
<td>Not applicable</td>
</tr>
<tr>
<td>User ID</td>
<td>N</td>
<td>WorkerProfile</td>
<td>Username</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Resource Organization</td>
<td>N</td>
<td>ResourceOrganizationMembership</td>
<td>OrganizationName</td>
<td>A resource organization is created for each set of business units or marketing entity that are staffed by a group of people in Oracle Banking Platform. For example, consider an organization structure with two business units</td>
</tr>
</tbody>
</table>
### Performing Bulk Import and Export of Employee Data

<table>
<thead>
<tr>
<th>Column Header (Source)</th>
<th>Ignore</th>
<th>Target Object (Target)</th>
<th>Target Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parent Resource Organization</td>
<td>N</td>
<td>ResourceOrganizationMembership</td>
<td>ParentOrganizationName</td>
<td>All organizations are imported in a root organization. For example, Global Sales and Marketing. For manager users, provide the parent organization as root organization.</td>
</tr>
<tr>
<td>Email</td>
<td>N</td>
<td>WorkerProfile</td>
<td>EmailAddress</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Manager Email</td>
<td>N</td>
<td>WorkerProfile</td>
<td>ManagerEmailAddress</td>
<td>If a user is a manager, leave this blank.</td>
</tr>
<tr>
<td>OSC Business Unit</td>
<td>N</td>
<td>WorkerProfile</td>
<td>BusinessUnit</td>
<td>Not applicable</td>
</tr>
<tr>
<td>OSC Legal Entity</td>
<td>N</td>
<td>WorkerProfile</td>
<td>LegalEntity</td>
<td>Not applicable</td>
</tr>
<tr>
<td>OSC Organization Usage</td>
<td>N</td>
<td>ResourceOrganizationMembership</td>
<td>OrganizationUsage</td>
<td>SALESRESOURCE_ORG</td>
</tr>
<tr>
<td>Not applicable</td>
<td>Y</td>
<td>ResourceProfile</td>
<td>CreateUserAccount</td>
<td>Not applicable</td>
</tr>
<tr>
<td>OSC Tree Code</td>
<td>N</td>
<td>ResourceOrganizationMembership</td>
<td>TreeCode</td>
<td>GLOBALSALESMARKETING</td>
</tr>
<tr>
<td>OSC Parent Tree Code</td>
<td>N</td>
<td>ResourceOrganizationMembership</td>
<td>ParentTreeCode</td>
<td>GLOBALSALESMARKETING</td>
</tr>
<tr>
<td>OSC Person Type</td>
<td>N</td>
<td>WorkerProfile</td>
<td>PersonType</td>
<td>EMP</td>
</tr>
<tr>
<td>OSC Role Type</td>
<td>N</td>
<td>ResourceProfile</td>
<td>RoleTypeCode</td>
<td>SALES</td>
</tr>
<tr>
<td>Job Title</td>
<td>N</td>
<td>WorkerProfile</td>
<td>JobCodeName</td>
<td>Not applicable</td>
</tr>
</tbody>
</table>
### Column Header (Source) | Ignore | Target Object (Target) | Target Attribute | Description
--- | --- | --- | --- | ---
Mr/Ms | N | WorkerProfile | Title | 

**Note:** When Title value given in import is not present in Oracle Sales Cloud, the records fail with error: Attribute set with value `<CUSTOM_JOB_TITLE>` for Title in WorkerAM. PersonEOPersonName PersonVOToPersonNameVO_PersonNameVO failed.

### Work Phone | N | WorkerProfile | WorkPhoneNumber | Work Phone. The value can have special characters such as +, (,),-,x, space. Example: (408) 853-7000 x2202

### Transaction Branch | N | ResourceProfile | _ ORAFS__TransactionBranch_ c | The Home Branch code value of the employee in Oracle Banking Platform.

### Troubleshooting File Import

Use the information in the following table to resolve any error that occur during the file import operation.

<table>
<thead>
<tr>
<th>Error Message or Issue</th>
<th>Possible Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Invalid value for PERSON-Invalid value for PERSON_ID. Enter PERSON_ID value from HRC_LOADER_BATCH_LINES300000001237252</td>
<td>Ensure that Legal Entity is present in Oracle Sales Cloud.</td>
</tr>
<tr>
<td>Invalid value for PERSON-Invalid value for PERSON_ID. Enter PERSON_ID value from HRC_LOADER_BATCH_LINES300000001237252</td>
<td>Ensure that relevant Business Unit is present in Oracle Sales Cloud.</td>
</tr>
<tr>
<td>Invalid value for PERSON-Invalid value for PERSON_ID. Enter PERSON_ID value from HRC_LOADER_BATCH_LINES300000001237581</td>
<td>Ensure that required Job Title is defined in Oracle Sales Cloud.</td>
</tr>
<tr>
<td>Attribute set with value <code>&lt;CUSTOM_JOB_TITLE&gt;</code> for Title in WorkerAM. PersonEOPersonName PersonVOToPersonNameVO_PersonNameVO failed</td>
<td>Ensure that the Title value provided in the import file is present in Oracle Sales Cloud.</td>
</tr>
</tbody>
</table>
Related Topics
- Understanding Terminologies Across Applications
- File-Based Data Import Objects: Explained

Importing and Exporting Employees Data in Oracle Banking Platform

Exporting Employees Data from Oracle Banking Platform

The data from Oracle Banking Platform must be exported in the form of .ldif file.

Use the following procedure to export data from Oracle Banking Platform:

1. Sign in to Oracle Directory Services Manager (ODSM).
2. Click Connect to a directory and select required Oracle Internet Directory (OID).
3. Once you are connected to the OID, navigate to the Data Browser tab.
4. Expand dc=com > dc=oracle > dc=in > cn=Users.
5. Right-click and select Export LDIF.
6. On the Export File dialog, select Export Operational Attributes, if you must export them.
7. Click OK and save a copy of LDIF file on the Download LDIF File dialog box.
8. Click OK.

