Oracle

Sales Cloud

Using Customer Data Management

Release 12
Contents

Preface

1 About this Guide
   Audience and Scope
   Related Guides

2 Using File-Based Import
   Overview of File-Based Import
   Import Consumers Using File-Based Import
   Import Contacts Using File-Based Import
   Import Country Structures Using File-Based Import
   Import Customer Hierarchies Using File-Based Import
   Import Customers Using File-Based Import
   Import Employee Resources Using File-Based Import
   Import Geographies Using File-Based Import
   Import Households Using File-Based Import
   Import Legal Entities Using File-Based Import
   Import Resource Teams Using File-Based Import
   Import Common Supporting Objects

3 Using File-Based Export
   Overview of File-Based Data Export

4 Using Round-Trip Export-Import
   Exporting and Importing Data Between Oracle Sales Cloud Instances Using Automatic Mapping: Worked Example
   Exporting and Importing Data Between Oracle Sales Cloud Instances Using Automatic Mapping: Procedure

5 Importing Accounts
   Importing Common Supporting Objects: Documentation Overview
   Importing Accounts Using File-Based Import: Explained
   Account Import Objects: How They Work Together
   Importing Sales Profiles Using File-Based Import: Quick Start
## 6 Importing Contacts

- Importing Contacts and Consumers: Important Considerations: 183
- Importing Common Supporting Objects: Documentation Overview: 183
- Importing Contacts Using File-Based Import: Explained: 183
- Contact Import Objects: How They Work Together: 185
- Importing Person Profiles Using File-Based Import: Quick Start: 188
- Importing Contact Points Using File-Based Import: Quick Start: 194
- Importing Contact Point Preferences Using File-Based Import: Quick Start: 203
- Importing Contact Jobs Using File-Based Import: Quick Start: 212

## 7 Importing Country Structures

- Importing Country Structures Using File-Based Import: Explained: 217
- Country Structure Import Objects: How They Work Together: 218
- Importing Country Structures Using File-Based Import: Quick Start: 220

## 8 Importing Customer Hierarchies

- Importing Customer Hierarchy Using File-Based Import: Explained: 223
- Customer Hierarchy Import Objects: How They Work Together: 224
- Importing Customer Hierarchies Using File-Based Import: Quick Start: 226
- Importing Customer Hierarchy Member Using File-Based Import: Quick Start: 229
- Deleting Sales Account Hierarchies Using File-Based Data Import: Procedure: 234

## 9 Importing Geographies

- Importing Geographies Using File-Based Import: Explained: 239
- Geography Import Objects: How They Work Together: 240
- Importing Geographies Using File-Based Data Import: Worked Example: 242
- Importing Country Structures Using File-Based Import: Quick Start: 245
- Importing Geography Hierarchies Using File-Based Import: Quick Start: 247
- Importing Territory Geographies Using File-Based Data Import: Explained: 251
- Territory Geography Import Objects: How They Work Together: 253
- Importing Territory Geography Hierarchies Using File-Based Data Import: Quick Start: 254

## 10 Importing Households

- Importing Common Supporting Objects: Documentation Overview: 261
- Importing Households Using File-Based Import: Explained: 261
- Household Import Objects: How They Work Together: 263
- Importing Group Profiles Using File-Based Import: Quick Start: 266
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>Importing Source System References</td>
<td>273</td>
</tr>
<tr>
<td></td>
<td>Importing Source System References Using File-Based Import: Explained</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Enriching Data Using Social Data and Insight Cloud Service</td>
<td>275</td>
</tr>
<tr>
<td></td>
<td>About the Oracle Social Data and Insight Cloud service: Overview</td>
<td>275</td>
</tr>
<tr>
<td></td>
<td>Enriching Oracle Sales Cloud Data with Social Attributes: Process Overview</td>
<td>275</td>
</tr>
<tr>
<td></td>
<td>Social Attributes Supported by Data as a Service (DaaS): Explained</td>
<td>276</td>
</tr>
<tr>
<td></td>
<td>Creating Custom Fields for Social Attributes in Sales Cloud: Procedure</td>
<td>277</td>
</tr>
<tr>
<td></td>
<td>Adding Custom Sales Cloud Social Attributes to UI Pages: Procedure</td>
<td>279</td>
</tr>
<tr>
<td></td>
<td>Mapping Custom Social Attributes with the DaaS Supported Social Attributes: Procedure</td>
<td>280</td>
</tr>
<tr>
<td></td>
<td>Adding the Custom Social Attributes to Select Fields to Enrich Page: Procedure</td>
<td>281</td>
</tr>
<tr>
<td></td>
<td>Enriching Oracle Sales Cloud Account and Contact Data with Social Attributes: Procedure</td>
<td>282</td>
</tr>
<tr>
<td></td>
<td>Enriching Sales Cloud Data Manually</td>
<td>283</td>
</tr>
<tr>
<td></td>
<td>Enriching Sales Cloud Data Automatically</td>
<td>285</td>
</tr>
<tr>
<td>13</td>
<td>Managing Customer Data Quality</td>
<td>289</td>
</tr>
<tr>
<td></td>
<td>Overview of Data Quality Management</td>
<td>289</td>
</tr>
<tr>
<td></td>
<td>Cleanse Customer Information</td>
<td>289</td>
</tr>
<tr>
<td></td>
<td>FAQs for Cleanse Customer Information</td>
<td>292</td>
</tr>
<tr>
<td></td>
<td>Identify Duplicate Customer Information</td>
<td>292</td>
</tr>
<tr>
<td></td>
<td>View Nonduplicate Mapping</td>
<td>294</td>
</tr>
<tr>
<td></td>
<td>FAQs for Assign Duplicate Identification Batch</td>
<td>294</td>
</tr>
<tr>
<td></td>
<td>Resolve Duplicate Customer Information</td>
<td>295</td>
</tr>
<tr>
<td>14</td>
<td>Managing Party Information</td>
<td>303</td>
</tr>
<tr>
<td></td>
<td>Manage Party Information</td>
<td>303</td>
</tr>
<tr>
<td>15</td>
<td>Managing Hierarchy Information</td>
<td>309</td>
</tr>
<tr>
<td></td>
<td>Manage Hierarchy Information</td>
<td>309</td>
</tr>
<tr>
<td></td>
<td>FAQs for Manage Hierarchy Information</td>
<td>310</td>
</tr>
</tbody>
</table>
## 16 Managing Enterprise Resource Information

- Manage Resource Information 313
- FAQs for Manage Resource Information 316
- Manage Resource Organization Information 317
- FAQs for Manage Resource Organization Information 318
- Manage Resource Team Information 319
- FAQs for Manage Resource Team Information 320
- Perform Mass Transfer 320

## 17 Managing Protected Information

- Working with Protected Information: Explained 323

## 18 Viewing Audit Histories

- Audit History: Explained 325
- Audit Event Types: Explained 325
Preface

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

Oracle Applications Help

Use the help icon to access Oracle Applications Help in the application. If you don't see any help icons on your page, click the Show Help icon in the global header. Not all pages have help icons. You can also access Oracle Applications Help at https://fusionhelp.oracle.com.

Using Applications Help

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

Additional Resources

- **Community**: Use Oracle Applications 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.

Documentation Accessibility

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

Comments and Suggestions

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

Audience and Scope

This guide is intended for the end users of the customer data management cloud service such as customer data stewards and data steward managers.

This guide contains conceptual information and procedures needed to manage customer information and customer data quality. You can use this guide to work with the customer data management cloud service capabilities such as duplicate identification, duplicate resolution, address verification, and data enrichment.

Related Guides

You can refer to the following related guides to understand more about the tasks covered in this guide.

