Oracle

Sales Cloud

Integrating with Oracle E-Business Suite, Release 12.1.3

Release 13 (update 18B)
This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

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

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

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

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

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

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

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

The business names used in this documentation are fictitious, and are not intended to identify any real companies currently or previously in existence.
# Contents

## Preface

1. **Introduction to the Integration**
   - Overview of Oracle E-Business Suite and Oracle Sales Cloud Integration
   - Integration Component Architecture
   - Integration Services
   - Integration Process Flows
   - Customer Data Matching with Oracle E-Business Suite
   - Contact Data Matching with Oracle E-Business Suite
   - Quote Creation in Oracle E-Business Suite
   - Configuration Roadmap

2. **Oracle E-Business Suite Configuration**
   - Software Requirements for Oracle E-Business Suite
   - Applying Oracle E-Business Suite Patches
   - Creating a Generic Error Page
   - Assigning Roles to the Oracle E-Business Suite User
   - Exposing Web Services
   - Deploying Web Services
   - Process of Creating a Quotes Menu
## 3 Oracle Sales Cloud Configuration

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Before You Configure Oracle Sales Cloud</td>
<td>17</td>
</tr>
<tr>
<td>Software Requirements for Oracle Sales Cloud</td>
<td>17</td>
</tr>
<tr>
<td>Creating Web Service References to Oracle E-Business Suite Web Services</td>
<td>17</td>
</tr>
<tr>
<td>Using Sandboxes</td>
<td>20</td>
</tr>
<tr>
<td>About Creating Functions</td>
<td>21</td>
</tr>
<tr>
<td>About Creating Global Functions</td>
<td>21</td>
</tr>
<tr>
<td>Creating Global Functions</td>
<td>22</td>
</tr>
<tr>
<td>Creating Object Functions: Explained</td>
<td>22</td>
</tr>
<tr>
<td>Creating Validation Rules</td>
<td>23</td>
</tr>
<tr>
<td>Creating Custom Objects</td>
<td>23</td>
</tr>
<tr>
<td>Creating the Integration Configuration Custom Object</td>
<td>24</td>
</tr>
<tr>
<td>Creating the XREF Custom Object</td>
<td>27</td>
</tr>
<tr>
<td>Creating a Field Mapping Custom Object</td>
<td>28</td>
</tr>
<tr>
<td>Creating the EBS Match Child Object</td>
<td>30</td>
</tr>
<tr>
<td>Configuring Standard Objects</td>
<td>34</td>
</tr>
<tr>
<td>Configuring the Account Standard Object</td>
<td>35</td>
</tr>
<tr>
<td>Configuring the EBS Match Child Object</td>
<td>38</td>
</tr>
<tr>
<td>Configuring the Opportunity Standard Object</td>
<td>39</td>
</tr>
<tr>
<td>Connecting Oracle Sales Cloud to Oracle E-Business Suite</td>
<td>41</td>
</tr>
</tbody>
</table>

## 4 Postconfiguration Tasks

<table>
<thead>
<tr>
<th>Task</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mapping Products</td>
<td>45</td>
</tr>
<tr>
<td>Extracting the List of Oracle E-Business Suite Products</td>
<td>45</td>
</tr>
<tr>
<td>Extracting a List of Oracle Sales Cloud Products</td>
<td>46</td>
</tr>
<tr>
<td>Mapping Products Manually</td>
<td>46</td>
</tr>
<tr>
<td>Entering Mappings into Oracle Sales Cloud</td>
<td>47</td>
</tr>
<tr>
<td>Mapping Product Groups: Explained</td>
<td>47</td>
</tr>
<tr>
<td>About Hiding Oracle E-Business Suite UI Elements for Streamlined Navigation</td>
<td>50</td>
</tr>
<tr>
<td>Hiding Oracle E-Business Suite UI Elements</td>
<td>56</td>
</tr>
<tr>
<td>Testing the Integration: Explained</td>
<td>57</td>
</tr>
<tr>
<td>Configuring the Integration with the Security Console</td>
<td>57</td>
</tr>
</tbody>
</table>

## 5 Required Files

<table>
<thead>
<tr>
<th>File</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Required HTML File: Explained</td>
<td>59</td>
</tr>
<tr>
<td>Required Groovy Files: Explained</td>
<td>59</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><strong>boldface</strong></td>
<td>Boldface type indicates user interface elements, navigation paths, or values you enter or select.</td>
</tr>
<tr>
<td><strong>monospace</strong></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.
Introduction to the Integration

Overview of Oracle E-Business Suite and Oracle Sales Cloud Integration

This guide outlines the implementation and configuration steps that are required to integrate customer- and opportunity-management processes in Oracle Sales Cloud with quote-management processes in Oracle E-Business Suite.

The integration is designed to support customers who want to take advantage of the latest capabilities of the Oracle Sales Cloud application, and use their existing investment in an on-premise enterprise resource planning (ERP) system that provides complex product configuration, pricing, or discount information. In the integration, Oracle E-Business Suite quotes can be created from Oracle Sales Cloud opportunities. The Oracle E-Business Suite quote UI is embedded within a tab in the Opportunity Detail screen in Oracle Sales Cloud.

This guide is designed to be used as a template. The guide is a starting point that shows how Oracle Sales Cloud and Oracle E-Business Suite can be connected to create a value-added business process and user experience. You must enter the documented configurations and install the documented patches to create the integration.

However, it is not a turnkey solution. Each implementation of Oracle Sales Cloud and Oracle E-Business Suite is unique, and each customer has different needs that have led them to implement application configurations that support their unique business requirements. While the steps in this guide describe how to connect a non-configured Oracle Sales Cloud instance to a non-configured Oracle E-Business Suite instance, they can be combined with configurations that have already been applied to each instance.

Note that most settings related to security are not included in this document. For information and recommendations, see Oracle Sales Cloud Security Reference and Securing Oracle Sales Cloud both on Oracle Help Center (https://docs.oracle.com/en/).

Integration Component Architecture

The following lists the files required to implement the integration between Oracle Sales Cloud and Oracle E-Business Suite. Refer to the Related Topics section in this topic for a link to Integrating Oracle Sales Cloud with Oracle E-Business Suite (Doc ID 1643818.1) on My Oracle Support. The files are located in the Attachments section of the article.

- **Oracle Sales Cloud standard objects.** Standard objects in the Sales, and Common applications are configured by adding fields, object functions, triggers, and validations, and by configuring UI pages.

- **Oracle Sales Cloud custom objects.** Custom objects are created for the integration, and fields, pages, functions, and so on are added to the objects.

- **Groovy functions.** Groovy scripts create functions, triggers, and validations that modify the behavior of Oracle Sales Cloud objects.

- **Oracle E-Business Suite web services.** Oracle Sales Cloud calls Oracle E-Business Suite web services directly. For more information on web services, see the Integration Services topic.

- **Oracle E-Business Suite Quotes module.** Opportunities in the Sales application in Oracle Sales Cloud are integrated with the Quotes module in Oracle E-Business Suite.
The following figure shows the components of the integration between Oracle Sales Cloud and Oracle E-Business Suite and their relationship to each other.

Related Topics

- Integration Implementation Files on My Oracle Support
Integration Services

The following Oracle E-Business Suite web services are used in the integration:

- **DQM Search Service.** This web service allows the user to search for organizations. It returns a list of candidates based on configurable criteria defined in a Data Quality Management (DQM) matching rule within Oracle Trading Community. DQM Search Service returns records that match the specified criteria. It returns duplicate and multiple records.

- **Organization Service.** This web service manages organization records in Oracle E-Business Suite. New organizations are created and retrieved by Oracle Sales Cloud.

- **Organization Contact Service.** This web service manages contact records for specific organizations in Oracle E-Business Suite. It creates new contacts in Oracle E-Business Suite from Oracle Sales Cloud.

- **Quote Service.** This web service can be used to create quotes in Oracle E-Business Suite based on opportunities in Oracle Sales Cloud.

Integration Process Flows

The integration between Oracle Sales Cloud and Oracle E-Business Suite supports the following three main process flows:

- Matching customer data between Oracle Sales Cloud and Oracle E-Business Suite
- Matching contact data with Oracle E-Business Suite
- Creating quotes in Oracle E-Business Suite

Customer Data Matching with Oracle E-Business Suite

This process allows you to match customer records in Oracle Sales Cloud with Oracle E-Business Suite. During the customer synchronization process, customer records must be searched in Oracle E-Business Suite before synchronizing. The process calls the DQM Search Service web service in Oracle E-Business Suite, which returns a list of potential matches. DQM Search Service also offers fuzzy-matching capabilities.

If the customer record exists in Oracle E-Business Suite and there is only one match, then the process saves a cross-reference in Oracle Sales Cloud as an XREF custom object. The cross-reference contains the local identification in Oracle Sales Cloud and the remote identification in Oracle E-Business Suite.

If the customer record exists in Oracle E-Business Suite and has multiple potential matches, then the user is presented with a user interface (UI) where the potential matches are displayed. The user selects the candidate match. The process then associates the record in Oracle E-Business Suite with the record in Oracle Sales Cloud by saving cross-references.

If the customer record does not exist in Oracle E-Business Suite, then the Organization Service web service is called in Oracle E-Business Suite to create a new customer record. The process then saves a cross-reference in Oracle Sales Cloud with the local identification in Oracle Sales Cloud and the remote identification in Oracle E-Business Suite.

Detailed Customer Data Matching Process
The following figure illustrates the customer data matching sequence.

When a salesperson performs a customer matching operation in Oracle Sales Cloud the following sequence takes place:

1. The process returns the Account object from Oracle Sales Cloud, and determines the customer synchronization status.
2. If the customer is not synchronized, or if the status is null, then the process calls the FindParties method in the DQM Search Service web service to perform a customer search in Oracle E-Business Suite. The customer search returns a list of matching candidates.
3. If there is no matching record, then the process calls the CreateOrganization method in the Organization Service web service in Oracle E-Business Suite. Organization Service returns the remote customer identification.
4. The process creates an XREF custom object in Oracle Sales Cloud using the local identification in Oracle Sales Cloud and the remote identification from Oracle E-Business Suite.
5. The process sets the synchronization status to Synchronized.

Contact Data Matching with Oracle E-Business Suite

This process allows you to match contact records in Oracle Sales Cloud with Oracle E-Business Suite. When the user clicks the Quotes tab or the Create Quote button in an opportunity in Oracle Sales Cloud, the primary contact assigned to the opportunity is compared with organization records in Oracle E-Business Suite. The primary contact is a contact in Oracle Sales Cloud that has a null cross-reference in Oracle E-Business Suite.
The process retrieves the entire organization record from Oracle E-Business Suite and loops through all of the contact records in the organization record. If it finds a single match, then it creates a cross-reference for the contact in Oracle E-Business Suite: it stores the local identification for the contact in Oracle Sales Cloud as the remote identification in Oracle E-Business Suite.

If the process does not find a match, or if there are multiple potential matches, then it calls the Create Organization Service web service in Oracle E-Business Suite. It also stores the local identification of the contact in Oracle Sales Cloud as the remote identification in Oracle E-Business Suite.

Detailed Contact Data Matching Process
The following figure details the contact data matching sequence.

When a salesperson clicks the Quote tab or the Create Quote button in an opportunity in Oracle Sales Cloud:

1. The process returns the Account object from Oracle Sales Cloud, and determines the synchronization status.
2. If the customer was synchronized previously, then the process returns the remote customer identification from the XREF custom object. The remote customer identification already exists in Oracle E-Business Suite because of a previous synchronization.
3. The process reads the remote contact identification from the XREF object.
4. The process fetches the primary contact from the Opportunity Contact object in Oracle Sales Cloud.
5. The process calls the GetOrganization method in the Organization Service web service in Oracle E-Business Suite using the remote customer identification as an input argument.
6. If the returned organization has a matching contact, then the process creates an XREF object with the local and remote contact identifications.

7. If there are no matching contacts or more than one matching contact, then the process calls the createOrgContact method in the Organization Contact Service web service in Oracle E-Business Suite. The createOrgContact method returns the remote contact identification from Oracle E-Business Suite.

8. The process creates an XREF object using the local and remote contact identifications.

**Quote Creation in Oracle E-Business Suite**

This process allows you to synchronize quote data with Oracle E-Business Suite. After the customer record is created in Oracle Sales Cloud and synchronized with Oracle E-Business Suite, the Oracle E-Business Suite quote UI can be accessed. The quote UI is embedded within a tab in the Opportunity Detail screen in Oracle Sales Cloud.

This embedded quote UI allows the user to take advantage of Oracle E-Business Suite directly in Oracle Sales Cloud. The quotes, orders, products, pricing, and configuration details are captured in Oracle E-Business Suite. The process also maps revenue line items in opportunity records in Oracle Sales Cloud to quote line items in Oracle E-Business Suite.