Related Topics
- Understanding Terminologies Across Applications

Creating Enterprise Structure in Oracle Banking Platform

You must setup enterprise structure in Oracle Banking Platform before you import your data.

Creating Legal Entities

The enterprise and legal entities in Oracle Banking Platform are created during the initial customer setup. Verify records from cs03 screen using Fast Path.

Creating Marketing Entities

For each business unit in Oracle Sales Cloud, and organization in Siebel UCM, you must create a corresponding marketing entity or business unit in Oracle Banking Platform. Perform the following steps to create a marketing entity in Oracle Banking Platform.

1. Sign in to Oracle Banking Platform.
2. Search for ME991 using Fast Path.
3. On the Marketing Entity - Business Unit Definition page, in the Marketing Entity region, enter the Marketing Entity Code and wait for the rest of the fields to auto-populate.
4. In the Business Unit region, click Add Business Unit.
5. Add Business Unit Code and Business Unit Name. Repeat this for each business unit in Oracle Sales Cloud.
Creating Home Branch
To create a home branch:

1. Sign in to Oracle Banking Platform.
2. Search for CS06 using Fast Path.
3. On the Branch Parameters page, enter the Branch Code. Branch code should be a numeric value.
4. Click Update.

Creating Users and Setting Up User Attributes and Roles in Oracle Banking Platform
Oracle Banking Platform uses Oracle Internet Directory (OID) as a specialized database that stores and retrieves collections of information about objects such as users.

Use Oracle Directory Services Manager (ODSM) as a graphical administrative interface for OID. You can launch ODSM directly from its own URL or from the Oracle Enterprise Manager Fusion Middleware Control pages for Oracle Internet Directory and Oracle Virtual Directory.

Before you proceed with the next steps, you must create a user login for Oracle Directory Services Manager (ODSM) which should be installed on the same server as Oracle Internet Directory (OID) on port 7005. For example, you can find ODSM at <IP address of OID.:7005/odsm>.

Creating a user for ODSM
To create a user for ODSM, perform the following steps.

1. Sign in to ODSM.
2. Click Connect to a directory and select required OID.
   If you cannot find required OID, create a new connection using steps provided in Creating New Connections section.
3. Once you are connected to the OID, navigate to the Data Browser tab.
4. Expand dc=com > dc=oracle > dc=in > cn=Users.
5. Right-click cn=Users and select Create.
6. On the Create New Entry dialog box, Entry Properties step, select top.person, fcPerson, and inetOrgPerson as Object Class.
7. In the Parent of the entry field, click Browse. Expand the selection and select cn=Users.
8. Click Next on the Create New Entry dialog box.
10. Select Relative distinguished Name value as uid and click Next.
11. Click Finish on the Status step. The user is now created.

Modifying User Attributes
Perform the following steps to modify user attributes.

1. Sign in to ODSM.
2. Click Connect to a directory and select required OID.
   If you cannot find required OID, create a new connection using steps provided in Creating New Connections section.
3. Once you are connected to the OID, navigate to the Data Browser tab.
4. Expand dc=com > dc=oracle > dc=in > cn=Users.
5. Expand **cn=Users** and select required user.
6. On the user page, navigate to **Attributes** tab.

Initially, only attributes that are not empty are visible. To add more attributes:

   a. Switch between Managed Attributes and Show All by using the Views drop-down list.
   b. Click Add icon in the Optional Attributes region to change the list of attributes that are shown as Managed Attributes.
   c. Select attributes to move from the All Attributes list to the Shown Attributes list. Use Move Move All icons if required.
   d. Click Add Attributes to apply your changes.
7. Click **Apply** to save your changes.

### Assigning Group to the User

To assign a group to the user that you just created, do the following:

1. Sign in to ODSM.
   Click **Connect to a directory** and select the OID.
   If you cannot find required OID, create a new connection using steps provided in Creating New Connections section.
2. Once you are connected to the OID, navigate to the **Data Browser** tab.
3. Expand **dc=com > dc=oracle > dc=in > cn=Groups**.
4. Expand **cn=Groups** and select the group in the data tree to which the user is to be added.
5. To add a member to the group, click **Plus** icon in the Members field.
6. Provide the Distinguished Name Path of the user and click **OK**.
7. Click **Apply** to save your changes.

### Creating New Connections

User following steps to create a new connection if you cannot find required OID in ODSM.

1. Sign in to ODSM.
2. Click **Connect to a directory** and select **Create a New Connection**.
3. On the New Connection dialog box, select **OID** as **Directory Group**.
4. Fill in the required details such as **Name**, **Server**, **Port**, **User Name**, and **Password**.
5. Click **Connect**.

### Importing Employees Data to Oracle Banking Platform: Points to Consider

Employees are known as Users in Oracle Banking Platform. They are imported in Oracle Internet Directory (OID) as LDAP Data Interchange Format (LDIF) files. We can use text files to import data into OID through Import LDIF option in ODSM.

Consider the following points while preparing your .ldif file for import:

1. Every user must have all the object classes given in the following table.
2. Each entry of the object class can be given in a new line in the .ldif file. For example:

```
objectclass: top
objectclass: person
objectclass: fcPerson
```
3. To import multiple users using a LDIF single file, separate each user details with a blank line.
4. Enter Code value for the fields `businessunit`, `homebranch`, `targetunit`, `accessiblebusinessunits`, and `accessibleTargetUnits`.
5. Each line in an LDIF file must be correctly formatted to be read by OID, with careful usage of white spaces and line breaks.

The following table lists different LDIF attributes, their details, and sample values.