<table>
<thead>
<tr>
<th>Title</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Customer Data Management Cloud Getting Started with</td>
<td>Describes how to get started with the implementation of Customer Data Management cloud service capabilities such as duplicate identification, duplicate resolution, address verification, and data enrichment. This guide uses the default settings provided by Oracle and does not provide detailed explanations of all available features.</td>
</tr>
<tr>
<td>Your Customer Data Management Implementation</td>
<td></td>
</tr>
<tr>
<td>Oracle Sales Cloud Implementing Customer Data Management</td>
<td>Provides conceptual information and procedures needed to implement Customer Data Management-specific components and features of Oracle Sales Cloud.</td>
</tr>
<tr>
<td>Oracle Sales Cloud Getting Started with Your Implementation</td>
<td>Describes how to set up a sales automation solution in Oracle Sales Cloud using a case study to describe concepts and procedures.</td>
</tr>
<tr>
<td>Oracle Sales Cloud Implementing Sales</td>
<td>Describes how to configure and set up Sales.</td>
</tr>
<tr>
<td>Oracle Sales Cloud Getting Started with Oracle Sales Cloud</td>
<td>Introduces you to simple, common customizations of Oracle Sales Cloud using tools, such as Application Composer and Page Composer.</td>
</tr>
<tr>
<td>Customizations</td>
<td></td>
</tr>
<tr>
<td>Oracle Sales Cloud Customizing Sales</td>
<td>Describes how to use tools to customize and extend Oracle Sales Cloud.</td>
</tr>
<tr>
<td>Oracle Sales Cloud Groovy Scripting Reference for Application</td>
<td>Explains the basics of using the Groovy scripting language to enhance your Oracle Sales Cloud offerings.</td>
</tr>
<tr>
<td>Composer</td>
<td></td>
</tr>
<tr>
<td>Oracle Sales Cloud Understanding File-Based Data Import and</td>
<td>Describes how to import legacy and other data into Oracle Sales Cloud using File-Based Data Import.</td>
</tr>
<tr>
<td>Export</td>
<td></td>
</tr>
<tr>
<td>Oracle Sales Cloud File-Based Data Import for Oracle Sales</td>
<td>Directs you to reference information that you can use to create an importable source data file for the import of individual objects.</td>
</tr>
<tr>
<td>Cloud</td>
<td></td>
</tr>
</tbody>
</table>
Related Topics

- Oracle Help Center
2 Using File-Based Import

Overview of File-Based Import

File-Based Data Import: How It Works

The Define File-Based Data Import group of tasks relies on integration with different Oracle Sales Cloud and architecture components, such as interface tables and application base tables. This topic provides an overview of these components to help you understand the import process and the different import activity statuses.

The following figure provides an overview of the major application components used when you import data from a file. These components include:

- Import object
- Import mapping
- Import activity
- File repository
- Application Composer
- Interface tables
Import Objects, Import Mapping, and Import Activity

The import objects you select when you create an import activity are provided by Oracle. They are managed using the Manage File Import Objects task.

**Note:** Avoid concurrent submission of File-Based Data Import jobs for the same import object. Concurrent submission of multiple import jobs with same content results in creation of duplicate object records.

When you create an import activity, you must specify a mapping of the fields in your file to the attributes of the import object. You can create the mapping while creating an import activity or separately using the Manage File Import Mappings task. The mapping is stored and managed as a separate object.
File Repository
The text or XML data file you upload for import is stored in a file repository so that it is available for import processing when you schedule an import activity. Any attachments you upload are stored in the same repository.

Application Composer and Custom Extensions
When you create additional attributes for import using Application Composer, these extensions are stored in a separate repository and are available when you generate import and export artifacts in composer.

Interface Tables
The import activity populates the application interface tables with your data.

Oracle Sales Cloud Base Tables
The import activity loads your data into Oracle Sales Cloud base tables to complete the import.

Import Activity Source File Options: Explained
The Import Activity is a step-by-step guided process to assist you with creating an import activity for a given object. This topic describes the source file options defined in the Import Activity that are used by the import process to locate and parse the source file data.

Source File Data
Enter attribute details pertaining to the source file from the following table:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>File Type</td>
<td>Source file must be either Text, ZIP, or XML.</td>
</tr>
<tr>
<td>Data Type, Delimiter, and Header Row Included</td>
<td>A Text file type can further be defined based on how the data is delimited and if the source file is expected to include a row of headings for each column.</td>
</tr>
<tr>
<td>Import Mapping</td>
<td>Displays a list of predefined mappings for the object selected for this import activity. The selected mapping will be used as the basis for mapping your source file in the next Import Activity step.</td>
</tr>
</tbody>
</table>

Source File Location
The following outlines the options that are available to you when locating your source file for import.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Upload From</td>
<td>You can upload the source file from the following locations:</td>
</tr>
<tr>
<td></td>
<td>• Desktop</td>
</tr>
<tr>
<td></td>
<td>If you select Desktop, then a File Name field with an associated Update button is displayed.</td>
</tr>
<tr>
<td></td>
<td>Click Update and browse to search for and select the file you want to upload.</td>
</tr>
<tr>
<td></td>
<td>• WebCenter Content Server</td>
</tr>
</tbody>
</table>
Using Customer Data Management

Chapter 2

Using File-Based Import

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>If you select WebCenter Content Server, then a File Name field with an associated Browse button is displayed. Click and browse to search for and select the file you want to upload.</td>
<td></td>
</tr>
</tbody>
</table>

## Importing Data from a File: Procedure

Using the **Define File-Based Data Import** group of tasks that are available from the **Setup and Maintenance** work area, you can import a wide range of application data from text or XML files. Your import data can either create or update application records. Use the steps outlined in this topic as a guide. The import options and other details differ by the type of data that you’re importing. Before importing, you must understand how the data in your file maps to the attributes in the application and what values are expected as described in the related topics.

To help you get started, you can use the example templates provided in Document ID 1503223.1 on My Oracle Support. The templates include columns for the most common application attributes and reference numbers.

### Importing Data from a File

To import data from a file:

1. In the Setup and Maintenance work area, click the **All Tasks** tab in the **Overview** region, search for the Manage File Import Activities task.
2. In the search results, click the **Go to Task** button for the task.

   **Tip:** You can also navigate to the Manage Import Activities page by selecting the import task in the work areas for objects that support importing data.

3. Click the **Create** icon.
4. In the **Name** field, enter a name for your import.
5. In the **Object** field, select the object that you’re importing.
6. Depending on the object that you are importing, the corresponding file type is selected by default in the **File Type** drop-down list. For example, Deal Registration object supports only ZIP file.
7. In the **Source File** region, select one of the **Upload From** options.

   **Note:** If you’re importing a file into one of the Oracle Sales Cloud services, then you must upload files from your desktop.

   - **Desktop**
     - After you select this option, you must browse for the file.
9. If your data file includes a header row, then select the **Header Row Included** option. Although you can upload files without header rows, doing so makes it more difficult to complete the mapping between the data in your file and the application. If the values in the CSV file contain new line character, then they must be enclosed within quotation marks.
10. If your file doesn’t use a comma to separate values, then select the correct delimiter in the **Data Type** field.

   **Note:** If you had previously imported a file with the same format as the file that you’re importing now, then select an existing mapping from the **Import Mapping** list. By default, the application saves the import mapping under the import activity name appended with the activation date and time. If you’re importing a file in this format for the first time, then you must create the mapping in the next step.
11. If you're importing one of the following objects, then you can import attachments by selecting the **Browse** option in the **Attachments** region and selecting a .zip or .jar file of the attachments. You must include a column with the names of individual attachments in your import file as described in the Importing Attachments Using File-Based Data Import: Procedure topic. Do not submit File-Based Data Import jobs for the same import object because they create duplicate object records.