**Detailed Quote Creation Process**

The following describes the seven-step quote creation process.
The following figure displays the seven-step quote creation sequence.

When a salesperson clicks Create Quote on an Opportunity page in Oracle Sales Cloud:

1. The process returns the Account object in Oracle Sales Cloud, and determines the synchronization status.
2. If the customer is synchronized, then the process reads the remote contact identification from the XREF object. If the customer is not synchronized, then an error is returned.
3. If the remote contact identification is not found, then the contact synchronization process is executed.
4. The quote creation process collects opportunity details such as revenue line items from the Opportunity Revenue object.
5. The quote creation process attempts to map all of the revenue lines in the opportunity record to Oracle E-Business Suite products.
6. The quote creation process calls the Create Quote method in the Quote Service web service in Oracle E-Business Suite, which returns the quote number.

7. The quote creation status, which includes the quote number, is updated in the opportunity. This status does not persist with the opportunity.

Configuration Roadmap

To use this integration, you must configure application artifacts in both Oracle E-Business Suite and Oracle Sales Cloud. In the Oracle E-Business Suite instance several patches must be applied, and web services must be configured and exposed. In the Oracle Sales Cloud instance, custom Groovy logic must be added, custom objects created, and standard objects enhanced. These steps must be executed in order, because many of them rely on the successful completion of previous steps. The following figure shows a high-level overview of the configuration sequence.
The following figure displays a high-level overview of the configuration sequence.

Many of the Oracle Sales Cloud configuration steps in Chapter 3 have several component procedures, such as creating sandboxes, creating or augmenting objects, defining fields, defining business logic with functions, defining rules or triggers, and defining user interface changes. These component procedures are used in many different configuration steps, but not all procedures are used in all steps. For example, the procedure to create an object or the process to define a trigger is the same from one step to the next, but the names and details of the objects or triggers will change from step to step.

The general process for each component procedure is listed before the specific objects that are to be created or configured. The topics for the specific objects provide details on each of the configuration steps, and which component procedures and details are required in each step.
2 Oracle E-Business Suite Configuration

Software Requirements for Oracle E-Business Suite

The supported Oracle E-Business Suite version for the integration is 12.1.3.

Your Oracle E-Business Suite environment must have the following features for the integration to work:

- Oracle Quoting Application
- Oracle Trading Community Architecture
- Oracle Trading Community Architecture Data Quality Management
- Oracle E-Business Suite Integrated SOA Gateway

For the list of supported web browsers for Oracle E-Business Suite Release 13, see 389422.1 (Article ID) on My Oracle Support.

Applying Oracle E-Business Suite Patches

You must apply the patches listed in the following table to your Oracle E-Business Suite environment before performing any configuration steps.

<table>
<thead>
<tr>
<th>Patch</th>
<th>Description</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>13391839:R12. ASO.B</td>
<td>POST 12.1.3 ORDER CAPTURE (ASO) CUMULATIVE PATCH</td>
<td>GA</td>
</tr>
<tr>
<td>16794688:R12. QOT.B</td>
<td>INTERMEDIA SEARCH RETURNS NO RESULTS WITH ASO_WEB_ATTR_FILTER=YES</td>
<td>GA</td>
</tr>
<tr>
<td>16995310:R12. ASO.B</td>
<td>DIRECT ITEM ENTRY IS CLEARING ASO_QUOTE_LINES_ALL COLUMNS UPON UPDATE</td>
<td>GA</td>
</tr>
<tr>
<td>17025892:R12. QOT.B</td>
<td>FUSION CRM OPPORTUNITY INTEGRATION</td>
<td>GA</td>
</tr>
<tr>
<td>9328600:R12. OWF.B</td>
<td>12.1.3: REMOVE WSIF AND USE ORASAAJ IN SIF</td>
<td>GA</td>
</tr>
<tr>
<td>16872352:R12. OWF.B</td>
<td>10FF:12. 1.3: ADD NONCE AND CREATION TIME TO WSSE SECURITY HEADER FOR USERNAME TOKEN</td>
<td>GA</td>
</tr>
</tbody>
</table>
### Creating a Generic Error Page

You must create a custom, static error page. Oracle Sales Cloud will point to this error page if a user experiences an error while trying to access the correct view in the Oracle E-Business Suite Quotes module.

To create a generic error page, do the following:

1. Upload the following HTML file to the Oracle E-Business Suite application server: O_INT_EBS_ErrorPage.html.
2. Source the Oracle E-Business Suite applications environment file, if not already sourced.
3. Copy the uploaded HTML file to the $OA_HTML directory.

### Assigning Roles to the Oracle E-Business Suite User

The following roles must be assigned to the Oracle E-Business Suite user who will be used to authenticate the web services required by the integration:

- Trading Community Manager
- Customers Online Superuser
- Quoting User

To assign roles to the Oracle E-Business Suite user do the following:

1. Log in to Oracle E-Business Suite as a user with the system administrator role.
2. Navigate to **User Management**, then **Users**.
3. Search for the user.
4. In the search results, click the **Update** icon in the row for the user.
5. On the **Roles** tab, scroll through the required roles to confirm that they have been assigned to the user.
6. For each required role that has not been assigned, do the following:
   a. Click **Assign Roles**.
   b. In the **Search and Select: Assign Roles** dialog box, search for the role by **Roles and Responsibilities**.
   c. In the search results, select the check box for the role.
   d. Click **Select**.

### Exposing Web Services

For Oracle Sales Cloud to be able to call the Oracle E-Business Suite web services that you will be deploying, the web services will need to be exposed securely to the Internet. The Oracle Sales Cloud to Oracle E-Business Suite integration
requires that published web services be secured using Secure Sockets Layer (SSL). For more information, see the topic on Secure Sockets Layer Configuration in Oracle E-Business Suite System Administrator’s Guide - Configuration.

- Use a valid signed server certificate in the setup, not a self-signed certificate. Without a valid certificate Oracle Sales Cloud will not be able to call Oracle E-Business Suite web services.
- Use the standard HTTPS port (443) when enabling SSL in Oracle E-Business Suite so that Oracle Sales Cloud can call the web services.
- Do not disable HTTP access if you want users to be able to access Oracle E-Business Suite using standard HTTP from the internal network or over VPN. After configuring SSL, the firewall rules must be configured to allow your Oracle Sales Cloud instance to make inbound calls to the Oracle E-Business Suite web services.
- Do not allow inbound calls over the standard HTTP port (80).
- It is recommended to allow access only to the web service URLs for the web services specified in this guide.
- If possible, it is also recommended to only allow access from the Oracle Sales Cloud instance you are using.
- Enable these restrictions after testing successful access to the web services from Oracle Sales Cloud.

Deploying Web Services

You must deploy the required Oracle E-Business Suite web services which are listed in the following table. Oracle Sales Cloud calls these web services directly. The web services must also be granted to the Oracle E-Business Suite user with the roles specified in the Assigning Roles to the Oracle E-Business Suite User topic.

*Note:* When logging into Oracle E-Business Suite, make sure you use the HTTPS URL so that the generated Web Services Description Language files (WSDL) will also use HTTPS in their internal references.

The following table lists supported web services, their internal name, type and associated methods.

<table>
<thead>
<tr>
<th>Web Service</th>
<th>Internal Name</th>
<th>Type</th>
<th>Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>DQM Search Service</td>
<td>/ oracle/ apps/ ar/ hz/ service/ party/ DqmSearchService</td>
<td>Business Service Object</td>
<td>findParties</td>
</tr>
<tr>
<td>Organization Service</td>
<td>/ oracle/ apps/ ar/ hz/ service/ party/</td>
<td>Business Service Object</td>
<td>createOrganization</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>getOrganization</td>
</tr>
<tr>
<td>Organization Contact Service</td>
<td>OrgContactService</td>
<td>Business Service Object</td>
<td>createOrgContact</td>
</tr>
<tr>
<td>Quote Service</td>
<td>ASO_QUOTE_PUB</td>
<td>PL/SQL</td>
<td>Create_Qoute</td>
</tr>
</tbody>
</table>

To deploy web services do the following:

1. Log into Oracle E-Business Suite as a user with the system administrator role.
2. Navigate to Integrated SOA Gateway, then Integration Repository.
3. For each of the web services listed in the preceding table, perform the following steps if the web service has not already been deployed:
   a. Click Search.
b. Enter the internal name and type, then click Go.
c. In the search results, click the link for the web service.

The web service details are displayed.
d. Click Regenerate WSDL.
e. In the Methods area, select the correct method, and then click Create Grant.
f. On the Create Grants page, enter the user’s name in the format LastName, FirstName.
g. In the search results pop-up menu that appears, click the user’s name.

The page refreshes itself.
h. Click Apply.
i. Repeat the previous steps for other methods as necessary.
j. In the Web Service - Web Service Provider area, select the User name Token check box, and then click Deploy for all web services.

Process of Creating a Quotes Menu

You create a new Quotes menu to display Oracle E-Business Suite quotes in Oracle Sales Cloud, then a new responsibility to allow the new menu to be displayed. Use the following tasks to create a Quotes menu.

Creating the Quotes Menu

You create a quotes menu for the integration by copying the existing Quoting User menu.

1. Log into Oracle E-Business Suite as system administrator.
2. Navigate to Function Developer, Core Services, and then Menus.
3. Search for the menu code QOT_QUOTE_ROOT.

This is the menu code for the Quoting User menu.

4. Click Duplicate.
5. Fill in the fields listed in the following table, and then click Apply.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>QOT_QUOTE_ROOT_INT</td>
</tr>
<tr>
<td>Code</td>
<td>QOT_QUOTE_ROOT_INT</td>
</tr>
<tr>
<td>Description</td>
<td>Quote Root Menu for integration</td>
</tr>
</tbody>
</table>

6. Click Add Another Row.
7. Add the function displayed in the following table, and then click Apply.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Function Name</td>
<td>Quote List</td>
</tr>
</tbody>
</table>
8. Make sure that the function QOT_OAUI_QUOTE_LIST is not on the menu QOT_OAUI_OTHER_MENU by doing the following:
   a. In Menus, search for QOT_OAUI_OTHER_MENU.
   b. If the Quote List function, with code QOT_OAUI_QUOTE_LIST, appears on the list, then remove it.

Creating the Integration Responsibility

Next, you create a responsibility for using the QOT_QUOTE_ROOT_INT menu. The responsibility must be added to all Oracle Sales Cloud users who will use the Oracle E-Business Suite quoting functionality.

To create the integration responsibility for the menu, do the following:

1. Click Navigator, System Administrator, Security, Responsibility, and then Define.
   The form-based Oracle Applications UI appears, with the Responsibilities form active.
2. Click New.
3. Populate the fields with the values from the following table, and then click Save.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Responsibility Name</td>
<td>Integration Responsibility</td>
</tr>
<tr>
<td>Application</td>
<td>Quoting</td>
</tr>
<tr>
<td>Responsibility Key</td>
<td>FND_QUOTE_INT</td>
</tr>
<tr>
<td>Description</td>
<td>Quoting responsibility for integration</td>
</tr>
<tr>
<td>Data Group Name</td>
<td>Standard</td>
</tr>
<tr>
<td>Data Group Application</td>
<td>Quoting</td>
</tr>
<tr>
<td>Menu</td>
<td>QOT_QUOTE_ROOT_INT</td>
</tr>
</tbody>
</table>

Adding the Integration Responsibility to a User

Finally, you add the new responsibility to an existing user. This user will use the responsibility when launching the Quoting application from Oracle Sales Cloud.

To add the integration responsibility to an existing user do the following:

1. In the Navigator - System Administrator window, click the Functions tab.
2. In the Top Ten List, click Users.
3. In the Users form, click Find.
4. In the Users dialog box, search for the menu user, for example, LJONES, then click **OK**.

5. In the Users form, click the **Direct Responsibilities** tab, and then scroll to the end of the list.

6. Click **New**.

   A new row appears in the Direct Responsibilities area.

7. Click the ellipsis in the new row.

   The Responsibilities dialog appears.

8. Search for **Integration Responsibility**, and then click **OK**.

   After logging out, and then logging back in as the menu user, you will see Integration Responsibility in the main menu.
3 Oracle Sales Cloud Configuration

Before You Configure Oracle Sales Cloud

Before configuring Oracle Sales Cloud, you must perform the following tasks in your Oracle E-Business Suite environment.

Related Topics
- Applying Oracle E-Business Suite Patches: Explained
- Creating a Generic Error Page: Explained
- Assigning Roles to the Oracle E-Business Suite User: Explained
- Exposing Web Services: Explained
- Deploying Web Services: Explained

Software Requirements for Oracle Sales Cloud

The integration works with Oracle Sales Cloud, Release 13.