<table>
<thead>
<tr>
<th>LDIF Attribute Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dn:</td>
<td>Distinguished Name</td>
</tr>
<tr>
<td>preferredlanguage:</td>
<td>Preferred Language</td>
</tr>
<tr>
<td>givenname:</td>
<td>User First Name</td>
</tr>
<tr>
<td>accessiblebusinessunits:</td>
<td>Codes of the Business units which the user can access, separated using comma.</td>
</tr>
<tr>
<td>objectclass:</td>
<td>Oracle Banking Platform specific field. Each Object class is entered as a new line in the .ldif File.</td>
</tr>
<tr>
<td>uid:</td>
<td>User ID</td>
</tr>
<tr>
<td>mail:</td>
<td>User Email</td>
</tr>
<tr>
<td>businessunit:</td>
<td>Code of the Parent Business unit.</td>
</tr>
<tr>
<td>cn:</td>
<td>User Common Name. Ensure that the value for cn and uid is the same.</td>
</tr>
<tr>
<td>telephonenumber:</td>
<td>Telephone Number</td>
</tr>
<tr>
<td>accessibleTargetUnits:</td>
<td>Code of the Business units which the user can access, separated by comma.</td>
</tr>
<tr>
<td>sn:</td>
<td>User Last Name</td>
</tr>
<tr>
<td>targetunit:</td>
<td>Code of the Parent Business Unit.</td>
</tr>
<tr>
<td>title:</td>
<td>User Title</td>
</tr>
</tbody>
</table>

Importing Data from Siebel UCM

Before importing data from Siebel UCM, ensure the following:

- Export resources from Siebel UCM, using instructions provided in Exporting Data from Siebel UCM.
• Enter Code values for the fields `businessunit`, `homebranch`, `targetunit`, `accessiblebusinessunits`, and `accessibleTargetUnits`.

• Ensure that you have created necessary marketing entities and business units. For more information, see Creating Enterprise Structure in Oracle Banking Platform.

• All Marketing Entities and Business Units in Oracle Banking Platform can be found in `ME991` screen for Market Entity - Business Unit Definition.

Refer to the Understanding Column Mappings section to understand the column mappings between Oracle Banking Platform and Siebel UCM:

### Importing Data from Oracle Sales Cloud

Before importing data from Oracle Sales Cloud, ensure the following:

• Export both employee resource and person (for Title field) data from Oracle Sales Cloud. Use instructions provided in Exporting Data from Oracle Sales Cloud.

• Enter code values for fields `businessunit`, `homebranch`, `targetunit`, `accessiblebusinessunits`, and `accessibleTargetUnits`.

• Ensure that you have created necessary marketing entities and business units. For more information, see Creating Enterprise Structure in Oracle Banking Platform. Business units and marketing entities can be found in Oracle Banking Platform `ME991` screen.

Refer to the Understanding Column Mappings section to understand the column mappings between Oracle Banking Platform and Oracle Sales Cloud:

### Understanding Column Mappings

<table>
<thead>
<tr>
<th>Column names in Oracle Banking Platform</th>
<th>Column Names from the file exported from Siebel UCM</th>
<th>Column Names from the file exported from Oracle Sales Cloud</th>
<th>Additional Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>dn</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>preferredlanguage</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>givenname</td>
<td>First Name</td>
<td>First Name</td>
<td>NA</td>
</tr>
<tr>
<td>accessiblebusinessunits</td>
<td>Organization</td>
<td>Associated Business Units</td>
<td>Associates all business units or organizations associated with the user.</td>
</tr>
<tr>
<td>objectclass</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>uid</td>
<td>User ID</td>
<td>User Name</td>
<td>NA</td>
</tr>
<tr>
<td>mail</td>
<td>Email addr</td>
<td>Primary Email</td>
<td>Assign only primary business units or organizations.</td>
</tr>
<tr>
<td>businessunit</td>
<td>Organization</td>
<td>Business Unit</td>
<td>NA</td>
</tr>
<tr>
<td>cn</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
</tbody>
</table>
Performing Bulk Import and Export of Employee Data

<table>
<thead>
<tr>
<th>Column names in Oracle Banking Platform</th>
<th>Column Names from the file exported from Siebel UCM Cloud</th>
<th>Column Names from the file exported from Oracle Sales Cloud</th>
<th>Additional Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>telephonenumber</td>
<td>Work Phone #</td>
<td>Primary Phone</td>
<td>NA</td>
</tr>
<tr>
<td>accessible targetunits</td>
<td>Organization</td>
<td>Associated Business Units</td>
<td>Associate all business units or organizations associated with the user.</td>
</tr>
<tr>
<td>sn</td>
<td>Last Name</td>
<td>Last Name</td>
<td>NA</td>
</tr>
<tr>
<td>targetunit</td>
<td>Organization</td>
<td>Business Unit</td>
<td>NA</td>
</tr>
<tr>
<td>title</td>
<td>Mr/Ms</td>
<td>Title</td>
<td>NA</td>
</tr>
<tr>
<td>homebranch</td>
<td>NA</td>
<td>Transaction Branch</td>
<td>NA</td>
</tr>
</tbody>
</table>

**Related Topics**
- Understanding Terminologies Across Applications

**Importing Employees Data to Oracle Banking Platform: Procedure**

Perform the following steps to import entries from an .ldif file to Oracle Banking Platform:

1. Sign in to ODSM.
2. Click **Connect to a directory** and select required OID.
   
   If you cannot find required OID, create a new connection using steps provided in Creating New Connections section in Creating Users and Setting Up User Attributes and Roles in Oracle Banking Platform.
3. Once you are connected to the OID, navigate to the **Data Browser** tab.
4. Expand `dc=com > dc=oracle > dc=in > cn=Users`.
5. Right-click `cn=Users` and select **Import LDIF**.
6. On the Import File dialog, enter the path to the LDIF file to be imported and click **Open**.
7. Click **OK**.

Import Process window shows the progress of the import operation. Expand **View Import Process Table** to see detailed progress.
8. Once import is finished, **Data Browser** tree refreshes and contains new entries for the imported users.