The following import objects support importing a single attachment for each record:

- Assets
- Opportunities
- Partners

The following import objects support multiple attachments for each record:

- Leads
- Activities

12. In the Import Options region, some of the options in the following table might be available depending on the object that you’re importing

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Import Mode</td>
<td>For importing leads and employee resources only, you can specify whether you want to create and update records or update them only.</td>
</tr>
<tr>
<td></td>
<td>If you select update, then the import process ignores any new records.</td>
</tr>
<tr>
<td></td>
<td>For all other import objects, both create and update operations are available.</td>
</tr>
<tr>
<td>Allowable Error Count</td>
<td>The Allowable Error Count threshold determines whether to automatically cancel an import if the number of consecutive rows with validation errors exceeds the user-defined threshold value. The default value is 2000 but you can change this value when configuring your import. During the &quot;Preparing Data for Import&quot; step only, if the number of consecutive rows in your input file with a validation error exceeds the Allowable Error Count value, then the import is canceled. If the Allowable Error Count is not reached in this step, then the import proceeds. Errors encountered during the &quot;Importing Data&quot; step are not counted toward the Allowable Error Count.</td>
</tr>
<tr>
<td></td>
<td>Validation errors include:</td>
</tr>
<tr>
<td></td>
<td>- Missing required values</td>
</tr>
<tr>
<td></td>
<td>- Values that exceed the attribute length</td>
</tr>
<tr>
<td></td>
<td>- Invalid identifiers and lookup codes</td>
</tr>
<tr>
<td></td>
<td>Duplicates found using matching configurations for Customer Data Management objects do not contribute to the error count.</td>
</tr>
<tr>
<td>Notification e-mail</td>
<td>The e-mail of the individual who receives import processing notifications. Currently this option is supported only for Opportunity, Lead, and custom objects.</td>
</tr>
<tr>
<td>Account Data Management Duplicates</td>
<td>This option is available only if you have licensed data quality functionality. You can select different options for handling duplicates when you’re importing accounts, contacts, and legal entities either alone or as part of another object.</td>
</tr>
<tr>
<td>Decimal Separator</td>
<td>The decimal separator used in your import file.</td>
</tr>
<tr>
<td>Option</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>Date Format</td>
<td>The format of the date fields in your file.</td>
</tr>
<tr>
<td>Time Stamp Format</td>
<td>The format of time fields in your file.</td>
</tr>
<tr>
<td>File Encoding</td>
<td>The overall encoding for your file.</td>
</tr>
</tbody>
</table>

13. Click **Next**.

The Create Import Activity: Map Fields page displays the attributes in your file with some sample data. If you’re reusing an import mapping, then both the **Source** and **Target** columns are already populated. The **Source** columns represent your file. If you’re creating a new mapping, then the **Target Object** and **Attribute** columns are blank.

*Note:* Oracle recommends that you use the Object Number (such as Opportunity Number) fields when available instead of Object ID (such as Opportunity ID) fields when importing objects.

14. For each column in your file that you want to import, select the target object and one of its attributes.

Each import object comes with its own set of attributes.

You can avoid selecting an import a column in your file by selecting the **Ignore** check box.

15. If you want to specify a constant value for an attribute in the application, then you can add the object, the attribute, and its value in the **Set Constant Values** region.

If you’re importing data using the import templates provided on My Oracle Support, then each column header in your template includes the names of both the target object and the target attribute to simplify your mapping process.

16. Click **Next**.

17. By default, the import starts immediately after you click **Activate** in step 18. You can start the import when you want by selecting **Specific Date** from the **Schedule** list and entering a start date.

18. Click **Next**.

19. Click **Activate**.

20. As a result of previous step, you return to the Manage Import Activities page to view the status of your import. An import activity with a status of Completed or Completed with Errors indicates that the import activity completed.

21. Click the status link to see data on the success of the import, including logs and error files that are linked in the Files Processed region.

**Viewing and Interpreting the Import Notification E-mail**

Once the import has been completed, an e-mail is generated showing the detailed results of your import and links to any failed operations for troubleshooting purposes. If you had specified an e-mail in the Notification E-mail field (step 11 above) e-mail is sent to that address also. Included in the e-mail are the following:

- Name of the import job and the import activity identifier
- The URL for the environment
- The object name
- Status of the import
- The Name of the import (source) file, and the type of file that was used
- Start and end times for the import (date and time)
• Elapsed time
• Record Counts: Read, Successful, Errors, Warnings
• Submitted by

Related Topics
• Getting Started with File-Based Import: Documentation Overview
• File-Based Data Import Objects: Explained

Importing Data with Special Characters Using File-Based Data Import: Quick Start

This topic describes how to format your data if your import file includes special characters. You can follow the methods described in this topic to ensure that input data is properly parsed and imported by Oracle Sales Cloud.

Building the CSV File

You can follow these steps to build your CSV file for importing the data.

1. Navigate to the Manage Import Mapping task in the Setup and Maintenance work area.
2. On the Manage File Import Mappings page, select the object name (say Lead object), select Seeded check box and click Search.
3. On the Edit Import Mappings page, click Download Template to save the CSV template file to your desktop.
4. You can edit the file to include only the fields that you want to import.

Entering Data into the CSV File

After creating the CSV file, you must enter the data to import. Some of the special characters are quotation marks(“) and a comma(,). You must prefix the special characters with quotation marks(“). The following list shows the various formats in which the input data can be entered for the Lead object CSV file and how it is interpreted due to the presence of special characters. LeadName and LeadNumber are the header fields for the file.

The following are 5 lead records in a CSV file, each with the LeadName first, followed by the LeadNumber (120, 121, and so on):

<table>
<thead>
<tr>
<th>LeadName</th>
<th>LeadNumber</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mark Twain lead</td>
<td>120</td>
</tr>
<tr>
<td>&quot;Twain &quot;secondary&quot; lead&quot;</td>
<td>121</td>
</tr>
<tr>
<td>Twain, Mark</td>
<td>122</td>
</tr>
<tr>
<td>Twain, Mark &quot;Spring 2015&quot; lead</td>
<td>123</td>
</tr>
<tr>
<td>&quot;Twain &quot;Fall 2014&quot; lead</td>
<td>124</td>
</tr>
<tr>
<td>Row Number</td>
<td>Source Data for LeadName</td>
</tr>
<tr>
<td>------------</td>
<td>--------------------------</td>
</tr>
<tr>
<td>1</td>
<td>Mark Twain lead</td>
</tr>
<tr>
<td>2</td>
<td>&quot;Twain &quot;secondary&quot; lead*</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Twain, Mark</td>
</tr>
<tr>
<td>4</td>
<td>Twain, Mark &quot;Spring 2015&quot; lead</td>
</tr>
</tbody>
</table>
As can be seen from the above table,

- If the source data contains at least one special character, then the entire field of source data must be surrounded by double quotation marks in the CSV file.
- If the data for a field contains separator like a comma (,), then you must enclose the content within two double quotation marks.
- If the first field has a special character at the beginning but not any matching special character at the end, then the content of second field will also be considered as part of the first field, and the second field is blank. You must provide a matching escape character to fix this problem.
- If you don’t precede the special characters by an escape character, then the import process may write incorrect data for a field, or incorrectly write data to the subsequent fields.

**Note:** Microsoft Excel and other spreadsheet programs may automatically escape the special characters.

### Using Predefined Templates to Import Data Through File-Based Data Import

This topic describes how to use predefined templates to import data using file-based data import. To import data using the predefined templates, you must:

- Download templates you can use for import
- Understand the import templates
- Adapt the import templates to your needs
Download Templates You Can Use for Import