For the list of supported web browsers for Oracle Sales Cloud, see: https://www.oracle.com/system-requirements/index.html.

Related Topics
- System Requirements for Oracle Applications Cloud

Creating Web Service References to Oracle E-Business Suite Web Services

You must create Sales Cloud references to the Oracle E-Business Suite web services required for the integration. This is performed in the Application Composer.

When registering the first web service reference in Oracle Sales Cloud, you must create a new credential key, which Oracle Sales Cloud will use to authenticate the web service calls to Oracle E-Business Suite. Choose the Oracle E-Business Suite user that you created in the Assigning Roles to the Oracle E-Business Suite User task. All web service calls will be executed by authenticating as this user.

Note: You must also grant the Oracle E-Business Suite user explicit access to the web services as done in the Deploying Web Services task, using the Integration Repository user interface in Oracle E-Business Suite.
Creating Web Service References for the Sales Application

You create web service references for the Sales application using Application Composer. Use the following task to create web service references:

1. Click Navigator, and select Application Composer.
2. Click the Create Web Service Reference icon on the Web Service page.
3. For connection type, select SOAP or REST, and then click OK.
4. Enter the name and WSDL for the DQM Search Service web service reference as shown in the following table, and then click Read WSDL.

   ![Note:](image)

   If a warning appears with the following message: "The preferred security options could not be determined. Contact your web service provider or consult your web service documentation to obtain the correct security details. If errors occur during service invocation, contact your system administrator with the web service and error details" click OK to continue.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>O_INT_EBS_DQMSearchService</td>
</tr>
<tr>
<td>WSDL</td>
<td>https://&lt;EBS host;port&gt;/webservices/AppsWSProvider/oracle/apps/ar/hz/service/party/ DqmSearchServicewsdl</td>
</tr>
</tbody>
</table>

5. Click the Add (+) button, then enter the information listed in the following table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credential Key</td>
<td>Enter the credential key using the following format: EBS_user_name_KEY.</td>
</tr>
<tr>
<td>User name</td>
<td>Enter the Oracle E-Business Suite user name.</td>
</tr>
<tr>
<td>Password</td>
<td>Enter the Oracle E-Business Suite password.</td>
</tr>
</tbody>
</table>

6. Select the new credential key from the Credential Key drop-down list.
7. Fill in or confirm the following fields listed in the following table, and then click Save and Close.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service</td>
<td>DqmSearchService_Service.</td>
</tr>
<tr>
<td>Port</td>
<td>DqmSearchService_Port.</td>
</tr>
<tr>
<td>Security Scheme</td>
<td>Call with separate user credentials over SSL.</td>
</tr>
<tr>
<td>Disable time stamp verification</td>
<td>Select the check box.</td>
</tr>
</tbody>
</table>
8. Repeat the previous steps for each of the following web services.

**a.** The following table lists the fields and values required to create the OrganizationService

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>O_INT_EBS_OrganizationService</td>
</tr>
<tr>
<td>WSDL</td>
<td><a href="https://EBS">https://EBS</a> host:port name/ webservices/ AppsWSProvider/ oracle/ apps/ ar/ hz/ service/ party/ OrganizationServicewsdll</td>
</tr>
<tr>
<td>Service</td>
<td>OrganizationService_Service</td>
</tr>
<tr>
<td>Port</td>
<td>OrganizationService_Port</td>
</tr>
<tr>
<td>Security Scheme</td>
<td>Call with separate user credentials over SSL</td>
</tr>
<tr>
<td>Credential Key</td>
<td>EBS_user_name_key</td>
</tr>
<tr>
<td>Disable time stamp verification</td>
<td>Select the check box</td>
</tr>
</tbody>
</table>

**b.** The following table lists the fields and values required to create the Organization Contact Service

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>O_INT_EBS_OrgContactService</td>
</tr>
<tr>
<td>WSDL</td>
<td><a href="https://EBS">https://EBS</a> host:port name/ webservices/ AppsWSProvider/ oracle/ apps/ ar/ hz/ service/ party/ OrgContactServicewsdll</td>
</tr>
<tr>
<td>Service</td>
<td>OrgContactService_Service</td>
</tr>
<tr>
<td>Port</td>
<td>OrgContactService_Port</td>
</tr>
<tr>
<td>Security Scheme</td>
<td>Call with separate user credentials over SSL</td>
</tr>
<tr>
<td>Credential Key</td>
<td>EBS_user_name_key</td>
</tr>
<tr>
<td>Disable time stamp verification</td>
<td>Select the check box</td>
</tr>
</tbody>
</table>

**c.** The following table lists the fields and values required to create the Quote Service

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>O_INT_EBS_QuoteService</td>
</tr>
<tr>
<td>WSDL</td>
<td><a href="https://EBS">https://EBS</a> host:port name/ webservices/ SOAPProvider/ plsql/ asoquote pub/?wsdl</td>
</tr>
</tbody>
</table>
### Oracle Sales Cloud Configuration

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service</td>
<td>ASO_QUOTE_PUB_Service</td>
</tr>
<tr>
<td>Port</td>
<td>ASO_QUOTE_PUB_Port</td>
</tr>
<tr>
<td>Security Scheme</td>
<td>Call with separate user credentials over SSL</td>
</tr>
<tr>
<td>Credential Key</td>
<td>EBS_user_name_key</td>
</tr>
<tr>
<td>Disable time stamp verification</td>
<td>Select the check box</td>
</tr>
</tbody>
</table>

**Related Topics**
- Assigning Roles to the Oracle E-Business Suite User: Explained
- Exposing Web Services: Explained
- Deploying Web Services: Explained

### Using Sandboxes

Sandboxes in Oracle Sales Cloud Applications provide functionality to help isolate and control configuration efforts without impacting other users’ environments or the production environment. Subsequent configuration steps in the integration require you to use sandboxes to test configurations before publishing them. For more information on using sandboxes, see 1484898.1 (Article ID) on My Oracle Support.

### Creating a Sandbox

You create sandboxes from the Administration menu. Use the following task to create a sandbox:

1. Click the **Settings and Actions** drop down list, and select **Manage Sandboxes**.
2. Click the New (+) icon.
3. In the **Create Sandbox** dialog box, name the new sandbox, and then click **Save and Close**.
4. After you see the newly created sandbox in the list, click in the sandbox row (not on the name link), and then click **Set as Active** to activate the new sandbox.
5. Sign out, and then sign back in to start using the sandbox.

### Validating Sandboxes

To validate changes made in a sandbox, test them by using the added Oracle Sales Cloud functionality.

### Publishing Sandboxes

To publish a sandbox do the following:

1. Make sure the sandbox is currently active.
2. Click the sandbox link.
   The **Sandbox Details** window appears.
3. Click the **More** link.
4. Click **Publish**.
5. Sign out, then sign back in.
   The changes in the sandbox are published and can be viewed by all users.

### About Creating Functions

There are two types of functions:

- **Global Functions.** Global functions work in all applications in Oracle Sales Cloud that have extensibility enabled. They are created in Application Composer.

- **Object Functions.** Object functions are attached to a specific object. They are created in the Object Functions tab of the Server Scripts section of the object definition in the Application Composer.

Each function for the integration is defined in a Groovy script file. At the beginning of each file, a set of comments are listed in the following order:

- Function name
- Function return type
- Function parameters (one per line)

Following the comments, the body of the function is listed.

### About Creating Global Functions

The global functions for the integration are defined in the following Groovy script files:

- O_INT_GetLogMsg.groovy
- O_INT_GetSysParam.groovy
- O_INT_AddMultiValueCriteriaItem.groovy
- O_INT_ApplyFilter.groovy
- O_INT_Debug.groovy
- O_INT_Error.groovy
- O_INT_FindRowByKey.groovy
- O_INT_GetLogMsg.groovy
- O_INT_GetRecordCount.groovy
- O_INT_GetRecords.groovy
- O_INT_EBS_GetSysParam.groovy
- O_INT_Info.groovy
- O_INT_Log.groovy
Creating Global Functions

To create a global function, do the following:

1. Click **Navigator**, and select **Application Composer**.
2. In the Overview window, click **Global Functions**.
3. For each Groovy script file do the following:
   a. Select the row, then click the **Action** drop-down list, and select **Add**.
   
   The Create Global Function window appears.
   
   b. In the **Function Name** field, enter the function name from the first comment in the Groovy file.
   c. From the **Returns** drop-down list, select the return type specified in the second comment in the Groovy file.
   d. In the Parameters list, click the **Action** drop-down list, then click the **Add** icon.
   e. Enter the name and type for the first parameter listed in the comments in the Groovy file.
   f. Repeat the steps for the remaining parameters.
   g. Copy the Groovy code from the file into the Function Body window.
   h. Click **Validate** to validate the function.
   i. Click **Save and Close**.

> **Note:** When validating, certain functions will trigger a warning. This is expected; you can click OK to continue. However, if an error is triggered, then contact Oracle Global Customer Support.

Creating Object Functions: Explained

The object functions for the integration are defined in Groovy script files. These files are listed along with the specific objects that must be created or configured. Use the following general procedure to create object functions.

1. Click **Navigator**, and select **Application Composer**.
2. Expand the object for which you want to create a function, and then click **Server Scripts**.
3. In the Server Scripts window, click the **Object Functions** tab.
4. For each Groovy script file do the following:
   a. Click the **Action** drop-down list, and select **Add**.
   
   The Create Object Function window appears.
   
   b. In the **Function Name** field, enter the function name from the first comment in the Groovy file.
   c. From the **Returns** drop-down list, select the return type specified in the second comment in the Groovy file.
   d. In the Parameters list, click the **Add Parameter** icon.
e. Enter the name and type for the first parameter listed in the comments in the Groovy file.

f. Repeat the previous steps for the remaining parameters.

g. Copy the Groovy code from the file into the **Function Body** window.

h. Click **Validate** to validate the function.

\[\text{Note:} \] When validating, certain functions may trigger a warning. This is expected; you can click **OK** to continue. However, if an error is generated, then contact Oracle Global Customer Support.

i. Click **Save and Close**.

---

## Creating Validation Rules

Validation rules are created in the Validation Rules tab of the Server Scripts section of the object definition in the Application Composer.

Validation rules for the Oracle Sales Cloud to Oracle E-Business Suite integration are defined in Groovy script files. At the beginning of each file, a set of comments is listed in the following order:

- Name
- Description
- Error message

Following the comments, the body of the rule is listed. The Groovy files are displayed as rules to be created for specific objects. Use the following general procedure to create validation rules for objects.

1. Click Navigator, and then Tools, and then select **Application Composer**.
2. Expand the object for which you want to create a validation rule, and then click **Server Scripts**.

   In the Server Scripts window, the Validation Rules tab is shown by default.

3. In the Object Rules area, click the **Add a new validation rule** icon.

4. In the Create Object Validation Rule window, in the **Name** field, enter the rule name and description from the comments in the Groovy file.

5. In the **Error Message** field, enter the error message from the comments in the Groovy file.

6. Copy the Groovy code from the file into the **Rule Definition** window.

7. Click **Save and Close**.

---

## Creating Custom Objects

Create a sandbox for each custom object. After creating each custom object, validate the object and publish the sandbox before moving on to the next custom object. Only create a new sandbox after publishing the previous one. Never have two sandboxes active at the same time.

You create custom objects in the Application Composer.

Use the following procedure to create a custom object:

1. Click **Navigator**, and then select **Application Composer**.
2. In the Custom Objects list, click the **Create a New Object** icon.
3. In the Create Custom Object dialog box, enter the object information, and then click OK.

   The custom object appears in the Custom Objects list in the Objects menu.

Adding Fields to a Custom Object

You add fields using the Fields link nested within the custom object in the Application Composer. Use the following procedure to add fields to a custom object.

   1. Expand the custom object’s node in the Custom Objects list in the Objects menu.
   2. Click Fields.
   3. On the Fields page, click the Create a custom field icon.
   4. In the Select Field Type dialog box, select the appropriate field type, and then click OK.
   5. In the Create <Object Type> Field window, enter the parameters, and then click Save and Close.
   6. Repeat the previous steps for other fields as necessary.

Creating Simplified Pages for a Custom Object

After you create a custom object and add fields to it, you create pages expose the new object and fields to users. Every top-level Oracle Sales Cloud object has an overview page, a creation page, and a details page. These pages make up a work area. You use an object’s Simplified Pages tab in the Application Composer to create pages. Use the following procedure to create pages for a custom object.