Importing and Exporting Cross References
Cross Reference for Employees and Business Units: Overview

This topic provides necessary cross reference related information required to integrate Oracle Sales Cloud for Financial Services with Siebel UCM and Oracle Banking Platform.

Note: Ensure that all the necessary objects from `employee_import_export_xref.sif` have been imported and compiled before you proceed.

Understanding Cross References

Cross references enable you to dynamically map values for equivalent entities created in different applications. When you create or update a record in Oracle Sales Cloud, the changes must be propagated to Siebel UCM and Oracle Banking Platform, for the same record. However, these applications have different entities to represent same information.

For example, Business Units in Oracle Sales Cloud are known as Marketing Entities or Business Units in Oracle Banking Platform, and Organizations in Siebel UCM. So, for each new Business Unit in Oracle Sales Cloud, a new row is inserted in its Business Unit database with unique identifier. Let’s call it BU_01. When the same information is propagated to Siebel UCM and Oracle Banking Platform, a new row should be inserted with different identifier, say Org_01 and ME_01. In such cases, you need some type of functionality to map these identifiers with each other so that they can be interpreted by different applications to be referring to the same entity. This is done by using cross references.

Application Cross References

The following table lists cross references for Resources and Business Units that are stored in Oracle Sales Cloud and Siebel UCM.

<table>
<thead>
<tr>
<th>Application</th>
<th>Cross References Stored</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Sales Cloud</td>
<td>• Siebel UCM Employees&lt;br&gt;• Siebel UCM Organizations&lt;br&gt;• Oracle Banking Platform Business Units and Marketing Entities</td>
</tr>
<tr>
<td>Siebel UCM</td>
<td>• Oracle Sales Cloud Resources&lt;br&gt;• Oracle Sales Cloud Business Units</td>
</tr>
</tbody>
</table>

Use the XREF IDs listed in the following table for entities in different applications.

<table>
<thead>
<tr>
<th>Entity Name (in Oracle Sales Cloud)</th>
<th>XREF ID for Oracle Sales Cloud</th>
<th>XREF ID for Oracle Banking Platform</th>
<th>XREF ID for Siebel UCM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resource</td>
<td>Party ID</td>
<td>Position ID</td>
<td>NA</td>
</tr>
<tr>
<td>Business Unit</td>
<td>Business Unit ID</td>
<td>Organization ID</td>
<td>undefined</td>
</tr>
</tbody>
</table>

Related Topics

• Understanding Terminologies Across Applications
Exporting Cross References from Oracle Sales Cloud: Explained

Exporting cross references comprises of exporting employee details for party ID and exporting business unit IDs.

Exporting Employee Details for Party ID

Use Bulk Export Process to extract the data from Oracle Sales Cloud.

1. Sign into Oracle Sales Cloud using Administrator role.
2. In the Setup and Maintenance work area, go to the following:
   a. Offering: Sales
   b. Functional Area: Data Import and Export
   c. Task: Schedule Export Processes
3. On the Overview page, click **Actions > Create** to define a new Bulk Export Process.
4. On the Enter Basic Information step, provide a valid name for the Bulk Export Process and click **Next**.
5. On the **Configure Export Objects** step, in the Export Objects region, click **Actions > Create**.
6. On the Manage Export Objects dialog box, move **Employee Resource** to the Selected Objects, and click **Done**.
7. In the Details region, expand **EmployeeResourceExpVO** and ensure that only fields **User Name** and **Party ID** have **Enabled** check box selected.
8. Click **Next**.
9. On the Create Schedule step, click **Next**.
10. On the Review step, review your Export Process Definition and click **Activate**.
11. To download the exported data file,
   a. On the Overview page, click **Refresh** to verify if the export process is completed.
   b. Select your Export Process from the list.
   c. In the History region, the .csv file should be listed in the **Export Data file** column. Click the .csv file to download.

Use the information from the exported .csv file to fill in OSC_EMPLOYEES worksheet of the template employees_xref_template_for_osc_and_och.xlsm.

Exporting Business Unit IDs

To export business unit IDs:

1. Sign into Oracle Sales Cloud using Administrator role.
2. In the Setup and Maintenance work area, go to the following:
   a. Functional Area: Application Extensions
   b. Task: Manage Business Unit
3. On the Manage Business Units page, click **Search** without specifying any search criteria.
4. Search Results region lists all the business units.
5. Click **View > Columns** to ensure that all **BusinessUnitId** and **Name** columns are selected.
6. Click **Actions > Export to Excel** to save the exported .xls file.

Use the information from the .xls file to fill in OSC_BUSINESSUNIT worksheet of the template business_units_xref_template_for_osc_and_och.xlsm.
Related Topics

- Bulk Export: Overview
- Defining Bulk Export Process: Procedure

Exporting Cross References from Siebel UCM: Explained

Each employee record in Siebel UCM must have a unique Position associated with it.

To export employee details from Siebel UCM to obtain their Position ID, use the following steps:

1. Sign into Siebel UCM with a user that has administrative privileges.
2. From the Navigator menu, select Site Map.
3. On the next page, click Administration - User > Employees.
4. Click Employees Menu > Export...
5. Select All Rows in Current Query, Visible Columns, and Tab Delimited Text File.
6. Click Next.
7. Save the output .csv file.

Use the information from the .csv file to fill in OCH_EMPLOYEES worksheet of the template employees_xref_template_for_osc_and_och.xlsm.

To export organization details from Siebel UCM, use the following steps:

1. Sign into Siebel UCM with a user that has administrative privileges.
2. From the Navigator menu, select Site Map.
3. On the next page, click Administration - Group > Organizations.
4. Click Employees Menu > Export...
5. Select All Rows in Current Query, Visible Columns, and Tab Delimited Text File.
6. Click Next.
7. Save the output .csv file.