Oracle Sales Cloud provides templates to help you import data using File-Based Data Import tool. The templates are spreadsheets that include the commonly used fields, in addition to the required. Each import object could have one or more templates associated with it. The below table lists the objects and the templates for the object.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Template</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account</td>
<td>Data Cloud Account Import - Advanced</td>
<td>Predefined mapping to import files generated from Data Cloud and Insight service into Oracle Sales Cloud for account enrichment.</td>
</tr>
<tr>
<td>Account Create and Update Predefined Mapping</td>
<td>This mapping lets the user create or update accounts with the most basic information.</td>
<td></td>
</tr>
<tr>
<td>Quick Create Account Predefined Mapping</td>
<td>Predefined mapping to create Accounts with basic information. This mapping cannot be used to update any attribute other than those mapped to the Account object.</td>
<td></td>
</tr>
<tr>
<td>Account Address Create and Update Predefined Mapping</td>
<td>This mapping lets the user create or update multiple addresses for existing accounts.</td>
<td></td>
</tr>
<tr>
<td>Account Relationship Create and Update Predefined Mapping</td>
<td>This mapping lets users to associate an account with one or multiple contacts.</td>
<td></td>
</tr>
<tr>
<td>Activity</td>
<td>Activity Predefined Mapping - Import Task</td>
<td>Template to import activity predefined mapping (Import Task)</td>
</tr>
<tr>
<td>Activity</td>
<td>Activity Predefined Mapping - Import Appointment</td>
<td>Template to import activity predefined mapping (Import Appointment)</td>
</tr>
<tr>
<td>Contact</td>
<td>Data Cloud Contact Import - Advanced</td>
<td>Predefined mapping to import files generated from Data Cloud and Insight service into Oracle Sales Cloud for contact enrichment.</td>
</tr>
<tr>
<td>Contact Create and Update Predefined Mapping</td>
<td>This mapping lets the user create or update contacts with the most basic information. Additionally, this template can be used to associate individual contacts with existing accounts.</td>
<td></td>
</tr>
<tr>
<td>Quick Create Contact Predefined Mapping</td>
<td>Predefined mapping to create Contacts with basic information. This mapping cannot be used to update any attribute other than the ones mapped to the Contact object. Use Original System and Original System Reference fields to establish the relationship between Contacts and Accounts.</td>
<td></td>
</tr>
<tr>
<td>Contract</td>
<td>Hierarchical Predefined Mapping-Contract Header, Parties and Contacts</td>
<td>Template to map hierarchical objects</td>
</tr>
<tr>
<td>Import Object</td>
<td>Template</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>----------------------------------------------------</td>
<td>------------------------------------------------------------------</td>
</tr>
<tr>
<td>Contract</td>
<td>Contract Predefined Mapping-Contract Header, Primary Party and Contacts</td>
<td>Template to map contract objects</td>
</tr>
<tr>
<td>Customer Sales Team</td>
<td>Sales Account Resource Team Seeded Mapping</td>
<td>Template to import sales account resource team</td>
</tr>
<tr>
<td>Customer hierarchy</td>
<td>Customer Hierarchy Create Predefined Mapping</td>
<td>This template is used to create a customer hierarchy. Customers who are part of this hierarchy should be created first using the “Create or update account” template.</td>
</tr>
<tr>
<td>Customer hierarchy member</td>
<td>Customer Hierarchy Member Create Predefined Mapping</td>
<td>This template can be used to add customers (nodes) to an existing customer hierarchy (tree).</td>
</tr>
<tr>
<td>Employee Resource</td>
<td>Employee Resource Seeded Mapping - Comprehensive</td>
<td>Template to import Employee Resource with role, resource organization and hierarchy information</td>
</tr>
<tr>
<td>Lead</td>
<td>Lead Predefined Mapping Import Leads</td>
<td>Template to import Leads with contact and company information.</td>
</tr>
<tr>
<td></td>
<td>Lead Predefined Mapping - Import Leads with Qualification information</td>
<td>Template to import Leads with contact and qualification information.</td>
</tr>
<tr>
<td>Note</td>
<td>Note Predefined Mapping - Default required Set with identifiers</td>
<td>Template to import Note - Default required Set with identifiers</td>
</tr>
<tr>
<td></td>
<td>Note Predefined Mapping - Default required Set with original system reference</td>
<td>Template to import Note - Default required Set with original system reference</td>
</tr>
<tr>
<td></td>
<td>Opportunity Predefined Mapping - All related objects</td>
<td>Template to import Opportunity with all related objects.</td>
</tr>
<tr>
<td>Partner</td>
<td>Oracle Fusion Hierarchical File Import Map for Partner - Includes Child objects</td>
<td>Template to import Partners including child objects.</td>
</tr>
<tr>
<td></td>
<td>Oracle Fusion Hierarchical File Import Map for Partner</td>
<td>Template to hierarchically import partners with minimum attributes</td>
</tr>
<tr>
<td></td>
<td>Oracle Fusion File Import Map for Partner Update</td>
<td>Template to import updates to partners.</td>
</tr>
</tbody>
</table>
### Import Object

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Template</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Oracle Fusion File Import Map for Partner</td>
<td>Template to import partners with minimum attributes</td>
</tr>
<tr>
<td>Partner Contact</td>
<td>Oracle Fusion File Import Map for Partner Contacts</td>
<td>Template to import partner contacts</td>
</tr>
<tr>
<td>Partner Program Enrollments</td>
<td>Oracle Fusion File Import Map for Enrollment Update</td>
<td>Template to import updates to enrollment</td>
</tr>
<tr>
<td></td>
<td>Oracle Fusion File Import Map for Enrollment</td>
<td>Template to import enrollments with minimum attributes</td>
</tr>
<tr>
<td>Product Group</td>
<td>Product Group Predefined Mapping - Product Group Header and Child Entities</td>
<td>Template to import product groups, items related to product groups and product group relationships used to define a hierarchy.</td>
</tr>
<tr>
<td>Quota</td>
<td>Quota Predefined Mapping - Territory and Resource Quotas</td>
<td>Template to import Territory and Resource Quotas</td>
</tr>
<tr>
<td>Sales Promotion</td>
<td>Sales Promotion Predefined Mapping - Header and Coupons</td>
<td>Template to import promotions and coupons associated with promotions.</td>
</tr>
<tr>
<td>Territory</td>
<td>Territory Import Seeded Map</td>
<td>Template to import territory</td>
</tr>
<tr>
<td>Territory Geographies</td>
<td>Territory Geographies Import Mapping</td>
<td>Template to migration import territory geographies</td>
</tr>
<tr>
<td></td>
<td>Territory Geographies Incremental Import Mapping</td>
<td>Template to import territory geographies incrementally</td>
</tr>
</tbody>
</table>

To download an import template:

1. Navigate to the **Setup and Maintenance** work area and search for the Manage File Import Mappings task.
2. Click the **Go to Task** icon.
3. In the **Manage File Import Mappings** page, select the object for which you need the template (such as Account) from the Object drop-down list.
4. Select the **Seeded** check box.
5. Click **Search**.
6. Click the template you want to download from the search results. The **Edit Import Mapping** page is displayed.
7. In the **Edit Import Mapping** page, click Download Template.
8. Save the .csv file to a location on your desktop.

### Understanding the Import Templates

You must understand the columns in the import templates to populate the templates with the correct data. To update existing records, your source file must include the attributes that enable the import process to identify these records. These values are source system and source system reference value combination, Oracle Sales Cloud internal ID, or public unique identifiers, such as business keys or external IDs. When the source of your data isn’t an external system and you don’t intend to regularly update the data, you don’t need the source system information.
Note: It is a best practice recommendation to include the source system information as requirements change over time.

Copying and Modifying a predefined Mapping

The template file includes a header row with columns that map to the attributes for the objects. You can use the mapping that is used to download the template and also to map columns to attributes during import. The columns are arranged in the same order as the mapping in Oracle Sales Cloud. You can add additional attributes to the file by appending them after the last column. In case you add additional attributes, you must copy and modify the predefined mapping to create a custom mapping that includes these additional attributes. You can use the following steps to copy and modify a predefined mapping:

1. Navigate to the Setup and Maintenance work area and search for the Manage File Import Mappings task.
2. Click the Go to Task icon
3. In the Manage File Import Mappings page, select the object for which you need the template (such as Account) from the Object drop-down list.
4. Select the Seeded check box. Click Search.
5. Click the template you want to copy or modify from the search results. The Edit Import Mapping page is displayed.
6. In the Edit Import Mapping page, click Copy Mapping.
7. The target object attributes can be edited. You can also add attributes to the object by clicking Create icon.
8. Click Save to save the modified template file.

Adapting the Import Templates

You may want to add additional attributes to your files to import additional information for your import data. You can review all available attributes for your import object in a reference file located in File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). Review the reference file to explore other attributes that are available for import and the possible prerequisite steps for those attributes.

The reference file includes the following information:

- All attributes available for your import object.
- The attribute descriptions, including the user interface display name for the attribute if the attribute is displayed on the user interface.
- The data type, length, and validation logic, including the task or work area where you can view or define valid values before importing data.
- The logic for default values, if values are not provided in your source file.

To view the reference file, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm).

Selecting the Template for Import Process

You can select the template that you downloaded during the import process, using the following steps:

1. Navigate to the Setup and Maintenance work area and search for the Manage File Import Activities task.
2. Click the Go to Task icon
3. On the Manage Import Activities page, click the Create icon.
4. On the Create Import Activity page, provide the import name and object to import. Provide the file type, select Header row included and Seeded check boxes.
5. On the Import Mapping drop-down list, select one of the predefined templates
6. Click Next to view the mapping and schedule the activation of import process.
Importing Attachments Using File-Based Data Import: Procedure

This topic describes how to include attachments when you use the Manage File Import Activities task.

Importing Attachments

You can use the following steps to import attachments using file-based data import:

1. Add one or more of the following columns to the source file that you're importing.

   The ATTACHMENT_FILE_NAME header is the only required header. All other headers are optional. The Manage File Import Activities task reserves these header names for their described usage. You must not use them for any other purpose.