   1. Expand the custom object’s node in the Custom Objects list in the Objects menu.
   2. Click Pages, and then click the Simplified Pages tab.
   3. On the landing page, click Edit Summary Table then do the following:
      a. In the Configure Summary Table area, add the fields that you want to display in the Summary Table to the Selected Fields list.
      b. In the Configure Summary Table: Buttons and Actions area, make sure Create is in the Selected Buttons list.
      c. Click Save and Close.
   4. In the Creation Page Layouts area, click the Default Custom Layout link, and then do the following:
      a. Click the Edit icon.
      b. In the Configure Detail Form area, add the fields you want to display in the detailed summary.
      c. Click Save and Close.
      d. In the Creation Layout: Standard layout page, click Done.
   5. In the Details Page Layouts area, select click the Standard layout link, then do the following.
      a. On the Details Layout: Standard layout, in the Summary area, click the Edit (pencil) icon.
      b. In the Configure Detail Form area, select the fields you want to display in the default summary.
      c. Click Save and Close.
      d. In the Details Layout: Standard layout page, click Done.

Creating the Integration Configuration Custom Object

You create the Integration Configuration custom object in the Common application. Use a new sandbox to create the Integration Configuration object.
Enter the parameters listed in the following table for the Integration Configuration custom object, using the Creating Custom Objects procedure.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display Label</td>
<td>Integration Configuration.</td>
</tr>
<tr>
<td>Plural Label</td>
<td>Integration Configuration.</td>
</tr>
<tr>
<td>Record Name Label</td>
<td>Integration Configuration ID.</td>
</tr>
<tr>
<td>Record Name Data Type</td>
<td>Select Automatically Generated Sequence from the drop-down list</td>
</tr>
<tr>
<td>Sequence Format</td>
<td>{000000000000000} (15 zeroes inside braces)</td>
</tr>
<tr>
<td>Object Name</td>
<td>O_ INT_ IntegrationConfig</td>
</tr>
<tr>
<td>Description</td>
<td>Leave blank.</td>
</tr>
</tbody>
</table>

Creating Fields for the Integration Configuration Custom Object

Create the fields listed in the following table for the Integration Configuration custom object using the Creating Custom Objects procedure.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Field: Key</th>
<th>Field: RemoteSystemID</th>
<th>Field: Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field Type</td>
<td>Text</td>
<td>Text</td>
<td>Text</td>
</tr>
<tr>
<td>Display Label</td>
<td>Key</td>
<td>RemoteSystem</td>
<td>Value</td>
</tr>
<tr>
<td>Name</td>
<td>Key</td>
<td>RemoteSystemID</td>
<td>Value</td>
</tr>
<tr>
<td>Display Width</td>
<td>20</td>
<td>20</td>
<td>50</td>
</tr>
<tr>
<td>Display Type</td>
<td>Simple Text Box</td>
<td>Simple Text Box</td>
<td>Simple Text Box</td>
</tr>
<tr>
<td>Constraints</td>
<td>Required, Updatable, Searchable, Indexed</td>
<td>Required, Updatable, Searchable, Indexed</td>
<td>Required, Updatable, Searchable</td>
</tr>
<tr>
<td>Maximum Length</td>
<td>100</td>
<td>80</td>
<td>500</td>
</tr>
</tbody>
</table>
Creating Simplified Pages for the Integration Configuration Custom Object

Create the simplified pages listed in the following tables for the Integration Configuration custom object using the Creating Custom Objects procedure.

The following table lists the required information to configure simplified pages for the Integration Configuration custom object.

<table>
<thead>
<tr>
<th>Page Layout</th>
<th>Layout Name</th>
<th>Item</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Landing Page Layouts</td>
<td>Default Custom Layout</td>
<td>Drill Down Column</td>
<td>Configuration ID</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Selected Fields</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Remote System</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Configuration ID</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Key</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Value</td>
</tr>
<tr>
<td>Creation Page Layouts</td>
<td>Default Custom Layout</td>
<td>Selected Fields</td>
<td>Integration Configuration ID</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Remote System</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Key</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Value</td>
</tr>
<tr>
<td>Details Page Layouts</td>
<td>Default Custom Layout</td>
<td>Selected Fields</td>
<td>Integration Configuration ID</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• LastUpdateBy</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• LastUpdateDate</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Remote System</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Key</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Value</td>
</tr>
</tbody>
</table>

Creating a Validation Rule for the Integration Configuration Custom Object

Create the object validation rule from the UniqueKey.groovy file, using the Creating Validation Rules procedure.

Creating Global Functions for the Integration Configuration Custom Object

Create global functions from the following Groovy files, using the Creating Global Functions procedure.

- O_INT_GetIntegConfigParameter.groovy (returns a specific parameter for a category)
- O_INT_GetIntegConfigParameters.groovy (returns all parameters for a category)

**Note:** These global functions depend on the Integration Configuration custom object being created, so they were not created previously in the Creating Global Functions procedure.
Validating and Publishing the Sandbox
Validate and publish the sandbox for the Integration Configuration custom object, using the Using Sandboxes procedures.

Creating the XREF Custom Object
You create the XREF custom object in the Common application. Use a new sandbox to create the XREF custom object.

Creating the Object for the XREF Custom Object
Enter the parameters listed in the following table for the XREF custom object, using the Creating Custom Objects procedure.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display Label</td>
<td>XREF</td>
</tr>
<tr>
<td>Plural Label</td>
<td>XREF</td>
</tr>
<tr>
<td>Record Name Label</td>
<td>XREF ID</td>
</tr>
<tr>
<td>Record Name Data Type</td>
<td>Select Automatically Generated Sequence from the drop-down menu.</td>
</tr>
<tr>
<td>Sequence Format</td>
<td>(000000000000000) (15 zeroes inside braces)</td>
</tr>
<tr>
<td>Object Name</td>
<td>O_INT_XREF</td>
</tr>
<tr>
<td>Description</td>
<td>Leave blank.</td>
</tr>
</tbody>
</table>

Creating Fields for the XREF Custom Object
Create the fields listed in the following table for the XREF custom object using the Creating Custom Objects procedure.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>FusionObjectType</th>
<th>FusionRecordID</th>
<th>RemoteObjectType</th>
<th>RemoteRecordID</th>
<th>RemoteSystemID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field Type</td>
<td>Text</td>
<td>Text</td>
<td>Text</td>
<td>Text</td>
<td>Text</td>
</tr>
<tr>
<td>Display Label</td>
<td>Fusion Object Type</td>
<td>Fusion Record ID</td>
<td>Remote Object Type</td>
<td>Remote Record ID</td>
<td>Remote System</td>
</tr>
<tr>
<td>Name</td>
<td>FusionObjectType</td>
<td>FusionRecordID</td>
<td>RemoteObjectType</td>
<td>RemoteRecordID</td>
<td>RemoteSystemID</td>
</tr>
<tr>
<td>Display Width</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
</tr>
</tbody>
</table>
Creating Global Functions for the XREF Custom Object

Create global functions from the following Groovy files, using the Creating Global Functions procedure.

- O_INT_CreateXREF.groovy
- O_INT_GetAllXREF.groovy
- O_INT_GetXREF.groovy
- O_INT_UpdateXREF.groovy

**Note:** These global functions depend on the XREF custom object being created, so they were not created previously in the Creating Global Functions procedure.

Validating and Publishing the Sandbox

Validate and publish the sandbox for the XREF custom object, using the Using Sandboxes procedures.

Creating a Field Mapping Custom Object

You create the Field Mapping custom object in the Sales application. Use a new sandbox to create the Field mapping custom object.

**Note:** You must validate and publish the sandbox before generating import and export artifacts for the Field Mapping object. The procedure on generating artifacts appears later in this topic.

Creating the Field Mapping Custom Object

Enter the parameters listed in the following table for the Field Mapping custom object, using the Creating Custom Objects procedure.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display Label</td>
<td>Field Mapping</td>
</tr>
<tr>
<td>Plural Label</td>
<td>Field Mappings</td>
</tr>
</tbody>
</table>
Oracle Sales Cloud
Integrating with Oracle E-Business Suite, Release 12.1.3

Chapter 3
Oracle Sales Cloud Configuration

Parameter | Value
---|---
Record Name Label | Field Mapping ID
Record Name Data Type | Select Automatically Generated Sequence from the drop-down menu.
Sequence Format | (000000000000000) (15 zeroes inside braces)
Object Name | O_INT_FieldMapping
Description | Leave blank.

Creating Fields for the Field Mapping Custom Object
Create the fields listed in the following table for the Field Mapping custom object using the Creating Custom Objects procedure.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>FusionFieldNameField</th>
<th>FusionFieldValueField</th>
<th>RemoteFieldIDField</th>
<th>RemoteFieldNameField</th>
<th>RemoteFieldValueField</th>
<th>RemoteSystemIDField</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field Type</td>
<td>Text</td>
<td>Text</td>
<td>Text</td>
<td>Text</td>
<td>Text</td>
<td>Text</td>
</tr>
<tr>
<td>Display Label</td>
<td>Local Field Name</td>
<td>Local Field Value</td>
<td>Remote Field ID</td>
<td>Remote Field Name</td>
<td>Remote Field Value</td>
<td>Remote System Name</td>
</tr>
<tr>
<td>Name</td>
<td>FusionFieldName</td>
<td>LocalFieldValue</td>
<td>RemoteFieldID</td>
<td>RemoteFieldName</td>
<td>RemoteFieldValue</td>
<td>RemoteSystemID</td>
</tr>
<tr>
<td>Display Width</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>Display Type</td>
<td>Simple Text Box</td>
<td>Simple Text Box</td>
<td>Simple Text Box</td>
<td>Simple Text Box</td>
<td>Simple Text Box</td>
<td>Simple Text Box</td>
</tr>
<tr>
<td>Constraints</td>
<td>Required, Updatable, Searchable, Indexed</td>
<td>Required, Updatable, Searchable, Indexed</td>
<td>Updatable, Searchable</td>
<td>Required, Updatable, Searchable, Indexed</td>
<td>Required, Updatable, Searchable, Indexed</td>
<td></td>
</tr>
<tr>
<td>Maximum Length</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>80</td>
</tr>
</tbody>
</table>

Creating Simplified Pages for the Field Mapping Custom Object
Create the simplified pages listed in the following tables for the Field Mapping custom object using the Creating Custom Objects procedure.
Creating the Object Validation Rule for the Field Mapping Custom Object
Create the object validation rule from the UniqueKey.groovy file, using the Creating Validation Rules procedure.

Creating the Global Function for the Field Mapping Custom Object
Create a global function from the O_INT_GetMappedField.groovy file using the Creating Global Functions procedure.

> **Note:** This global function depends on the Field Mapping custom object being created, so it was not created previously in the Creating Global Functions procedure.

Validating and Publishing the Sandbox
Validate and publish the sandbox for the Integration Configuration custom object, using the Using Sandboxes procedures.

Generating Artifacts for Import and Export
To support the import and export of a custom object, such as Field Mapping, you must first generate the object artifacts required for both file-based import and bulk export.

> **Note:** Before performing this step, validate the current sandbox and publish it. Generating import and export artifacts in sandboxes is not supported.

To generate artifacts for import and export do the following:

1. Exit the sandbox, click **Navigator**, and select **Application Composer**.
2. In the Generate Import and Export Artifacts area, click **Generate**.
3. Refresh the page if necessary.
4. In the Details window, make sure that the statuses for the Field Mapping custom object are the following:
   - Overall Status: Completed
   - Import Status: Success
   - Export Status: Success

Creating the EBS Match Child Object
You create EBS Match as a child object of the Account standard object.

> **Note:** You perform the Creating the EBS Match Child Object, and Configuring the EBS Match Child Object procedures in the same sandbox, validate the configurations, and then publish the sandbox.
Creating the Child Object

Use the following procedure to create the EBS Match child object. The procedure for creating a child object is similar to the procedure described in Creating a Custom Object, but you also complete the Child Collection Name field to specify the internal name for the set of child object records.

1. Click **Navigator**, and select **Application Composer**.
2. Expand the Standard Objects node, and then select **Account**.
3. On the Account: Overview page, click the **Create Child Object** button.
4. In the Create Child Object dialog box, enter the object information listed in the following table, and then click OK.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display Label</td>
<td>EBS Match</td>
</tr>
<tr>
<td>Plural Label</td>
<td>EBS Match</td>
</tr>
<tr>
<td>Record Name Label</td>
<td>EBS Match ID</td>
</tr>
<tr>
<td>Record Name Data Type</td>
<td>Select Automatically Generated Sequence from the drop-down menu.</td>
</tr>
<tr>
<td>Sequence Format</td>
<td>(000000000000000) (15 zeroes inside braces)</td>
</tr>
<tr>
<td>Object Name</td>
<td>O_ INT_EBS_Match</td>
</tr>
<tr>
<td>Description</td>
<td>Leave blank</td>
</tr>
<tr>
<td>Child Collection Name</td>
<td>O_ INT_EBS_MatchCollection</td>
</tr>
</tbody>
</table>

Adding Fields to the Child Object

Create the fields listed in the following table for the EBS Match child object using the following procedure.