Use the information from the .csv file to fill in the OCH_ORGANIZATION worksheet of the template business_units_xref_template_for_osc_and_och.xlsm.

Exporting Cross References from Oracle Banking Platform: Explained

To obtain Marketing Entity codes and Business Unit codes from Oracle Banking Platform, use the following steps:

1. Sign into Oracle Banking Platform.
2. Search for ME991 using Fast Path.
4. On the Search dialog box, use available search options to find Marketing Entity Code for each Marketing Entity. Select the marketing entity from the search results.
5. Back on the Marketing Entity - Business Unit Definition page, all associated Business Unit related information, including Business Unit Codes, is listed in the Business Unit region.
6. Repeat these steps for all the Marketing Entities to find related Marketing Entity Codes and Business Units Codes.

Use the information obtained this procedure to fill in the OBP_TARGETUNIT worksheet of the template business_unit_xref_template_for_osc_and_och.xlsm.
Importing Cross References to Oracle Sales Cloud: Explained

Use this topic to import cross references for resources and business units to Oracle Sales Cloud.

Preparing File for Import of Resources Cross References

Cross References for employees are stored in the Original System Reference table. The record's original system, also referred to as its source system, is the unique identifier of an external system, and the record’s original system reference is a unique identifier of the record within that external system. If a record includes its original system and original system reference when it is imported, then that pair of attributes can uniquely identify the record when it is updated or when other records are associated with it. For many objects, you can associate two new records that are in the same import file by providing the original system original system reference pair for each record. You cannot use the internal ID because the internal ID does not yet exist for either record.

Use one of the following methods to create a .csv file for import:

- Create a new .csv file with the following column names and example values as shown in the following table.

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Description</th>
<th>Example Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>OrigSystem</td>
<td>This column should have same value for all the rows.</td>
<td>Siebel UCM</td>
</tr>
<tr>
<td>OwnerTableName</td>
<td>This column should have same value for all the rows.</td>
<td>HZ_PARTIES</td>
</tr>
<tr>
<td>ExistingOwnerId</td>
<td>This column should contain the Party ID for each employee, obtained from the exported data file from Oracle Sales Cloud.</td>
<td>NA</td>
</tr>
<tr>
<td>OrigSystemReference</td>
<td>This column should contain corresponding Position ID of the same employee, obtained from the exported data file from Siebel UCM.</td>
<td>NA</td>
</tr>
</tbody>
</table>

⚠️ Note: You must ensure that both ExistingOwnerId and OrigSystemReference columns contain Party ID and Position ID of the same employee.

- Use the spreadsheet file employees_xref_template_for_osc_and_och.xlsm to consolidate exported data and prepare data for import. This template helps consolidate the exported data from Siebel UCM and Oracle Sales Cloud and prepare cross reference data to be imported to Oracle Sales Cloud. Click Prepare Employee XREF Data for Import to create worksheet XREF_IMPORT_OSC. This worksheet contains consolidated cross reference data to be used for import.

Fill out the following worksheets in the template for this:

<table>
<thead>
<tr>
<th>Worksheet Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSC_EMPLOYEES</td>
<td>Employees exported from Oracle Sales Cloud through Scheduled Export Process task for Employee Resource Object. The worksheet should contain columns for User Name and Party ID columns.</td>
</tr>
</tbody>
</table>
Preparation of the File for Import of Business Units Cross References

The cross references for Business Units are stored in XREF Common Custom Object.

Use one of the following methods to create a .csv file for import:

- Create a new .csv file with the following column names and example values as shown in the following table.

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
<th>Example Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Record Name</td>
<td>A unique number for the record. The value must not present in the result of the DB query on HZ_REF_ENTITIES table. Use the following query to ensure the unique value.</td>
<td>For example, 100001</td>
</tr>
<tr>
<td>FusionObject Type</td>
<td>Business Unit</td>
<td>Business Unit</td>
</tr>
<tr>
<td>FusionRecord ID</td>
<td>Business Unit ID obtained from the exported data from Oracle Sales Cloud.</td>
<td>BU_ID</td>
</tr>
<tr>
<td>RemoteObject Type</td>
<td>Use Organization if the remote application is Siebel UCM and Target Unit if the remote application is Oracle Banking Platform.</td>
<td>Organization or Target Unit</td>
</tr>
<tr>
<td>Note: Target Unit refers to either Business Unit or Marketing Entity on Oracle Banking Platform.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RemoteRecord ID</td>
<td>ID of the record in the remote application. The value should be obtained from the exported data from the remote application.</td>
<td>Organization ID, ME_CODE, or BU_CODE</td>
</tr>
<tr>
<td></td>
<td>o Organization ID if the remote application is Siebel UCM.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>o ME_CODE or BU_CODE if the remote application is Oracle Banking Platform.</td>
<td></td>
</tr>
<tr>
<td>RemoteSystem ID</td>
<td>Indicates whether the remote application is Siebel UCM or Oracle Banking Platform.</td>
<td>OCH or OBP</td>
</tr>
</tbody>
</table>

- Use the spreadsheet file business_units_xref_template_for_osc_and_och.xlsm to consolidate exported data and prepare data for import. This Template helps consolidate the exported data from Siebel UCM, Oracle Banking Platform, and Oracle Sales Cloud and prepare cross-reference data to be imported to Oracle Sales Cloud. Click Prepare Business Unit XREF Data for Import to create worksheet XREF_IMPORT_OSC. This worksheet contains consolidated cross-reference data to be used for import.
Fill in the following worksheets in the template:

<table>
<thead>
<tr>
<th>Worksheet Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSC_BUSINESSUNIT</td>
<td>Business units exported from Oracle Sales Cloud through Manage Business Unit task. Fill in the Name and BusinessUnitId columns in this worksheet.</td>
</tr>
<tr>
<td>OBP_TARGETUNIT</td>
<td>Business units and marketing entities obtained from Market Entity - Business Unit Definition page in Oracle Banking Platform. Fill in the worksheet for Name and ME or BU Code columns.</td>
</tr>
<tr>
<td>OCH_ORGANIZATION</td>
<td>Organizations exported from Siebel UCM. Fill in the worksheet for Name and Organization Id columns.</td>
</tr>
</tbody>
</table>