<table>
<thead>
<tr>
<th>Column Header</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATTACHMENT_FILE_NAME</td>
<td>The ZIP file selected when defining the import activity must contain a file with the specified ATTACHMENT_FILE_NAME value. Otherwise an error will be generated.</td>
</tr>
<tr>
<td>ATTACHMENT_FILE_DESC</td>
<td>A description of the file to be attached to the record.</td>
</tr>
<tr>
<td>ATTACHMENT_FILE_TITLE</td>
<td>Enter the file title. This is not a mandatory field. If you don’t specify a file title, then the Manage File Import Activities task uses the file name as the title. For example, if the file name is abc.txt, and the ATTACHMENT_FILE_TITLE column value is not explicitly passed in the CSV, then it takes up the value as abc.</td>
</tr>
<tr>
<td>ATTACHMENT_CATEGORY_NAME</td>
<td>Specify the attachment category. An attachment category is used to classify and secure attachments. Each attachment UI must be defined with at least one category so that users can add attachments. For example, you can categorize attachments for an expense report as a receipt, scanned invoice image, and so on.</td>
</tr>
</tbody>
</table>

2. In each row of the source file, enter the name of the attachment file that the Manage File Import Activities task must attach to the record.

   In the following example, you enter file_1.doc as the attachment file for Lead 1.

<table>
<thead>
<tr>
<th>LeadName</th>
<th>StatusCD</th>
<th>Score</th>
<th>ATTACHMENT_FILE_NAME</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lead 1</td>
<td>QUALIFIED</td>
<td>7</td>
<td>file_1.doc</td>
</tr>
<tr>
<td>Lead 2</td>
<td>QUALIFIED</td>
<td>8</td>
<td>file_2.doc</td>
</tr>
<tr>
<td>Lead 3</td>
<td>QUALIFIED</td>
<td>3</td>
<td>file_3.doc</td>
</tr>
<tr>
<td>Lead 4</td>
<td>QUALIFIED</td>
<td>7</td>
<td>file_4.doc</td>
</tr>
</tbody>
</table>

3. To specify the same attachment file for more than one record, enter the same file name for each record in the ATTACHMENT_FILE_NAME column. For example, you can specify file_1.doc for Lead 1 and for Lead 2.
Note:
- You can specify multiple attachments for any type of business object or custom object.
- If you specify more than one attachment for the same record, and if you add any of the optional attachment columns, then you must repeat all the columns in the same order for each attachment file.

4. Create a ZIP file that includes the files that you must attach.
   For example, create a file named myfiles.zip that includes the following files:
   - file_1.doc
   - file_2.doc
   - file_3.doc
   - file_4.doc

   You can also use a JAR file. You can organize these files in folders and subdirectories, at your discretion.

5. Sign in to the Oracle Sales Cloud application.

6. Navigate to the Setup and Maintenance work area, and then search for the Manage File Import Activities task.

7. In the Search Results list, click Go to Task.

8. On the Manage Import Activities page, click the Create icon.

9. On the Create Import Activity: Enter Import Options page, choose an object from the Object drop-down list that supports attachments. To identify the objects that support attachments, see the Prerequisites section.

10. In the Attachments section, click Browse, and then locate your ZIP file.

Note:
- You can upload only one ZIP file.
- You can't add more attachments after you upload the ZIP file.

11. Enter the data in the remaining fields and activate the file-based data import activity.

   The Manage File Import Activities task recognizes the reserved column header names, so you don't have to map them in the Create Import Activity: Map Fields page. For more information about finishing this task, see the Related Topics section at the end of this topic.

Constraints for Business Objects
You can import attachments only for the following business objects:

- Account
- Activity
- Contact
- Contract
- Campaign
- Household
- Lead
- Opportunity
- Partner
You cannot delete the file attachments from a record in Oracle Sales Cloud using file-based data import activities.

**Related Topics**
- File-Based Data Import Objects: Explained
- What’s an attachment category?
- Managing Attachments for Partners: Explained

## Import Activity Import Options: Explained

This topic describes the import options available while creating import activities.

### Source File Data Transformation

The following options are used to identify the formatting of source file data so that the data can be correctly interpreted and transformed by the import process:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Decimal Separator</td>
<td>The format of the fractional portion of numeric values in columns mapped to attributes with a decimal attribute type.</td>
</tr>
<tr>
<td>Date Format</td>
<td>The format for values in columns mapped to attributes with a date attribute type.</td>
</tr>
<tr>
<td>Time Stamp Format</td>
<td>The format for values in columns mapped to attributes with a time stamp attribute type.</td>
</tr>
<tr>
<td>File Encoding</td>
<td>The overall encoding of the characters within the file.</td>
</tr>
</tbody>
</table>

### Interface to Target Import Options

Depending on the object that you’re importing and the application modules you implemented, you can select one or more of these options.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Import Mode</td>
<td>For importing leads and employee resources only, you have the option of specifying if you want to create and update records or update only. If you select update, then any new records will be ignored by the import process. For all other import objects, both create and update operations are available.</td>
</tr>
</tbody>
</table>

<p>| Allowable Error Count | The Allowable Error Count threshold determines whether to automatically cancel an import if the number of consecutive rows with validation errors exceeds the user defined threshold value. The default value for Allowable Error Count is 2000 but you can change this value when configuring your import. During the &quot;Preparing Data for Import&quot; step only, if the number of consecutive rows in |</p>
<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
</table>
| Option                          | your input file with a validation error exceeds the Allowable Error Count value, then the import will be canceled. If the Allowable Error Count is not reached in this step, then the import will proceed. Errors encountered during the “Importing Data” step are not counted as part of the Allowable Error Count. Validation errors include:  
  • Missing required values  
  • Values that exceed the attribute length  
  • Invalid identifiers and lookup codes  
  Duplicates found using matching configurations for Customer Data Management objects do not contribute to the error count. |
| Notification E-Mail             | The e-mail of the intended recipient of import processing notifications. Currently this option is supported only for Opportunity, Lead, and custom objects.                                                                 |
| Customer Data Management Duplicates | You can select the options in the Customer Data Management LOV for handling duplicates when you’re importing accounts and legal entities either alone or as part of another object. |

**Note:** You can use the Customer Data Management Duplicates LOV to retrieve duplicates only if you have licensed the data quality functionality. Once licensed, you must rebuild the keys for your matching configuration using the Manage Enterprise Data Quality Matching Configurations task. For more information, see the Define Data Quality chapter in the Oracle Sales Cloud Implementing Customer Data Management guide.

The duplicates are determined using the following matching configurations:

- Address Duplicate Identification
- Contact Duplicate Identification
- Account Duplicate Identification

Using this option, which is available only if you licensed the data quality functionality, you can select different options for handling duplicates when you’re importing accounts and legal entities either alone or as part of another object.

The duplicates are determined using the following matching configurations:

- Batch Location Basic Duplicate Identification
- Batch Person Basic Duplicate Identification
- Batch Organization Basic Duplicate Identification

You can select from one of the following:

- Do Not Import Duplicate Records
  If the main object of the Import Activity is an account or a legal entity object, then the rows that are matched to existing records will not be imported. These duplicates records are reported in the Exception and Error reports.
  If the Customer Data Management objects are components of another object and one or more matches are found, then the existing duplicate records are evaluated to determine the most recent record. The most recent record will be associated with the main object being imported.
Using File-Based Import

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Import Duplicate Records</td>
<td>The Customer Data Management objects will be imported even if matched records exist.</td>
</tr>
<tr>
<td>Import Duplicate Records and Create Resolution Request</td>
<td>The Customer Data Management objects will be imported even if matched records exist. In addition, a duplicate resolution request is created and displayed in the Customer Data Management, Duplicate Resolution work area.</td>
</tr>
<tr>
<td>Duplicate Look Back Days</td>
<td>This option applies only to the Lead import object. Only existing leads created within the period determined by the look back days value are evaluated for duplicates based on the attributes selected for duplicate validation in the predefined import mapping. If a duplicate is found, the lead will not be imported and the duplicate record will be reported on the Exception report. Duplicate leads are included in the calculation of the allowable error count threshold.</td>
</tr>
</tbody>
</table>

Import Activity Field Mapping: Explained

After entering your import options, the second step of the import activity process is to map fields in the source file to the corresponding target attributes. This topic explains the following steps in the process:

- Mapping the Fields
- Saving the Import Mapping
- Constant Values

Mapping the Fields

The Map Fields section can be subdivided into source file columns and target attribute columns. The source column header value is derived from one of the following:

- Predefined mapping, if one is selected
- The source file, if the **Header Row Included** option is selected in the first step of the import activity definition. This is for Text file type only.
- Generic values of Column A, Column B, and so on, if the **Header Row Included** option is not selected. This is for Text file type only.
- XML tagging structure. This is for XML file type only.