1. Click **Navigator**, and select **Application Composer**.
2. Click **Sales Object Tags**.
3. From the Objects menu, expand the Standard Objects node, and then select **Account**.
4. Expand the Child Objects node, then click **EBS Match** in the Account: Overview window, and then expand the EBS Match node from the **Custom Objects** list.
5. Click **Fields**.
6. On the Fields page, click the **Create a custom field** icon.
7. In the **Select Field Type** dialog box, select the appropriate field type, and then click **OK**.
8. In the Create Object_Type Field window, enter the parameters, and then click **OK**.
9. Repeat steps 6 through 8 for other fields as necessary.

The following table lists the additional fields required for creating the EBS Match child object.
The following table lists more fields required to create the EBS Match child object.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Selected Field</th>
<th>PartyName Field</th>
<th>PartyID Field</th>
<th>PartyNumber Field</th>
<th>PartyType Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field Type</td>
<td>Text</td>
<td>Text</td>
<td>Text</td>
<td>Text</td>
<td>Text</td>
</tr>
<tr>
<td>Display Label</td>
<td>Address Line 1</td>
<td>City</td>
<td>State</td>
<td>Province</td>
<td>Country</td>
</tr>
<tr>
<td>Name</td>
<td>Address1</td>
<td>City</td>
<td>State</td>
<td>Province</td>
<td>Country</td>
</tr>
<tr>
<td>Display Width</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>Display Type</td>
<td>Simple Text Box</td>
<td>Simple Text Box</td>
<td>Simple Text Box</td>
<td>Simple Text Box</td>
<td>Simple Text Box</td>
</tr>
<tr>
<td>Constraints</td>
<td>Updatable, Searchable</td>
<td>Updatable, Searchable</td>
<td>Updatable, Searchable</td>
<td>Updatable, Searchable</td>
<td>Updatable, Searchable</td>
</tr>
<tr>
<td>Maximum Length</td>
<td>80</td>
<td>80</td>
<td>80</td>
<td>80</td>
<td>80</td>
</tr>
</tbody>
</table>

The following table lists more fields required to create the EBS Match child object.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>PrimaryPhoneCountryCode Field</th>
<th>PrimaryPhoneAreaCode Field</th>
<th>PrimaryPhoneNumber Field</th>
<th>PrimaryEmail Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field Type</td>
<td>Text</td>
<td>Text</td>
<td>Text</td>
<td>Text</td>
</tr>
<tr>
<td>Display Label</td>
<td>Primary Phone Country Code</td>
<td>Primary Phone Area Code</td>
<td>Primary Phone Number</td>
<td>PrimaryEmail</td>
</tr>
</tbody>
</table>

The following table lists more fields required to create the EBS Match child object.
Adding the Constraint Expression for the Selected Check Box

After creating and saving the Selected check box, you edit it to add an expression to the Updatable constraint. To add an expression to the Updatable constraint for the Selected check box do the following:

1. After adding fields to the EBS Match Child object, click the link for the **Selected** check box on the **Fields** page.
2. In the **Edit Check box Field: Selected** window, click the **Expression Builder** in the Constraints area.

   The Expression Builder appears, with the Functions tab displayed by default.
3. In the expression window, enter the following text: `Selected_c == null || Selected_c == 'N'
4. Click **OK**.
5. In the **Edit Check box Field: Selected** window, click **Save and Close**.

Creating the Validation Rule

Create the object validation rule from the UniqueKey.groovy file, using the procedure in Creating Validation Rules.

Creating the Global Function

Create a global function from the O_INT_GetMappedField.groovy file, using the procedure in Creating Global Functions:

> Note: This global function depends on the Field Mapping custom object being created, so it was not created previously in Creating Global Functions.

Validating and Publishing the Sandbox

Validate and publish the sandbox for the XREF custom object, using the procedures in the Using Sandboxes topic.

Generating Artifacts for Import and Export

To support the import and export of a custom object, such as Field Mapping, you must first generate the object artifacts required for both file-based import and bulk export.
Note: Before performing this step, validate the current sandbox and publish it. Generating import and export artifacts in sandboxes is not supported.

To generate artifacts for import and export do the following:

1. Exit the sandbox, click Navigator, and select Application Composer.
2. In the Common Setup area, click Import and Export.
3. In the Generate Import and Export Artifacts area, click Generate.
4. Refresh the page if necessary.
5. In the Details window, make sure that the statuses for the Field Mapping custom object are the following:
   - Overall Status: Completed
   - Import Status: Success
   - Export Status: Success

Configuring Standard Objects

Adding fields to a standard object is very similar to Adding Fields to a Custom Object. To add a field to a standard object do the following:

1. Click Navigator, and then select Application Composer.
2. Select the object, and then click Fields.
3. On the Fields page, click the Create a custom field icon.
4. In the Select Field Type dialog box, select the appropriate field type, and then click OK.
5. In the Create Object_Type Field window, enter the parameters, and then click OK.
6. Repeat the previous steps for other fields as necessary.

Adding Actions and Links to an Object

Use the following procedure to add actions and links to an object.

1. Click Navigator, and then select Application Composer.
2. Select the object, and then click Actions and Links.
3. On the Object_Name: Actions and Links page, click the Create icon.
4. Enter the display label, name, and description.
5. Select the type, Action or Link.
6. Select the source:
   - Script. Select a method name, or click the New icon to enter a new object function.
   - URL. Define a URL expression.
7. Click Save.
8. Repeat the previous steps to add other actions or links as necessary.
Creating Triggers for an Object

Triggers are created in the Triggers tab of the Server Scripts section of the object definition in the Application Composer. Triggers for the integration are generally defined in Groovy script files. At the beginning of each file, a set of comments is listed in the following order:

- Type (object or field)
- Trigger
- Name
- Error message

Following the comments, the trigger definition is listed. The Groovy files are listed after the triggers to be created for specific objects.

Use the following general procedure to create triggers for objects.

1. Click Navigator, and then select Application Composer.
2. Expand the object for which you want to create a trigger, and then click Server Scripts.
3. In the Server Scripts window, click the Triggers tab.
4. Click the Add a new Trigger icon for Object Triggers or Field Triggers.
   - The Create Object Trigger or Create Field Trigger window appears.
5. Select the trigger from the drop-down list, and then enter the name.
   - The trigger and name come from the second and third comments in the Groovy file, respectively.
6. In the Error Message area, enter the error message, if any is provided.
7. Copy or enter the Groovy code from the file into the Trigger Definition window.
8. Click Save and Close.

Configuring the Account Standard Object

Create a new sandbox for configuring the Account standard object, using the Creating Sandboxes topic.

Adding Fields

Add the fields shown in the following table to the Account standard object, using the Adding Fields to a Standard Object procedure in the Configuring Standard Objects topic.

The following table lists fields required to create the Account standard object.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>EBS Sync Status Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field Type</td>
<td>Text</td>
</tr>
<tr>
<td>Display Label</td>
<td>EBS Sync Status</td>
</tr>
<tr>
<td>Name</td>
<td>O_INT_EBS_Sync_Status</td>
</tr>
<tr>
<td>Display Width</td>
<td>Leave blank</td>
</tr>
</tbody>
</table>
Creating Global Functions
Create functions from the following Groovy files, using the Creating Global Functions topic:

- O_INT_EBS_SearchCustomer.groovy
- O_INT_EBS_CreateCustomer.groovy
- O_INT_EBS_GetBaseURL.groovy
- O_INT_EBS_GenerateQuoteURL.groovy

Creating Object Functions
Create functions from the following Groovy files, using the Creating Object Functions topic:

- O_INT_EBS_CheckMatch.groovy
- O_INT_EBS_CleanupMatches.groovy
- O_INT_EBS_CustomerQuoteURL.groovy
- O_INT_EBS_CustomerSync.groovy
- O_INT_EBS_GetSearchCriteria.groovy
- O_INT_EBS_PopulateMatches.groovy

Creating Triggers
Create the following triggers using the Creating Triggers for an Object procedure from the Configuring Standard Objects topic.

- Trigger Type: "Before Insert in Database". Groovy File: O_INT_EBS_CustomerCreateTrigger.groovy file.

Creating the Action
Add the Retry Sync action using information from the following table to the Account object, using the Adding Actions and Links to an Object procedure from the Configuring Standard Objects procedure.
Configuring Simplified Pages

You modify simplified pages on the Simplified Pages tab for the Account object in the Application Composer. To configure simplified pages for the account standard object, do the following:

1. Click Navigator, and then select Application Composer.
2. For the Account object, click Pages, and then click the Simplified Pages tab.
3. In the Details Page Layouts area, select Standard Layout, and then click the Duplicate Layout icon.
4. In the Duplicate Layout dialog box, enter the following name for the duplicate layout: Integration Layout, and make sure that Standard Layout is listed as the Source Layout.
5. Click Save and Edit.
6. In the Details Page Layouts area, click the Standard Layout link.
   a. In the Subtabs Region area, click the Profile icon.
   b. In the Summary area, click the Edit icon (pencil) to view the Configure Detail form.
   c. In the Configure Detail Form, move EBS Sync Status to the Selected Fields list.
   d. Click Save and Close.
7. In the Details Layout: Integration Layout page, click the Edit icon (pencil).
8. In the Available Actions area, move Retry Sync from the Available Actions list to the Selected Actions list.
9. Click Save and Close.
10. In the Subtabs Region area, click the Add icon to add a new subtab.
11. In the Create Subtab page, select Web Content and click Next.
12. In the Create Subtab: Web Content page, enter the information in the following table, and when finished, click Save and Close.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display Label</td>
<td>Quotes</td>
</tr>
<tr>
<td>Source</td>
<td>URL</td>
</tr>
<tr>
<td>URL Definition</td>
<td>O_ INT_ EBS_ CustomerQuoteURL()</td>
</tr>
<tr>
<td>Display Icon</td>
<td>An icon is selected by default. Click Change Icon if needed.</td>
</tr>
</tbody>
</table>
Parameter | Value
---|---
Display Label | EBS Match

Adding the EBS Match Tab

You add the EBS Match tab on the Simplified Pages tab for the Account object in the Application Composer. To add the EBS Match tab, do the following:

1. Click Navigator, and then select Application Composer.
2. Click Pages, and then click the Simplified Pages tab.
3. In the Details Page Layout section, click the Integration Layout that you created in the previous task.
4. In the Subtabs Region area, click the Add icon to add a new subtab.
5. In the Create Subtab page, select Child or related object and click Next.
6. On the Create Subtab: Child or Related Object page, click the Data Object drop-down list and select EBS Match, then enter the information in the following table.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display Label</td>
<td>EBS Match</td>
</tr>
<tr>
<td>Display Icon</td>
<td>An icon is selected by default. Click Change Icon if needed.</td>
</tr>
</tbody>
</table>

7. In the Configure Summary Table area do the following move Selected, Party Name, Match Score, Party Id, Party Type, Address Line 1, EBS Match ID to the Selected Fields list.
8. In the Configure Summary Table: Buttons and Actions area, make sure the Show Edit check box is selected, and deselect the Show Create and Show Delete check boxes.
9. Move Create New EBS Customer to the Selected Buttons list.
10. Validate and publish the sandbox for the Account standard object using the Using Sandboxes topic.

Configuring the EBS Match Child Object

Now you continue configuring the EBS Match child object in Application Composer.

Creating the Object Function

Create an object function from the ForceCreateNewCustomer.groovy file, using the Creating Object Functions topic.

Creating the Validation Rule for the Child Object

Create a validation rule from the O_INT_EBS_MatchSelectRule.groovy file, using the Creating Validation Rules topic.

Creating Triggers for the Child Object

Create triggers from the O_INT_EBS_CleanupMatchesTrigger.groovy file, using the Creating Triggers for an Object procedure from the Configuring Standard Objects topic.
Adding an Action to the Child Object
Add the Create New EBS Customer action shown in the following table to the EBS Match child object, using the Adding Actions and Links to an Object procedure from the Configuring Standard Objects topic.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display Label</td>
<td>Create New EBS Customer</td>
</tr>
<tr>
<td>Name</td>
<td>Create_New_EBS_Customer</td>
</tr>
<tr>
<td>Type</td>
<td>Action</td>
</tr>
<tr>
<td>Description</td>
<td>Leave blank.</td>
</tr>
<tr>
<td>Source</td>
<td>Script</td>
</tr>
<tr>
<td>Method Name</td>
<td>ForceCreateNewCustomer</td>
</tr>
</tbody>
</table>

Validating and Publishing the Sandbox
Validate and publish the sandbox for the EBS Match object, using the Using Sandboxes topic.