**Importing Cross References to Oracle Sales Cloud: Procedure**

To import cross reference data to Original System Reference table or XREF Common Custom Object, perform the following steps:

1. Sign into Oracle Sales Cloud using Administrator role.
2. In the Setup and Maintenance work area, go to the following:
   - Offering: Sales
   - Functional Area: Data Import and Export
   - Task: Manage File Import Activities
3. On Manage Import Activities page, click **Create** icon.
4. On the Enter Import Options step, perform the following and click **Next**.
   - Enter a valid **Name** for the import.
   - From the **Object** drop-down list, select:
     - Source System Reference for employees, if the remote application is Siebel UCM.
     - Common Custom Object for business units, if the remote application is Oracle Banking Platform.
   - Select **Header row included** check box.
   - Select Desktop as **Uploaded From**.
   - Click **Browse** and select .csv file that you created.
5. Use one of the following tables to map the columns from the .csv file to the target objects and attributes depending on the cross reference being imported.

For business units cross references,, use the following table:

<table>
<thead>
<tr>
<th>Column in .csv File</th>
<th>Target Object</th>
<th>Target Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>Record Name</td>
<td>_ _ ORAFS_ _XREF_cWBIT</td>
<td>RecordName</td>
</tr>
<tr>
<td>FusionObjectType</td>
<td>_ _ ORAFS_ _XREF_cWBIT</td>
<td>_ _ ORAFS_ _ FusionObjectType_ c</td>
</tr>
<tr>
<td>FusionRecordID</td>
<td>_ _ ORAFS_ _XREF_cWBIT</td>
<td>_ _ ORAFS_ _ FusionRecordID_ c</td>
</tr>
<tr>
<td>RemoteObjectType</td>
<td>_ _ ORAFS_ _XREF_cWBIT</td>
<td>_ _ ORAFS_ _ RemoteObjectType_ c</td>
</tr>
</tbody>
</table>
Performing Bulk Import and Export of Employee Data

<table>
<thead>
<tr>
<th>Column in .csv File</th>
<th>Target Object</th>
<th>Target Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>RemoteRecordID</td>
<td>__ ORAFS__ XREF_cWBIT</td>
<td>__ ORAFS__ RemoteRecordID_c</td>
</tr>
<tr>
<td>RemoteSystemID</td>
<td>__ ORAFS__ XREF_cWBIT</td>
<td>__ ORAFS__ RemoteSystemID_c</td>
</tr>
</tbody>
</table>

For employees cross references, use the following table:

<table>
<thead>
<tr>
<th>Column in .csv File</th>
<th>Target Object</th>
<th>Target Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>ExistingOwnerTableId</td>
<td>SourceSystemReference</td>
<td>ExistingOwnerTableId</td>
</tr>
<tr>
<td>Origsystem</td>
<td>SourceSystemReference</td>
<td>OrigSystem</td>
</tr>
<tr>
<td>OrigSystemReference</td>
<td>SourceSystemReference</td>
<td>OrigSystemReference</td>
</tr>
<tr>
<td>OwnerTableName</td>
<td>SourceSystemReference</td>
<td>OwnerTableName</td>
</tr>
</tbody>
</table>

6. Click **Next** til you reach Review and Activate step.
7. On the Review and Activate step, click **Activate** to submit the import immediately.
8. Back on the Manage File Import Objects page, click **Refresh** icon and check the **Status** column for the import activity that you have created. Ensure that the status is **Completed**.

**Troubleshooting Import**

Use the suggestions provided in the following table if you face any problems during your import.

<table>
<thead>
<tr>
<th>Problem or Error Message</th>
<th>Probable Cause and Solution</th>
</tr>
</thead>
</table>
| XREF object is not found while mapping the file columns to target objects. | This can be resolved by generating artifacts for the objects, by using the following steps:  
  1. Sign in to Oracle Sales Cloud with a user that has Custom Object Administration role assigned.  
  2. Click **Navigator > Application Composer**.  
  3. Click **Import and Export**.  
  4. Click **Generate** to generate artifacts for the object. |
| Error message: You must provide a value for the attribute RECORD_NAME. | This error occurs when the value in the Record Name column is missing in the import file. |
| No new records are created after import operation. | This could be due to records that are already existing with the same name as those in the import file. |

**Related Topics**

- Understanding Terminologies Across Applications
Importing Cross Reference to Siebel UCM: Explained

Use this topic to import cross references for employees and organizations to Siebel UCM.

Importing Cross References for Employees

Use one of the following two methods to create the import file for Siebel UCM:

- Create a new .csv file with the following columns and values:

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Position</td>
<td>Position of the employee in Siebel UCM</td>
</tr>
<tr>
<td>Division</td>
<td>Division of the employee in Siebel UCM</td>
</tr>
<tr>
<td>Integration id</td>
<td>Party ID of the same employee in Oracle Sales Cloud</td>
</tr>
</tbody>
</table>

- Use the spreadsheet file employees_template_for_import_into_och.xlsm to consolidate exported data and prepare data for import. This template helps consolidate the exported data from Siebel UCM and Oracle Sales Cloud and prepare cross reference data to be imported to Siebel UCM. Click Prepare Employee XREF Data for Import to create worksheet XREF_IMPORT_OCH. This worksheet contains consolidated cross reference data to be used for import. Fill out the following worksheets in the template for this:

<table>
<thead>
<tr>
<th>Worksheet Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSC_EMPLOYEES</td>
<td>Employees exported from Oracle Sales Cloud through Scheduled Export Process task for Employee Resource Object. The worksheet should contain columns for User Name and Party ID columns.</td>
</tr>
<tr>
<td>OCH_EMPLOYEES</td>
<td>Employees exported from Siebel UCM. The worksheet should contain columns for User ID, Position, Position ID, and Division.</td>
</tr>
</tbody>
</table>

Use the following procedure to import cross references for employees in Siebel UCM:

1. Sign in to Siebel UCM with a user that has administrative privileges.
2. From the Navigator menu, select Site Map.
3. On the next page, click Administration - Group > Positions.
4. Click Positions Menu > Import....
5. On the Import dialog box, select the following:
   - Input File: the .csv file that you have created.
Performing Bulk Import and Export of Employee Data

- **Input Format**: Comma Separated Text File.
- **Input Source**: Auto Mapping as.
- **Conflict Resolution**: Overwriting Existing Record.