The following table outlines the source columns.

<table>
<thead>
<tr>
<th>Source Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Column Header</td>
<td>Represents the column header for Text file types and the tagging structure for XML file types.</td>
</tr>
<tr>
<td>Example Value</td>
<td>Values are derived from the first source file saved with the predefined mapping. If you didn’t select a predefined mapping, then the example values are taken from the first data row in the source file selected in the first step of the Import Activity definition.</td>
</tr>
<tr>
<td>Ignore</td>
<td>Select this option if you don’t want to import the source file data in that column.</td>
</tr>
</tbody>
</table>

The following table outlines the target columns.
Target Column | Description
---|---
Object | The group of import objects that represent the components of the business object being imported.
Attribute | The attribute name that represents the corresponding interface table column for the object.

**Saving the Import Mapping**

The mapping between source file information and target attributes is saved as a reusable mapping when the import activity is saved, using the import activity name and date to derive a mapping name. If you have selected a predefined mapping, then the modifications made in the Import Activity to an unlocked mapping will update and save the import activity to the predefined mapping. If the predefined mapping is locked, then a modified mapping will be saved as a new mapping. To specify a mapping name for new mappings, select the **Save As** option from the Map Fields **Actions** menu.

**Constant Values**

Constant values provide a way to specify a value for a target attribute that all imported objects will inherit. The value you're setting when you configure a constant value is at the record level. For example, if a source file doesn't contain a column for business unit and all of the objects in the file belong to the same business unit, then enter a constant value for the object and business unit attribute. Each imported record will have the specified attribute set to the constant value. The Constant value will take precedence over any values mapped or ignored in the Map Fields section. This value will apply to all source file types.

**Related Topics**

- Exporting and Importing Data Between Oracle Sales Cloud Instances Using Automatic Mapping: Procedure
- File-Based Import Mapping: Explained

**File Import Activity Statuses: Explained**

This topic explains the meaning of the different import activity statuses when you import data from a file using the Manage File Import Activities task.

The following table lists and describes the import activity statuses.

<table>
<thead>
<tr>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Queued for import</td>
<td>The import request is queued for processing.</td>
</tr>
<tr>
<td>Preparing data for import</td>
<td>The data is being staged to be imported.</td>
</tr>
<tr>
<td>Importing data</td>
<td>The data is being validated and imported.</td>
</tr>
<tr>
<td>Importing attachments</td>
<td>The attachments are stored in the file repository. This step is displayed only if the object being imported supports the import of file attachments.</td>
</tr>
<tr>
<td>Completing import activity</td>
<td>The cleanup tasks, such as the generation of log files, are in progress.</td>
</tr>
<tr>
<td>Completed</td>
<td>All records were processed and loaded into the application tables.</td>
</tr>
</tbody>
</table>
### Status Description

<table>
<thead>
<tr>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Completed with errors</td>
<td>The import activity found errors in the data file that you must correct. You can view details about the errors by clicking the status link.</td>
</tr>
<tr>
<td>Completed with warnings</td>
<td>The import activity found warnings (but no errors) in the data file. You can view details about the warnings by clicking the status link.</td>
</tr>
<tr>
<td>Unsuccessful</td>
<td>The import activity encountered system error and the import is unsuccessful.</td>
</tr>
</tbody>
</table>

**Related Topics**

- File-Based Data Import Objects: Explained

## File-Based Import Monitoring: Explained

This topic explains how to monitor all file import activities that are currently scheduled to run, have completed successfully, or failed with errors. For each import activity, you can view the details pertaining to each underlying process and make the necessary updates for any failed records that you want to import again. You can also filter the import activities that are in different states.

You can view the list of import activities from the Manage Import Activities page. You can filter the import activities based on the status of the activity, file type, name of the activity, activity identifier, object name, or name of the import mapping used. To filter,

1. Click on the **Query By Example** icon This brings up the text boxes on top of each column header.
2. Enter the filter criteria in text boxes to filter the records. For example, you may want to filter all import activities that are in the status Completed.
3. Select the import activity that you want to monitor by clicking on the hyperlink in the corresponding Status column. The **View Import Status** page is displayed, that shows status of the most recently executed instance of the import activity definition. The **View Import Status** page contains the following sections:
   - Files Processed (Import Processes in the case of multiple CSV files)
   - Import Activity Progress
   - Process Log

### Files Processed

The Files Processed section displays a row for each source file that is processed, in the case of single CSV files. The import processing details for single CSV files are summarized and displayed for each source file and include the following:

<table>
<thead>
<tr>
<th>File Processing Summary Information</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Records Read From File</td>
<td>The number of rows in the CSV file.</td>
</tr>
<tr>
<td>Records Uploaded with Format Errors</td>
<td>The number of records that were uploaded with format errors when processing data to insert into the interface tables from the source file. View the error details in the Exception and Error files attached to the process.</td>
</tr>
</tbody>
</table>
Oracle Sales Cloud
Using Customer Data Management

Chapter 2
Using File-Based Import

### File Processing Summary Information

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Records Failed with Errors</strong></td>
</tr>
<tr>
<td>The number of records that failed with errors when importing data from the interface tables to the destination application tables. View the error details in the Exception and Error files attached to the process.</td>
</tr>
<tr>
<td><strong>Records Uploaded Successfully</strong></td>
</tr>
<tr>
<td>The number of import object records that were imported to the application destination tables. If the import object is made up of multiple components, each component is counted as successfully loaded. Consequently the Successfully Loaded count may be larger than the Records Read From File count. View the successful record details in the Log file attached to the process.</td>
</tr>
<tr>
<td><strong>Records Uploaded With Warnings</strong></td>
</tr>
<tr>
<td>The number of records that were uploaded with warnings when importing data from the interface tables to the destination application tables.</td>
</tr>
</tbody>
</table>

### File Processing Summary Information

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Records Read From File</strong></td>
</tr>
<tr>
<td>The number of records read from the source file.</td>
</tr>
</tbody>
</table>

### Attachments

Once an Import Activity process has completed, processing reports are included in the Attachments column. This column will display the log file in case of successful import, and error and exception file in case of failure. The Log file includes the records that were successfully imported plus the unique destination application table identifiers for the objects. The Exception file includes a row for each record that failed. All the errors for that row are concatenated and displayed in the Error Message column of the file. The Error file includes all the errors for each record that failed validation.

In the event that log generation was not completed for an import activity, you can click the Generate Log link to generate the import log files. Once the request is submitted, you will get a confirmation message. The Generate Log link is displayed in the Attachments column.

The following are the contents of each file:

- Log files contain records that were successfully imported plus the unique destination application table identifiers for the objects.
- Exception files include a row for each record that failed. All the errors for that row are concatenated and displayed in the Error Message column of the file.
- Error files include all the errors for each record that failed validation.

Submitted By

The value of the user who last submitted an import using the import definition.