Configuring the Opportunity Standard Object
Create a new sandbox for configuring the Opportunity standard object, using the procedure in the Creating Sandboxes topic.

Adding Fields
Add the fields shown in the following table to the Opportunity standard object, using the procedure Adding Fields to a Standard Object from the Configuring Standard Objects topic.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Sales Document Status Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field Type</td>
<td>Text</td>
</tr>
<tr>
<td>Display Label</td>
<td>Sales Document Status</td>
</tr>
<tr>
<td>Display Type</td>
<td>Simple Text Box</td>
</tr>
<tr>
<td>Name</td>
<td>O_INT_SalesDocumentStatus</td>
</tr>
<tr>
<td>Constraints</td>
<td>Updateable (Constraint Expression: false)</td>
</tr>
</tbody>
</table>
Parameter | Sales Document Status Field
---|---
Maximum length | 150
Default Value | None

Creating Global Functions
Create functions from the following Groovy files, using the procedure in the Creating Global Functions topic.

- O_INT_EBS_CreateContact.groovy
- O_INT_EBS_IsMatchedContact.groovy
- O_INT_EBS_SearchContact.groovy

Creating Object Functions
Create functions from the following Groovy files, using the procedure in the Creating Object Functions topic.

- O_INT_EBS_CreateQuote.groovy
- O_INT_EBS_OpptyQuoteURL.groovy
- O_INT_EBS_SearchContactAndCreateXref.groovy
- O_INT_EBS_SyncPrimaryContact.groovy

Creating the Action
Add the Create Quote action shown in the following table to the Opportunity object, using the procedure Adding Actions and Links to an Object from the Configuring Standard Objects topic.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display Label</td>
<td>Create Quote</td>
</tr>
<tr>
<td>Name</td>
<td>Create_Quote</td>
</tr>
<tr>
<td>Type</td>
<td>Action</td>
</tr>
<tr>
<td>Description</td>
<td>Leave blank.</td>
</tr>
<tr>
<td>Source</td>
<td>Script</td>
</tr>
<tr>
<td>Method Name</td>
<td>O_INT_EBS_CreateQuote</td>
</tr>
</tbody>
</table>
Creating Triggers
Create a trigger from the O_INT_CleanUpStatusTrigger.groovy file using the procedure Creating Triggers from the Configuring Standard Objects topic.

Configuring Simplified Pages
You modify simplified pages on the Simplified Pages tab for the Opportunity object in the Application Composer. To configure simplified pages for the Opportunity standard object do the following:

1. Click Navigator, and select Application Composer.
2. Expand the Opportunity node from the Standard Objects list.
3. Click Pages, and then click the Simplified Pages tab.
4. In the Details Page Layouts area, select Standard Layout, and then click the Duplicate Layout icon.
5. In the Duplicate Layout dialog box, enter the following name for the duplicate layout: Integration Layout, and make sure that Standard Layout is listed as the Source Layout.
6. Click Save and Edit.
7. In the Details Layout: Integration Layout page, click the Edit icon (pencil).
8. On the Details Layout: Integration Layout: Buttons and Actions page, move Create Quote from the Available Buttons list to the Selected Buttons list, and then click Save and Close.
9. In the Details Layout: Integration Layout page, in the Subtabs Region area, click the Edit (pencil) icon.
10. Move Sales Document Status from the Available Fields list to the Selected Fields list, and then click Save and Close.
11. In the Configure Detail Form: Buttons and Actions area, move Create Quote to the Selected Buttons area, and then click Save and Close.
12. In the Subtabs Region area, click the Add icon to add a new subtab.
13. In the Create Subtab page, select Web Content and click Next.
14. In the Create Subtab: Web Content page, enter the information in the following table, and when finished, click Save and Close.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display Label</td>
<td>Quotes</td>
</tr>
<tr>
<td>Source</td>
<td>URL</td>
</tr>
<tr>
<td>URL Definition</td>
<td>O_INT_, EBS_, OpptyQuoteURL()</td>
</tr>
<tr>
<td>Display icon</td>
<td>An icon is selected by default. Click Change Icon if needed.</td>
</tr>
</tbody>
</table>

Validating and Publishing the Sandbox
Validate and publish the sandbox for the Opportunity standard object, using the procedures in the Using Sandboxes topic.
Connecting Oracle Sales Cloud to Oracle E-Business Suite

You use the Integration Configuration page to allow Oracle Sales Cloud to exchange data with Oracle E-Business Suite.

Note: The provided reference integration content and the associated Groovy script artifacts use EBS_12.1.3 as the name of the Oracle E-Business Suite instance. You can change this value to your own preferred value (such as EBS_12.2, or myEBSsystem, or any other value) by changing the value of the RemoteSystemID entry in the O_INT_EBS_GetSysParam.groovy file to your preferred value, and by substituting your preferred value with the Remote System values listed in the following table.

To connect Oracle Sales Cloud to Oracle E-Business Suite do the following:

1. Navigate to Tools, and then Integration Configuration.
2. On the Integration Configuration page, click the Create icon.
3. On the Create Integration page, enter the information for the first key-value pair in the following table, and then click Save and Close.
4. Repeat the previous steps for the remaining key-value pairs in the following tables.

<table>
<thead>
<tr>
<th>Remote System</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>EBS_12.1.3</td>
<td>host</td>
<td>Host name of your Oracle E-Business Suite instance, for example: myEBS_server. example.com</td>
</tr>
<tr>
<td>EBS_12.1.3</td>
<td>port</td>
<td>HTTPS port of your Oracle E-Business Suite instance, for example: 443</td>
</tr>
<tr>
<td>EBS_12.1.3</td>
<td>protocol</td>
<td>https</td>
</tr>
<tr>
<td>EBS_12.1.3</td>
<td>TCA_NLS_LANGUAGE</td>
<td>American</td>
</tr>
<tr>
<td>EBS_12.1.3</td>
<td>QOT_NLS_LANGUAGE</td>
<td>American</td>
</tr>
<tr>
<td>EBS_12.1.3</td>
<td>ORG_ID</td>
<td>Organization ID for your Oracle E-Business Suite instance, for example: 204</td>
</tr>
<tr>
<td>EBS_12.1.3</td>
<td>CUST_MATCH_RULE_ID</td>
<td>62</td>
</tr>
<tr>
<td>EBS_12.1.3</td>
<td>PRICE_LIST_ID</td>
<td>Price list ID for your Oracle E-Business Suite instance, which can be found by running the following SQL query: <code>select list_header_id from qp_list_headers_tl where name like '%&lt;Price List Name&gt;%' and language = 'US';</code> When a quote is created from an Oracle Sales Cloud opportunity, an attempt is made to map opportunity revenue line items to quote line items in Oracle E-Business Suite. This saves the user from having to re-create line items manually in the quote.</td>
</tr>
<tr>
<td>Remote System</td>
<td>Key</td>
<td>Value</td>
</tr>
<tr>
<td>--------------</td>
<td>---------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The price list ID is used as the default in the mapping and quote creation process, and drives the pricing that appears in the generated quote in Oracle E-Business Suite.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>After the quote has been generated with the default values, the user can make any necessary changes to the quote, such as the pricing.</td>
</tr>
</tbody>
</table>
4 Postconfiguration Tasks

Mapping Products

For the Oracle Sales Cloud to Oracle E-Business Suite integration to populate new quotes with quote line items that come from the revenue lines on associated opportunities, mappings must be created to associate products from the front office Sales application to the back-office ERP system.

Extracting the List of Oracle E-Business Suite Products

First you use a SQL query to extract the list of products from Oracle E-Business Suite. To extract the list of Oracle E-Business Suite products, do the following:

2. Run the following SQL query:

   ```sql
   SELECT msiv.inventory_item_id, msiv.organization_id, msiv.concatenated_segments item, msiv.description, msiv.primary_uom_code, msiv.serviceable_product_flag, msiv.service_item_flag, msiv.bom_item_type, msiv.item_type, msiv.service_duration, msiv.service_duration_period_code, msiv.shippable_item_flag, msiv.returnable_flag, msiv.web_status, msiv.orderable_on_web_flag, msiv.customer_order_enabled_flag
   FROM mtl_system_items_b_kfv msiv, ibe_ct_imedia_search icis,
   mtl_units_of_measure_vl muv
   WHERE msiv.organization_id = icis.organization_id
   AND msiv.inventory_item_id = icis.inventory_item_id
   AND icis.language = userenv('LANG')
   AND msiv.primary_uom_code <> 'ENR'
   AND (msiv.bom_item_type = 1 OR msiv.bom_item_type = 4)
   AND msiv.primary_uom_code = muv.uom_code
   AND (
       (select fnd_profile.value('ASO_WEB_ATTR_FILTER') from dual) = 'Y'
       AND msiv.web_status = 'PUBLISHED' AND orderable_on_web_flag = 'Y')
   OR
       (select fnd_profile.value('ASO_WEB_ATTR_FILTER') from dual) is
       null OR (select fnd_profile.value('ASO_WEB_ATTR_FILTER') from dual) != 'Y')
   AND icis.category_set_id = nvl((select fnd_profile.value('ASO_CATEGORY_SET') from dual),
   (SELECT category_set_id FROM mtl_default_category_sets WHERE functional_area_id = 7))
   AND msiv.organization_id = <organization_id>;
   
   3. You can use the following optional AND condition to filter results based on specific item codes: AND
   msiv.concatenated_segments in ('<Item name>')
   
   4. Save the results to a CSV file, then open in spreadsheet software.
The item code is listed in the Item column. The item description is listed in the Description column.

Extracting a List of Oracle Sales Cloud Products

Next you extract the list of products from Oracle Sales Cloud. To extract the list of products, do the following:

1. Log into Oracle Sales Cloud as a user with the Product Manager role.
2. Click Navigator, and then click Product Information Management link.
3. Click the Tasks icon, and then in the Item Management area, click the Browse Items link.
4. In the Advanced Search area of the Browse Items page, click the Add Fields button.
5. In the Select and Add: Attributes page, click the Query By Example icon, and then specify a Creation Date value for the attributes.
6. From search result, select the row where Object Item value has the Creation Date Attribute value you entered, and click Add and then click OK.
   Search criteria for the Creation Date field is added.
7. From the Creation Date drop-down menu, select After.
8. Select a date that is at least five years ago (for example, 1/1/09).
9. Click the Search button. The list of items appears. If required, you can filter by Item and Organization and Item.
10. Click the Export to Excel icon.
11. Save the file.

Note: Some spreadsheet tools remove the lead zero in a numeric field. This can affect the item numbers if the numbers are only numeric. Use a basic text editor or verify your item numbers to ensure that they are correctly reflected.

For more information on Oracle Sales Cloud products and creating sales catalogs, see Related Topics.

Related Topics
- Oracle Sales Cloud Library

Mapping Products Manually

You can create product mappings by comparing the SQL results from Oracle E-Business Suite to the CSV file from Oracle Sales Cloud. Here is one example for mapping desktop computers. To map products manually, do the following:

1. In the Oracle E-Business Suite SQL results, filter on descriptions containing Desktop.
2. Note the values in the ITEM and INVENTORY_ITEM_ID columns.
3. In the Oracle Sales Cloud Excel file, filter on descriptions containing Desktop.
4. Note the values in the Item column.
5. Map the corresponding products to each other.
   The following table shows an example:

<table>
<thead>
<tr>
<th>System</th>
<th>Column</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Sales Cloud</td>
<td>Description</td>
<td>Sentinel 9000 Desktop</td>
</tr>
</tbody>
</table>
Entering Mappings into Oracle Sales Cloud

You use the Field Mapping object in Oracle Sales Cloud to enter product mappings. The example in the following table uses the values from the example in the Mapping Products to Each Other Manually topic.

> **Note:** You can also import the product mappings using Oracle Sales Cloud File Import function.

To enter product mappings into Oracle Sales Cloud do the following:

1. Log into Oracle Sales Cloud as a user with the Administrator role.
2. Navigate to **Tools**, and then **Field Mapping**.
3. On the Field Mappings page, click the **Create** icon.
4. Enter the values shown in the following table, then click **Save and Close**.
5. Repeat steps 3 and 4 for other products.

The following steps and 4 for other products.

The following tables lists sample mapping values.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Remote System Name</td>
<td>Remote system you entered in Connecting Oracle Sales Cloud to Oracle E-Business Suite</td>
</tr>
<tr>
<td>Local Field Name</td>
<td>Product</td>
</tr>
<tr>
<td>Local Field Value</td>
<td>Reference Number column value in Oracle Sales Cloud, for example, AS54600</td>
</tr>
<tr>
<td>Remote Field Name</td>
<td>Product</td>
</tr>
<tr>
<td>Remote Field Value</td>
<td>ITEM column value in Oracle E-Business Suite, for example, AS18947</td>
</tr>
<tr>
<td>Remote Field ID</td>
<td>INVENTORY_ITEM_ID column value in Oracle E-Business Suite, for example, 155</td>
</tr>
</tbody>
</table>
Mapping Product Groups: Explained

Use the following tasks to create product group mappings from Oracle E-Business Suite to Oracle Sales Cloud.

Extracting the List of Oracle E-Business Suite Products

First you use a SQL query to extract the list of products from Oracle E-Business Suite. To extract the list of Oracle E-Business Suite products.

2. Run the following SQL query:

   ```sql
   SELECT msiv.inventory_item_id,
   msiv.organization_id,
   msiv.concatenated_segments item
   ,msiv.description
   ,msiv.primary_uom_code
   ,msiv.serviceable_product_flag
   ,msiv.service_item_flag
   ,msiv.bom_item_type
   ,msiv.item_type
   ,msiv.service_duration_period_code
   ,msiv.shippable_item_flag
   ,msiv.returnable_flag
   ,msiv.web_status
   ,msiv.orderable_on_web_flag
   ,msiv.customer_order_enabled_flag
   FROM mtl_system_items_b_kfv msiv,ibe_ct_imedia_search icis,
    mtl_units_of_measure_vl muv
   WHERE msiv.organization_id = icis.organization_id
   AND msiv.inventory_item_id = icis.inventory_item_id
   AND icis.language = userenv('LANG')
   AND msiv.primary_uom_code <> 'ENR'
   AND (msiv.bom_item_type = 1 OR msiv.bom_item_type = 4)
   AND msiv.primary_uom_code = muv.uom_code
   AND (select fnd_profile.value('ASO_WEB_ATTR_FILTER') from dual) = 'Y'
   OR
   (select fnd_profile.value('ASO_WEB_ATTR_FILTER') from dual) is null
   OR (select fnd_profile.value('ASO_WEB_ATTR_FILTER') from dual) != 'Y')
   AND icis.category_set_id = nvl((select
   fnd_profile.value('ASO_CATEGORY_SET') from dual), (SELECT
   category_set_id FROM mtl_default_category_sets WHERE functional_area_id =7))
   AND msiv.organization_id = <organization_id>;
   ```

3. You can use the following optional AND condition to filter results based on specific item codes:

   ```sql
   AND msiv.concatenated_segments in <'Item_name')
   ```

4. Save the results to a CSV file, then open in spreadsheet software.
Extracting the List of Oracle Sales Cloud Product Groups

Use this task to extract a list of product groups from Oracle Sales Cloud.

1. In Oracle Sales Cloud navigate to **Setup and Maintenance**.

   Oracle Functional Setup Manager appears.

2. From the Setup drop-down menu, select **Sales**.

3. In the Search Tasks box, search for **Manage Product Groups**.

4. Click the **Manage Product Groups** link in the search results.

5. In the Sales Catalog and Products task list, click the **Manage Product Groups** link.

6. In the Manage Product Groups area, click the **View** drop-down list, and then click **Columns**, and from that Columns menu click the following columns to enable them:
   - Name
   - Reference Number
   - Active
   - Effective from Data
   - Effective to Date

7. In the Manage Product Groups area click the **Export to Excel** icon to export the list.

Mapping Products to Each Other Manually

You can create product mappings by comparing the SQL results from Oracle E-Business Suite to the CSV file from Oracle Sales Cloud. The following is one example for mapping desktop computers. To map products manually, do the following:

1. Using the Oracle E-Business Suite SQL results generated in Extracting the List of Oracle E-Business Suite Products, filter on descriptions containing **Desktop**.

2. Note the values in the ITEM and INVENTORY_ITEM_ID columns.

3. In the Oracle Sales Cloud Excel file, filter on descriptions containing Desktop.

4. Note the values in the Reference Number column.

5. Map the corresponding products to each other.

<table>
<thead>
<tr>
<th>System</th>
<th>Column</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Sales Cloud</td>
<td>Product Group Name</td>
<td>WAVE Desktops</td>
</tr>
<tr>
<td></td>
<td>Product Group Reference Number</td>
<td>3-57FC-880</td>
</tr>
<tr>
<td>Oracle E-Business Suite</td>
<td>DESCRIPTION</td>
<td>Sentinel Deluxe Desktop</td>
</tr>
<tr>
<td></td>
<td>Reference Number</td>
<td>AS18947</td>
</tr>
<tr>
<td></td>
<td>INVENTORY_ITEM_ID</td>
<td>155</td>
</tr>
</tbody>
</table>
Entering Mappings into Oracle Sales Cloud
You use the Field Mapping object in Oracle Sales Cloud to enter product mappings. The example listed in the following table uses the values from the example in Mapping Products to Each Other Manually.

Note: You can also import the product mappings with the Oracle Sales Cloud File Import function. To enter product mappings into Oracle Sales Cloud, do the following:

1. Log into Oracle Sales Cloud as a user with the Administrator role.
2. Navigate to Tools, and then Field Mapping.
3. On the Field Mappings page, click the Create icon.
4. Enter the values shown in the following table, then click Save and Close.
5. Repeat steps 3 and 4 for other products.
6. Click Save and Close when finished.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Remote System Name</td>
<td>Remote system you entered in Connecting Oracle Sales Cloud to Oracle E-Business Suite</td>
</tr>
<tr>
<td>Local Field Name</td>
<td>Product Group</td>
</tr>
<tr>
<td>Local Field Value</td>
<td>Reference Number column value in Oracle Sales Cloud, for example, AS54600</td>
</tr>
<tr>
<td>Remote Field Name</td>
<td>Product</td>
</tr>
<tr>
<td>Remote Field Value</td>
<td>ITEM column value in Oracle E-Business Suite, for example, AS18947</td>
</tr>
<tr>
<td>Remote Field ID</td>
<td>INVENTORY_ITEM_ID column value in Oracle E-Business Suite, for example, 155</td>
</tr>
</tbody>
</table>

About Hiding Oracle E-Business Suite UI Elements for Streamlined Navigation
This integration relies on a technique known as a UI mashup. The Oracle E-Business Suite quote user interface (UI) is shown within an Oracle Sales Cloud UI by embedding it inside an HTML inline frame (iFrame). This means that the user will see both the Oracle Sales Cloud UI and the Oracle E-Business Suite UI at the same time in his or her browser.

Navigation Issues with UI Mashups
When combining two applications into one UI, however, certain navigational issues can result. While UI mashups are generally designed to show related content from the two applications (for example, a quote that is related to an opportunity or a customer), it is possible to use the navigational controls in the embedded application to go to a part of the embedded application that has no relationship to the parent application.

In the case of embedding an Oracle E-Business Suite UI inside of an Oracle Sales Cloud UI, it is theoretically possible to use the Oracle E-Business Suite navigator to go to a completely different module within the Oracle E-Business Suite application.
that has nothing to do with the related customer or opportunity from Oracle Sales Cloud, where the embedded UI was being displayed. For example, users with privileges in Oracle E-Business Suite that allow them to create quotes might also be able to navigate to Oracle E-Business Suite Human Resources or Financials pages that have no relationship to the customer or opportunity in Oracle Sales Cloud.

### Limiting Unwanted Navigation

To minimize the possibility that end users can navigate and change the context of the embedded Oracle E-Business Suite UI to something that is not related to the parent Oracle Sales Cloud UI, the integration relies on a specialized patch in Oracle E-Business Suite that hides the header and footer sections in the quote UI pages, which the user would normally see if logged directly in to Oracle E-Business Suite. This partially prevents users from navigating to other modules within Oracle E-Business Suite. However, this does not completely eliminate the possibility of navigating to something out of context. Within the Oracle E-Business Suite quote UI there are also a variety of hyperlinks and buttons that allow the user to navigate to content and modules that are outside of quoting, and therefore outside the context of the Oracle Sales Cloud-Oracle E-Business Suite quote UI mashup.

### Hiding or Disabling Links and Controls

To help minimize this issue, use the following table which shows links and controls that can be hidden or disabled. Note that disabling these UI elements is not required for the integration to work. Instead, it is left to implementers to determine whether limiting navigational options is important for their user communities. In some cases, user training might be sufficient to avoid any navigational confusion. In other cases, limiting what users can actually navigate to or drill down on will be desirable.

If limiting navigational options is required, the following table shows a number of controls that might be helpful to hide or disable. Note that additional controls not listed might need to be hidden or disabled, depending on what customizations have been applied to the Oracle E-Business Suite environment. This list is a starting point, and is not comprehensive.

The following tables list the Oracle E-Business Suite UI Elements to hide on the Oracle Sales Cloud Quote Update page.

The following table lists the Summary Area and the UI element within that area that must be changed.

<table>
<thead>
<tr>
<th>Application Area</th>
<th>Application Subarea</th>
<th>UI Element</th>
<th>Property Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Summary</td>
<td>Not applicable</td>
<td>Select button beside Contact Name</td>
<td>Rendered: false</td>
</tr>
</tbody>
</table>

The following table lists the Product tab area and the UI element within that area that must be changed.

<table>
<thead>
<tr>
<th>Application Area</th>
<th>Application Subarea</th>
<th>UI Element</th>
<th>Property Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product tab</td>
<td>Not applicable</td>
<td>Submit button: Add Service to Previous Purchase</td>
<td>Rendered: false</td>
</tr>
</tbody>
</table>

The following table lists the Pricing tab area and the UI element within that area that must be changed.

<table>
<thead>
<tr>
<th>Application Area</th>
<th>Application Subarea</th>
<th>UI Element</th>
<th>Property Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pricing tab</td>
<td>Not applicable</td>
<td>Add Adjustment in Product tab</td>
<td>Rendered: false</td>
</tr>
<tr>
<td>Application Area</td>
<td>Application Subarea</td>
<td>UI Element</td>
<td>Property Change</td>
</tr>
<tr>
<td>------------------</td>
<td>---------------------</td>
<td>------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Column: (QotHdrPrcAdjRN. QotHdrPrcAdjRmCol) Add Promotion in Product tab</td>
<td>Rendered: false</td>
</tr>
<tr>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Add Promotion in Product tab</td>
<td>Rendered: false</td>
</tr>
<tr>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Column: (QotHdrPrcPromCdRN. QotHdrPromoRmCol)</td>
<td>Rendered: false</td>
</tr>
<tr>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Add Charge in Product tab</td>
<td>Rendered: false</td>
</tr>
<tr>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Column: (QotHdrPrcChrgRN. QotHdrChrgRmCol)</td>
<td>Rendered: false</td>
</tr>
<tr>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Reset Pricing button</td>
<td>Rendered: false</td>
</tr>