6. Click Next.
7. In the Field Mappings step, ensure that all the three .csv file columns are mapped correctly to the corresponding Siebel UCM field. If not, select the fields to be mapped and click Update Mapping.
8. Click Next to start import of data.

**Importing Cross References for Organizations**

Use one of the following methods to prepare .csv files for importing organization cross references:

- Create a new .csv file with the following columns and values:

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Integration Id</td>
<td>Business Unit ID of the corresponding Business Unit.</td>
</tr>
<tr>
<td>Organization Id</td>
<td>Organization ID of the organization for which cross reference is to be stored.</td>
</tr>
</tbody>
</table>

- Use the spreadsheet file business_units_xref_template_for_osc_and_och.xlsm to consolidate exported data and prepare data for import. This template helps consolidate the exported data from Siebel UCM, Oracle Sales Cloud, and Oracle Banking Platform and prepare cross reference data to be imported to Siebel UCM. Click Prepare Business Unit XREF Data for Import to create worksheet XREF_IMPORT_OCH. This worksheet contains consolidated cross reference data to be used for import.

Fill out the following worksheets in the template for this:

<table>
<thead>
<tr>
<th>Worksheet Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSC_ BUSINESSUNIT</td>
<td>Business Unit data exported from Oracle Sales Cloud through Manage Business Unit task. Fill in the worksheet for Name and BusinessUnitId columns.</td>
</tr>
<tr>
<td>OBP_TARGETUNIT</td>
<td>Business Units and Market Entities data obtained from screen Market Entity - Business Unit Definition page on Oracle Banking Platform. Fill the worksheet for Name and ME or BU Code columns.</td>
</tr>
<tr>
<td>OCH_ ORGANIZATION</td>
<td>Organizations data exported from Siebel UCM. Fill in the worksheet for Name and Organization Id columns.</td>
</tr>
</tbody>
</table>

Use the following procedure to import cross references for organizations:

1. Sign in to Siebel UCM with a user that has administrative privileges.
2. From the Navigator menu, select Site Map.
3. On the next page, click Administration - Group > Organizations.
4. Click Organizations Menu > Import....
5. On the Import dialog box, select the following:
   - **Input File**: the .csv file that you have created as.
   - **Input Format**: Comma Separated Text File.
6. Click **Next**.
7. In the Field Mappings step, ensure that all the three .csv file columns are mapped correctly to the corresponding Siebel UCM field. If not, select the fields to be mapped and click **Update Mapping**.
8. Click **Next** to start import of data.
Using Integrated Sales Cloud for Financials

Using Integrated Sales Cloud for Financials: Overview

The Integrated Sales Cloud for Financial Services enables you to create, manage, and track your customer records and the respective financial accounts.

As a banker or financial adviser for your bank, you can use the following key capabilities:

- Create an automated process from lead to financial account origination with a single application, with minimal or almost no intervention once the opportunity is submitted.
- Perform Needs Analysis for customers in lead, opportunity, company, contact, and household.
- Onboard the contact or company from the Contact or Account summary view or after the opportunity is submitted.
- Once onboarded, view or perform due diligence checks by clicking the "Due Diligence Summary" button.
- Enter key customer information once and reuse the same information for your Application Form process as well.
- Once the opportunity is submitted, complete the application details through the Application Form and manage and update your application status through the Application Tracker.

Differences in Terminology between Oracle Sales Cloud and Oracle Banking Platform

Oracle Sales Cloud uses terms that are referred to differently in Oracle Banking Platform (OBP). The following table describes the difference in terminology.

<table>
<thead>
<tr>
<th>Term in Oracle Sales Cloud</th>
<th>Term in Oracle Banking Platform</th>
</tr>
</thead>
<tbody>
<tr>
<td>Opportunity</td>
<td>Submission</td>
</tr>
<tr>
<td>Product</td>
<td>Offers</td>
</tr>
<tr>
<td>Financial Accounts</td>
<td>Accounts</td>
</tr>
<tr>
<td>Contacts</td>
<td>Individuals</td>
</tr>
<tr>
<td>Accounts</td>
<td>Organizations</td>
</tr>
</tbody>
</table>
Terminology and Definitions

The following table defines the various terminologies used to demonstrate the use of Integrated Sales Cloud for Financials.