In the case of multiple CSV files, the Import Processes section displays a row for each source file that is processed,

### File Processing Summary Information

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Records Read From File</strong></td>
</tr>
<tr>
<td>The number of records read from the source file.</td>
</tr>
<tr>
<td><strong>Records Uploaded with Format Errors</strong></td>
</tr>
<tr>
<td>The number of records that were uploaded with format errors when processing data to insert into the interface tables from the source file. View the error details in the Exception and Error files attached to the process.</td>
</tr>
<tr>
<td><strong>Records Failed with Errors</strong></td>
</tr>
<tr>
<td>The number of records that failed with errors when importing data from the interface tables to the destination application tables. View the error details in the Exception and Error files attached to the process.</td>
</tr>
</tbody>
</table>
## File Processing Summary Information

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Records Uploaded Successfully</td>
</tr>
<tr>
<td>The number of import object records that were imported to the application destination tables. If</td>
</tr>
<tr>
<td>the import object is made up of multiple components, each component is counted as successfully</td>
</tr>
<tr>
<td>loaded. Consequently the Successfully Loaded count may be larger than the Records Read From File</td>
</tr>
<tr>
<td>count. View the successful record details in the Log file attached to the process.</td>
</tr>
</tbody>
</table>

| Rejected Data File                                                                                 |
| The data files that were rejected due to format errors.                                            |

| Error File                                                                                         |
| Once an Import Activity process has completed, error files are included in the Error File column.  |
| This column will display error and exception files in case of failure, and will be blank in case of |
| successful import. The Exception file includes a row for each record that failed. All the errors for |
| that row are concatenated and displayed in the Error Message column of the file. The Error file     |
| includes all the errors for each record that failed validation.                                     |

In the event that log generation was not completed for an import activity, you can click the **Generate Log** link to generate the import log files. Once the request is submitted, you will get a confirmation message. The **Generate Log** link is displayed in the **Error File** column.

The following are the contents of each file:

- Exception files include a row for each record that failed. All the errors for that row are concatenated and displayed in the Error Message column of the file.
- Error files include all the errors for each record that failed validation.

| Records Uploaded With Warnings                                                                     |
| The number of records that were uploaded with warnings when importing data from the interface tables to the destination application tables. |

| Submitted By                                                                                      |
| The value of the user who last submitted an import using the import definition.                     |

**Note:** In the files that are generated after the completion of an import activity, all error message values are enclosed within double quotation marks. The use of double quotation marks improves the format of the log files by ensuring that the error messages containing commas are not treated as multiple values.

## Import Activity Progress

From the Import Activity Progress section, you can view details pertaining to each process involved in importing the objects in the source file. A listing of brief messages provides information on processing steps within each underlying process. The various steps involved in the import activity are summarized in below table:

<table>
<thead>
<tr>
<th>Step Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Import activity queued</td>
<td>The import activity that is to be processed is being queued.</td>
</tr>
<tr>
<td>Preparing data for import</td>
<td>The data in the input files is being prepared for importing into the database.</td>
</tr>
<tr>
<td>Importing data</td>
<td>The data is being imported into the Oracle Sales Cloud database. This step will also display a count of the total number of substeps that are performed during the data loading process. The number of substeps required to load the data varies for each object.</td>
</tr>
</tbody>
</table>
Using File-Based Import

<table>
<thead>
<tr>
<th>Step Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Importing attachments</td>
<td>The application is importing attachment files and is associating them to the appropriate Oracle Sales Cloud records. This step will only be displayed if the object being imported supports the import of file attachments.</td>
</tr>
<tr>
<td>Completing import activity</td>
<td>The import log files are generated and internal import activity cleanup tasks are executed.</td>
</tr>
</tbody>
</table>

Process Log
The **Process Log** section displays the log messages that are generated during the processing of an import activity. You can filter the log messages based on the message types **Information**, **Error** and **Debug**. You can also export the log information into an excel file.

Related Topics
- Managing File-Based Import Exceptions: Worked Example
- Test link to OTN

Import Consumers Using File-Based Import

Importing Consumers: Explained

This topic explains how to prepare and import consumer data from an external data source into Oracle Sales Cloud, using the File-Based Data Import feature.

A consumer is a person who intends to purchase goods and services. Within Oracle Sales Cloud, a person with a selling relationship is referred to as a consumer. A consumer import object allows you to import all information about a consumer, such as the consumer profile, consumer address information, consumer contact point information, and consumer contact preference information.

*Note:* The Consumer object is only applicable if you are using a prior release. All capabilities of the Consumer object are now included in the Contact object. If you are upgrading from a prior release, then it is recommended that you use the Contact object to update your consumer information using File-Based Data Import.

You must consider the following questions before importing consumer information:

- How does your legacy or source system represent the consumer information compared to how Oracle Sales Cloud represents the same data?
- Do you have to configure values in Oracle Sales Cloud to map them to your existing data values?
- Do you have to customize Oracle Sales Cloud to capture attributes that are critical to the way you do business?
- What import features are available for importing your business object?
- How do I verify my imported data?
Comparing Business Object Structures

You must understand how your consumer data corresponds with the data in Oracle Sales Cloud so that you can map your legacy data to the data that Oracle Sales Cloud requires. First, you must understand how Oracle Sales Cloud represents the structure of the data for a consumer.

The consumer import object structure is hierarchical. The root of the hierarchy is the consumer profile, which must exist before you can import lower-level entities, such as e-mail, sell-to address, sales account profile, and fax. These child entities can have other entities as their child entities. This hierarchical structure supports one-to-many relationships between the components that make up the consumer.

The following figure shows the consumer and its child entities.

The person profile contains basic information about the consumer, such as the first name, last name, and party usage. For each consumer, you can have other information, such as sell-to addresses, phone details, and sales account profile.

*Note:* All entities referring to contact information, such as primary phone or e-mail, include a child entity that captures the contact preference. For example, the Primary Phone Contact Preference entity captures the contact preference of the consumer for the contact method primary phone.

Comparing Business Object Data

Each import object is a collection of attributes that helps to map your data to the Oracle Sales Cloud data and to support one-to-many relationships between the structural components that make up the consumer.

You must understand the attribute details of the import object so that you can prepare your import data. You can use reference guide files that contain attribute descriptions, values that populate attributes by default when you do not provide values, and validation information for each of the Oracle Fusion Applications attribute. The validation information includes the navigation path to the task where you can define values in Oracle Sales Cloud. For example, if you have values in your data that correlate to a choice list in Oracle Sales Cloud, then the validation information for that attribute provides the task name in the Setup and Maintenance work area where you can define your values.

Extensible Attributes

If you want to extend the Oracle Sales Cloud object to import your legacy or source data, then you must use Application Composer to design your object model extensions and to generate the required artifacts to register your extensions and make them available for importing. The corresponding import object is updated with the extensible attributes, which you can then map to your source file data. You can use the same source file to import both extensible custom attributes and the standard import object attributes.
Importing Consumers Using File-Based Data Import

For the consumers business object, you must use the File-Based Data Import feature. You prepare XML or text source data files in a form that is suitable for file-based import. The file-based import process reads the data included in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables.

The Define File-Based Data Import Setup and Maintenance task list includes the tasks that are required to configure the import objects, to create source-file mappings, and to schedule the import activities. You submit file-based import activities for each import object. When you’re creating a new consumer, you import the Consumer import object.

You must be assigned the Customer Relationship Management Application Administrator or Marketing Operations Manager job role to access and submit the import activities for consumers.

When importing consumer information, you first import the basic person profile information and then the child entities for the consumer. When importing child entities, you must provide the parent reference information for all parent levels for the entity. You must provide the PartyOrigSystem and PartyOrigSystemReference of the consumer when importing contacts for the consumer. PartyOrigSystem is the source system code that identifies the source of the information being imported. PartyOrigSystemReference is unique for each row of data within a single import, and is a combination of PartyOrigSystem and a unique reference. For example, you first import basic profile details, such as first name, last name, party type, and party usage. You then import other information, such as additional names, e-mail, sell-to address, and primary phone for the consumer.

Verifying Your Imported Data

Oracle Sales Cloud Applications provide File-Based Import activity reports, which you can use to verify imported data. You can verify your imported data by clicking the Status column for your import activity. Alternatively, you can also navigate to the Customer Center work area to view the consumer information that you have imported.

Related Topics

- Getting Started with File-Based Import: Documentation Overview
- Extending Oracle Sales Cloud: How It Works

Consumer Import Objects: How They Work Together

The Consumer import object allows you to import consumers, their detailed information, and contacts related to the consumer. This topic introduces the following:

- Target objects for the Consumer import object
- Target import object attribute
- Target object attribute reference guide files

Note: The Consumer object is only applicable if you are using a prior release. All capabilities of the Consumer object are now included in the Contact object. If you are upgrading from a prior release, then it is recommended that you use the Contact object to update your consumer information using File-Based Data Import.