<tr>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Message Text Input: Pricing Date</td>
<td>Read only: true</td>
</tr>
<tr>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Message Lov Input: Product Fiscal Classification</td>
<td>Read only: true</td>
</tr>
<tr>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Message Lov Input: Transaction Business Category</td>
<td>Read only: true</td>
</tr>
<tr>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Message Choice: Tax Handling</td>
<td>Read only: true</td>
</tr>
<tr>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Message Choice: Exemption Reason</td>
<td>Read only: true</td>
</tr>
<tr>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Message Text Input: Exemption Number</td>
<td>Read only: true</td>
</tr>
</tbody>
</table>

The following table lists the Shipping and Billing tab area and the UI element within that area that must be changed.

<table>
<thead>
<tr>
<th>Application Area</th>
<th>Application Subarea</th>
<th>UI Element</th>
<th>Property Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shipping and Billing tab</td>
<td>Address</td>
<td>Go button after Message Choice: Copy the Quote’s Customer, Contact and Address To</td>
<td>Rendered: false</td>
</tr>
<tr>
<td>Not applicable</td>
<td>Address</td>
<td>Message Choice: Copy the Quote’s Customer, Contact and Address To</td>
<td>Rendered: false</td>
</tr>
<tr>
<td>Not applicable</td>
<td>Address, Shipping</td>
<td>• Select button beside Costumer Name</td>
<td>Rendered: false</td>
</tr>
</tbody>
</table>
### Postconfiguration Tasks

The following table lists the Product tab area and the UI element within that area that must be changed.

<table>
<thead>
<tr>
<th>Application Area</th>
<th>Application Subarea</th>
<th>UI Element</th>
<th>Property Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sales Team tab</td>
<td>Team Members</td>
<td>Add Person button</td>
<td>Rendered: false</td>
</tr>
</tbody>
</table>

The table includes the following changes:

- **Select button beside Contact name**
- **Select button beside Address**
- **Message Choice: Request Date Type**
- **Message Choice: Shipping Method**
- **Message Choice: Shipment Priority**
- **Message Choice: Demand Class**
- **Message Choice: Freight Terms**
- **Message Choice: Free On Board (FOB)**
- **Message Choice: Packing Instructions**
<table>
<thead>
<tr>
<th>Application Area</th>
<th>Application Subarea</th>
<th>UI Element</th>
<th>Property Change</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Reassign Sales Team button</td>
<td>Rendered: false</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Flow Layout:</td>
<td>Rendered: false</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(QotHdrSlsTrmTabRN,</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>QotHdrSlsTrmActRN)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Message Choice: Role</td>
<td>Read only: true</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Message Check Box: Do Not</td>
<td>Read only: true</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Reassign</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Remove image in the table in Remove column</td>
<td>Rendered: false</td>
</tr>
<tr>
<td>Sales Credit, Revenue Credits</td>
<td>Add Person button</td>
<td>Rendered: false</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Remove image in the table in the</td>
<td>Rendered: false</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Remove column</td>
<td>Remove column</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Message Choice: Sales Group</td>
<td>Read only: true</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Message Choice: Credit Type</td>
<td>Read only: true</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Message Text Input: Credit Percent</td>
<td>Read only: true</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Column: (QotHdrSlsCrdtRN,</td>
<td>Rendered: false</td>
<td></td>
</tr>
<tr>
<td></td>
<td>QotRvnRemCol)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sales Credit, Non-Revenue Credits</td>
<td>Add Person button</td>
<td>Rendered: false</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Remove image in the table in the</td>
<td>Rendered: false</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Remove column</td>
<td>Remove column</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Message Choice: Sales Group</td>
<td>Read only: true</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Message Choice: Credit Type</td>
<td>Read only: true</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Message Text Input: Credit Percent</td>
<td>Read only: true</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Column: (QotHdrSlsCrdtRN,</td>
<td>Rendered: false</td>
<td></td>
</tr>
<tr>
<td></td>
<td>QotNonRvnRemCol)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The following table lists the Proposals area and the UI element within that area that must be changed.
Hiding or Disabling Links and Controls Continued

The following table lists the Approvals area and the UI element within that area that must be changed.

<table>
<thead>
<tr>
<th>Application Area</th>
<th>Application Subarea</th>
<th>UI Element</th>
<th>Property Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Approvals</td>
<td>Not applicable</td>
<td>Initiate Approvals button</td>
<td>Rendered: false</td>
</tr>
<tr>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Cancel Approval Process button</td>
<td>Rendered: false</td>
</tr>
<tr>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Table Actions</td>
<td>Rendered: false</td>
</tr>
</tbody>
</table>

The following table lists the Contract Terms area and the UI element within that area that must be changed.

<table>
<thead>
<tr>
<th>Application Area</th>
<th>Application Subarea</th>
<th>UI Element</th>
<th>Property Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contract Terms</td>
<td>Not applicable</td>
<td>Preview Terms button</td>
<td>Rendered: false</td>
</tr>
<tr>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Update Terms button</td>
<td>Rendered: false</td>
</tr>
<tr>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Message Text Input: Signed By</td>
<td>Read only: true</td>
</tr>
<tr>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Message Text Input: Signed Date</td>
<td>Read only: true</td>
</tr>
</tbody>
</table>

The following table lists the Supplement area and the UI element within that area that must be changed.

<table>
<thead>
<tr>
<th>Application Area</th>
<th>Application Subarea</th>
<th>UI Element</th>
<th>Property Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supplement</td>
<td>Not applicable</td>
<td>Update button</td>
<td>Rendered: false</td>
</tr>
<tr>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Row Layout: (QotHdrSuppTabRN, QotSupUpdateRowRN)</td>
<td>Rendered: false</td>
</tr>
<tr>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Remove button</td>
<td>Rendered: false</td>
</tr>
<tr>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Message Choice: Section</td>
<td>Read only: true</td>
</tr>
<tr>
<td>Not applicable</td>
<td>Not applicable</td>
<td>Message Choice: Template</td>
<td>Read only: true</td>
</tr>
</tbody>
</table>
The following table lists the Notes and Tasks area and the UI element within that area that must be changed.

<table>
<thead>
<tr>
<th>Application Area</th>
<th>Application Subarea</th>
<th>UI Element</th>
<th>Property Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Notes and Tasks</td>
<td>Tasks</td>
<td>Flexible Content: Tasks</td>
<td>Rendered: false</td>
</tr>
<tr>
<td>Not applicable</td>
<td>Attachments</td>
<td>Flexible Content: Attachments</td>
<td>Rendered: false</td>
</tr>
</tbody>
</table>

**Hiding Oracle E-Business Suite UI Elements**

There are two major steps required to hide Oracle E-Business Suite UI elements when accessing quotes from Oracle Sales Cloud:

- Hiding UI Elements
- Hiding the Integration Responsibility

**Hiding UI Elements**

You use personalization to hide Oracle E-Business Suite UI elements.

**Hiding UI Elements on the Quote Creation Page**

There are two select buttons to hide on the Quote Creation page. The Customer Name select button and the Contact Name select button. To hide these UI elements on the Quote Creation page do the following:

1. Log in as the EBS Integration user and navigate to **Integration Responsibility**, **Quote**, and then **Create Quote**.
2. In the **Quote Creation** page, click **Personalize Page**.
   - The personalization page organizes all of the controls in a tree structure.
3. Make sure that the **Personalization Level** is set to **Responsibility: Integration Responsibility**.
4. Click the pencil icon beside the **Select** button for Customer Name.
5. Change the **Rendered** property to **false**.
6. Repeat the steps for **Contact Name**.

**Hiding UI Elements on the Quote Update Page**

UI elements that can be hidden on the Quote Update page using personalization. To hide UI elements on the Quote Update page do the following:

1. Navigate to the **Quote List** page.
2. Click any quote hyperlink to navigate to the **Quote Update** page.
3. Click **Personalize Page**.
   - The personalization page organizes all of the controls in a tree structure.
4. Make sure that the **Personalization Level** is set to **Responsibility: Integration Responsibility**.
5. Hide any of the UI elements that you want to hide, using the procedure in Hiding UI Elements on the Quote Creation Page.
Hiding UI Elements on the Add Services Page

You hide UI elements to make the Add Service page read only. To make the Add Services page read only do the following:

1. On the Product tab on the Quote Update page, select a product.
2. Select Add Services from the drop-down list, then click Go.
3. Click Personalize Page.
4. Make sure that the Personalization Level is set to Responsibility: Integration Responsibility.
5. Set the Rendered property to false for the following UI elements:
   - Select button
   - Multiple Selection: (QotAddSvcSel)

Hiding the Integration Responsibility

After hiding the UI elements, you can hide the integration responsibility. First you must remove the Quotes menu from the integration responsibility. To remove the Quotes Menu Globally from Integration Responsibility do the following:

1. Log in as System Administrator.
2. Navigate to Function Developer, Core Services, and then Menus.
3. Search for the QOT_QUOTE_ROOT_INT menu code.
4. Click Update.
5. Delete the word Quotes from the Prompts column.
6. Deselect the Grant check box in the same row, then click Apply. The QOT_QUOTE_ROOT_INT menu is removed from Integration Responsibility.

Removing Integration Responsibility from the Navigator

Next you can optionally remove the integration responsibility from the Navigator. To remove Integration Responsibility from the Navigator do the following:

1. Click Personalize in the Navigator menu on the home page.
2. In the Search and Select area, click Go.
3. Select Integration Responsibility from the search results, then click Hide.
4. Click Apply.

Testing the Integration: Explained

Test the integration by navigating to an Oracle Sales Cloud opportunity, confirming that the Oracle E-Business Suite quote UI appears, and then clicking Create Quote to confirm that the quote appears in the quote UI.

Configuring the Integration with the Security Console

You can use the Security Console to perform unique configurations to your integration, such as restricting access to specific users are particularly administration roles to the Customer Matching UI, and so on. Perform the following tasks in Oracle Sales Cloud after you have successfully imported Oracle E-Business Suite items. To create and use a custom role do the following:

1. Click Navigator, and select Security Console.
2. Click the **Create Role** button, and add the information contained in the following table.

**Note:** When assigning a role, consider the following: assigning a role to a specific matching user allows individual users who have been assigned that role to perform customer matching. Assigning the role to an existing role hierarchy allows the group of users contained within the role hierarchy to perform customer matching.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Role Name</td>
<td>Customer Matching Admin</td>
</tr>
<tr>
<td>Role Code</td>
<td>INT_Customer_Matching_Admin_Role</td>
</tr>
<tr>
<td>Role Category</td>
<td>CRM - Job Roles</td>
</tr>
<tr>
<td>Description</td>
<td>Custom role for granting the access to perform customer matching.</td>
</tr>
</tbody>
</table>

After you have created the Customer Matching Admin role using Security Console, the role is exposed in Application Composer. You can then assign privileges which allow the role to view custom objects. To assign privileges, see the following procedure.

**Defining Security Policies for a Custom Object**

To define security policies for a custom object, do the following:

1. Click **Navigator**, and select **Application Composer**.
2. From the **Application** drop-down list, select the relevant application.
3. Expand the Custom Objects node, and then expand the relevant object.
4. Click **Security**, and then in the **Define Policies: Object_name** page, select the appropriate check boxes for the privileges you wish the role to have for the object, and then click **Save and Close**.
5 Required Files

Required HTML File: Explained

The following HTML file is required for the integration:
O_INT_EBS_ErrorPage.html

Required Groovy Files: Explained

The following Groovy script files are required for global functions used in the integration. Refer to the Related Topics section in this topic for a link to Integrating Oracle Sales Cloud with Oracle E-Business Suite (Doc ID 1643818.1) on My Oracle Support. The files are located in the Attachments section of the article.

Global Function Groovy Files

The following Groovy script files are required for global functions used in the integration.

- O_INT_GetLogMsg.groovy
- O_INT_GetSysParam.groovy
- O_INT_AddMultiValueCriterionItem.groovy
- O_INT_ApplyFilter.groovy
- O_INT_Debug.groovy
- O_INT_Error.groovy
- O_INT_FindRowByKey.groovy
- O_INT_GetLogMsg.groovy
- O_INT_GetRecordCount.groovy
- O_INT_GetRecords.groovy
- O_INT_EBS_GetSysParam.groovy
- O_INT_Info.groovy
- O_INT_Log.groovy
- O_INT_LogMessage.groovy
- O_INT_Warn.groovy

Integration Configuration Custom Object Groovy Files

The following Groovy script files are required for the Integration Configuration custom object.

Validation Rule

UniqueKey.groovy
Chapter 5

Required Files

Global Functions
- O_INT_GetIntegConfigParameter.groovy
- O_INT_GetIntegConfigParameters.groovy

XREF Custom Object
The following Groovy script files are required for the XREF custom object.

Global Functions
- O_INT_CreateXREF.groovy
- O_INT_GetAllXREF.groovy
- O_INTGetXREF.groovy
- O_INT_UpdateXREF.groovy

Field Mapping Custom Object
The following Groovy script files are required for the Field Mapping custom object.

Validation Rule
UniqueKey.groovy

Global Function
- O_INT_GetMappedField.groovy
- O_INT_EBS_GetMappedField.groovy

EBS Match Child Object
The following Groovy script files are required for the EBS Match Child object.

Object Function
ForceCreateNewCustomer.groovy

Object Validation
O_INT_EBS_MatchSelectRule.groovy

Triggers
O_INT_CleanupMatchesTrigger.groovy

Account Standard Object
The following Groovy script files are required for the Account standard object.

Global Functions
- O_INT_EBS_CreateCustomer.groovy
- O_INT_EBS_GenerateQuoteURL.groovy
- O_INT_EBS_GetBaseURL.groovy
Required Files

Object Functions

• O_INT_EBS_SearchCustomer.groovy
• O_INT_EBS_CheckMatch.groovy
• O_INT_EBS_CleanupMatches.groovy
• O_INT_EBS_CustomerQuoteURL.groovy
• O_INT_EBS_CustomerSync.groovy
• O_INT_EBS_GetSearchCriteria.groovy
• O_INT_EBS_PopulateMatches.groovy

Triggers

• O_INT_EBS_CustomerSyncTrigger.groovy
• O_INT_EBS_CustomerCreateTrigger.groovy

Opportunity Standard Object

The following Groovy script files are required for the Opportunity standard object.

Global Functions

• O_INT_EBS_CreateContact.groovy
• O_INT_EBS_IsMatchedContact.groovy
• O_INT_EBS_SearchContact.groovy

Object Functions

• O_INT_EBS_CreateQuote.groovy
• O_INT_EBS_OpptyQuoteURL.groovy
• O_INT_EBS_SearchContactAndCreateXref.groovy
• O_INT_EBS_SyncPrimaryContact.groovy

Triggers

O_INT_CleanUpStatusTrigger.groovy

Related Topics

• Integration Implementation Files on My Oracle Support.