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Form</td>
<td>Application Form is an account opening request placed by one or more applicants for one or more products offered by the bank. An application form comprises of various stages for which the banker has to record the applicant’s details and the details of the product that is being applied for.</td>
</tr>
<tr>
<td>Application Tracker</td>
<td>Application Tracker helps the banker to monitor the status of an account opening application submitted by an applicant. It is also used to update information and perform various actions to process the application.</td>
</tr>
<tr>
<td>KYC</td>
<td>Know Your Customer refers to the activities related to customer due diligence that banks must perform to identify their clients and ascertain relevant information pertinent for doing financial business with them and bank regulation which governs those activities.</td>
</tr>
<tr>
<td>Offers</td>
<td>Offers are different variants of a product offered to the customer based on market research and customer expectation. Offers facilitate to provide more than the product itself by including different combinations of the features of the product that represent additional value to the customer.</td>
</tr>
<tr>
<td>Onboarding</td>
<td>Onboarding refers to the process of enrolling a party (including a customer, service provider or other such entity) with the bank by recording its details and uploading the required documents.</td>
</tr>
<tr>
<td>Submission</td>
<td>A submission contains one or more account opening applications submitted by the applicants in a single request.</td>
</tr>
</tbody>
</table>

Using Integrated Sales Cloud for Financials: Business Scenario

Daniel Grayson owns Green Corp. and wants to get a loan for his company from the Vision Corporation bank. Daniel participates in the marketing promotions of the Vision Corporation bank and as a result a lead is generated and assigned to a banker, who further contacts Daniel about his requirements. The banker converts the lead to an opportunity and proceeds with Customer Onboarding by clicking the Onboarding Customer button in OBP, which opens up the Party Details page where the information created by the Sales Representative is already pre-populated from OSC.

On successful completion of the Onboarding process, the banker initiates the Due Diligence Checks. On confirmation of the KYC check, the Onboarding Status appears as Successful and the KYC Status appears as Confirmed on the Contact page. The banker clicks the Create Opportunity button on the Edit Contact page and adds Daniel as an Opportunity after performing a Needs Analysis. Depending on Daniel’s requirements, the banker adds the relevant products and submits the opportunity to OBP. On Opportunity Submission, the Submission Status changes to In Progress and a Submission ID is generated.

The banker proceeds with configuring the account by filling in all the necessary details in the Application Form and submitting it. If Auto-Approval of products is applicable, a Financial Account is created. Once the submission is processed, the banker goes to the Application Tracker on the Opportunity page, to view the Application status and the Financial Account details. To
view the details of the financial account created, the banker goes to the Financial Account subtab on the Company/Contact/Household 360 degree view page.

An alternate scenario is that post identifying Daniel as a HOT Lead, the banker converts the lead to an opportunity, followed by Needs Analysis. Post Opportunity Submission, the Application Form is filled and Customer Onboarding is performed. The Application form is submitted and a Financial Account is created, the details of which can be seen on the Financial Account subtab on the Company/Contact/Household 360 degree view page.

Customer Onboarding: Overview

This topic describes customer onboarding, which is the process of capturing customer information and adding the customer to the Oracle Banking Platform (OBP) for validation.

You can onboard a customer in the two following ways:

- You can do it directly for a Contact or an Company by clicking the Onboard Customer button from the Edit Contact page.
- You submit an Opportunity through Opportunity Submission.

Customer onboarding is performed for Contacts and Companies but not for Households. Currently, customer onboarding is a one-time process for Companies and Contacts. Depending on the customer being an account or a contact, certain onboarding fields can be different. Once an onboarding is invoked but not completed, any details further updated in OSC do not get reflected in OBP.

An integral component of customer onboarding is Due Diligence checks. Once the Onboarding Status of a customer changes to Success, the View Due Diligence Summary button gets enabled. KYC checks are triggered for customers and on completion of KYC checks the KYC status changes to Confirmed. These checks mostly involve KYC, Anti-money laundering, and FATCA. The components of this check can depend on your geography and so there could be different checks for US and India.

Opportunity Submission

You can perform opportunity submission for a customer who is already onboarded or for a new customer who is not onboarded. Submission happens through the Submit Opportunity action, which has certain mandatory conditions and validations associated with it. The customer information must abide by the following validations and mandatory conditions to complete a submission:

- Confirming that an opportunity must be submitted with a product. On the Edit opportunity page, the Customer name and Product details must be populated. If an opportunity is submitted without a customer then an error message "Enter a Contact or a Company" appears, whereas if an opportunity is submitted without a product, an error message "You must select at least one product when you are submitting an opportunity" appears.
- Validating that all the products for an opportunity must have the same currency. If the products in an opportunity have different currencies, an error message "{PRODUCT_NAME} is not maintained in the {OPPORTUNITY_CURRENCY}. Please enter another product." appears.
- Contact or company information for which an opportunity is being submitted must exist in Siebel UCM. If the OBP is not able to query the customer data from Siebel UCM, an error message "You cannot submit the opportunity now. Try again later" appears. This mandate checks that OBP does have the customer record.
• While performing an opportunity submission for a member of your household, the Primary Contact field must be populated, else an error message “You must associate the opportunity with a primary contact before submitting the opportunity” appears.

Every submission in OBP corresponds to one and only one opportunity in OSC. Post Opportunity Submission, the Submission Status and the Product Status in the Revenue Lines changes to In Progress and the Submission ID is generated. This indicates that the opportunity is submitted and the submission is in progress in OBP. Once the Submission ID is generated, the record must be saved. If the record is not saved, the Submission ID is lost. On Submission ID generation, the Open Application Tracker and Open Application Form buttons are also enabled. The Customer, Submission Status, and the Products Details can no longer be changed at this stage.

### Application Form

Opportunity submission creates a draft submission at OBP. The details of the draft submission are filled out in a process called Application Form. On filling up of the Application Form, the approval process is initiated. A submission can be split into a single or multiple applications depending on the OBP rules.

If the customer is not already onboarded, then the customer gets onboarded during the processing of the application form. If the customer is already onboarded, then the customer information must be already filled in the Application Form.

### Application Tracker

The status of all the submissions and their respective applications can be viewed through the Application Tracker. Once the submission is completed, the opportunity status changes to Won or Lost. If the opportunity status is Won, new financial account(s) gets displayed in the Customer 360 degree view in OSC.

### Financial Accounts: Overview

At the end of the submission process, a financial account is created in OSC. You can:

• View financial account detail in Customer 360.
• Drill-down to the details of the financial account.
• View the holders of the financial account for retail banking customers.
• View a detailed statement, if the product is serviced from OBP.
• Access the roll-up of the financial account as a direct member of the household.