You use the Consumer import object to import consumer information. It is split into separate target import objects for organizing the individual attributes of the different aspects of the consumer. To map the source data in your import file to the target attributes in Oracle Sales Cloud, you must understand how the target objects are related and what attributes are included in each target object.
The target import objects in the Consumer import object are grouped into information about the consumer. The person profile is the target import object containing attributes to import information about the consumer. When updating an existing consumer with additional information, you must provide the parent reference information for the existing consumer.

To update the information for an existing consumer or to create a consumer record, you can import consumer profile information, addresses, and contact points, such as phone and fax number. The following target import objects are for creating and updating the corresponding consumer information: PersonProfile, PrimaryPhone, Fax, Mobile, Classification, AdditionalIdentifier, AdditionalName, e-mail, SellToAddress, and SalesAccountProfile.

**Target Import Objects Attributes**

You must compare the attributes that you want to import with the target object attributes that are available and with their valid values. To evaluate your source data and Oracle Sales Cloud attributes for mapping and validation, you use a reference file. See the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. A reference guide file includes attribute descriptions, default values, and validations performed by the import process. Review the validation for each attribute to determine whether there are functional prerequisites or prerequisite setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target import object and target object attribute. You can predefine and manage import mappings using the Manage File Import Mappings task, or you can define the mapping when you define the import activity using the Manage File Import Activities task. Both tasks are available in the Setup and Maintenance work area.

> **Note:** If any of the attributes that you want to import do not have an equivalent target object attribute, then review the Application Composer features for the consumer.

**Reference Files for Target Import Object Attributes**

To access reference files for this object’s target import objects, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

The following table lists the reference files that are available by target import object.

<table>
<thead>
<tr>
<th>Target Import Object</th>
<th>Description</th>
<th>Reference Guide File Names</th>
</tr>
</thead>
<tbody>
<tr>
<td>PersonProfile</td>
<td>Includes detailed person information such as person name, relationship type, gender, and marital status.</td>
<td>HZ_IMP_PARTIES_T_Reference</td>
</tr>
<tr>
<td>SellToAddress</td>
<td>Includes consumer addresses and party sites information. If party site usage of an address is not defined, then the import process sets it to Sell-To. In case there are multiple addresses without party site usage information, then the import process designates one of the addresses as the Sell-To address.</td>
<td>HZ_IMP_PARTYSITES_T_Reference</td>
</tr>
<tr>
<td>PrimaryPhone</td>
<td>Indicates the primary phone number of the consumer. If the consumer has multiple phone numbers, one of the phone numbers is designated as the primary phone number.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>Target Import Object</td>
<td>Description</td>
<td>Reference Guide File Names</td>
</tr>
<tr>
<td>-----------------------</td>
<td>----------------------------------------------------------------------------</td>
<td>---------------------------------------------</td>
</tr>
<tr>
<td>Mobile</td>
<td>Indicates the mobile phone number of the consumer.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>Fax</td>
<td>Indicates the fax number of the consumer.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>Email</td>
<td>Indicates the e-mail of the consumer.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>InstantMessenger</td>
<td>Indicates the instant messenger or social networking information of a consumer.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>SalesAccountProfile</td>
<td>Includes detailed information about a consumer sales account. A sales account is a specific sell-to entity within a given consumer. A consumer can have multiple sales accounts and sales account profiles.</td>
<td>ZCA_IMP_SALES_ACCOUNTS_Reference</td>
</tr>
<tr>
<td>AdditionalName</td>
<td>Indicates alternative name of a consumer.</td>
<td>HZ_IMP_ADDTNLPARTYNAMES_T_Reference</td>
</tr>
<tr>
<td>Classification</td>
<td>Includes classification information for a consumer. Classification allows you to categorize entities such as parties, projects, tasks, and orders as hierarchies.</td>
<td>HZ_IMP_CLASSIFICS_T_Reference</td>
</tr>
<tr>
<td>AdditionalIdentifier</td>
<td>Includes detailed information about an additional identifier for a consumer.</td>
<td>HZ_IMP_ADDTNLPARTYIDS_T_Reference</td>
</tr>
<tr>
<td>Relationship</td>
<td>Includes information about a relationship between the consumer and another entity, such as an organization or a person.</td>
<td>HZ_IMP_RELSHIPS_T_Reference</td>
</tr>
<tr>
<td>PrimaryPhoneContactPreference</td>
<td>Indicates the consumer’s preference about being contacted through phone.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>MobileContactPreference</td>
<td>Indicates the consumer’s preference about being contacted through mobile phone.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>FaxContactPreference</td>
<td>Indicates the consumer’s preference about being contacted through fax.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>EmailContactPreference</td>
<td>Indicates the consumer’s preference about being contacted through mobile phone.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
</tbody>
</table>

**Related Topics**

- Getting Started with File-Based Import: Documentation Overview
- Extending Oracle Sales Cloud: How It Works
Import Contacts Using File-Based Import

Importing Contacts: Explained

This topic explains how to prepare and import contact data from an external data source into Oracle Sales Cloud using the File-Based Data Import feature. A contact is an individual who is a customer or a prospect, or a contact for an existing customer or consumer, or a contact that does not yet have an established business association with a customer or consumer. Thus, a contact could be an employee of a customer organization, a person you may have met who could help with your business, or a prospective or current individual customer. The contact object contains information that identifies the contact and offers the contact points of the contact. Contact points can be geographical addresses, phone numbers, e-mail IDs, URLs, messenger IDs, and so on. The contact object also contains contact preference information for the contact. You must create or import contacts before you can associate them with account objects.

Consider the following questions when importing your data:

- How does your legacy system or source system represent the contact compared to how Oracle Sales Cloud represents the same data?
- Do you have to configure values in Oracle Sales Cloud to map to your data values?
- Do you have to customize Oracle Sales Cloud to capture additional attributes that are critical to the way you do business?
- What import features are available for importing your business object?
- How do you verify your imported data?

Comparing Business Object Structures

You must understand how your contact data corresponds with the data in Oracle Sales Cloud so that you can map your legacy data to the data that Oracle Sales Cloud requires. First, you must understand how Oracle Sales Cloud represents the structure of the data for a contact.

In Oracle Sales Cloud, one table stores the contact information, and another table optionally stores contact point details for that contact. A contact point is an identified means to contact a party. Contact points can be phone numbers, e-mail IDs, Web site addresses, or instant messenger IDs.

The contact profile contains basic information about the contact, such as the contact name, party type, and party usage. For each contact, you can assign classifications, phone details, and additional names. If the contact is a consumer, then the contact relationship of the consumer includes other child entities that capture information about the consumer contact, such as contact job, contact phone, contact address, and contact e-mail.

**Note:** All contact-related entities, such phone or e-mail, include a child entity that captures the contact preference. For example, the Phone Contact Preference entity captures the contact preference of the consumer for the contact method phone.

To facilitate importing contacts, Oracle Sales Cloud incorporates the structure of the contact business object into import objects. The import object for contacts is Contact.

Comparing Business Object Data

Each import object is a collection of attributes that helps to map your data to the Oracle Sales Cloud data and to support one-to-many relationships between the structural components that make up the contact.
You must understand the attribute details of the import objects so that you can prepare your import data. You can use reference guide files that contain attribute descriptions, values that populate attributes by default when you do not provide values, and validation information for each attribute. The validation information includes the navigation path to the task where you can define values in Oracle Sales Cloud. For example, if you have values in your data that correlate to a choice list in Oracle Sales Cloud, then the validation information for that attribute provides the task name in the Setup and Maintenance work area where you can define your values. For additional information, including a list of reference guide file names and locations that you need to complete this task, see the topic listed in the following table.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Related Import Object Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contact</td>
<td>Contact Import Objects: How They Work Together</td>
</tr>
</tbody>
</table>

⚠️ Note: ⚠️

You can use the keyword importing contacts to search for related topics in Oracle Sales Cloud help.

### Extensible Attributes

If you want to extend the Oracle Sales Cloud object to import your legacy or source data, then you must use Application Composer to design your object model extensions and to generate the required artifacts to register your extensions and make them available for importing. The corresponding import object is updated with the extensible attributes, which you can then map to your source file data. You can use the same source file to import both extensible custom attributes and the standard import object attributes.

### Importing Contacts Using File-Based Data Import

For the contact business object, you must use the File-Based Data Import feature. You prepare XML or text source data files in a form that is suitable for a file-based import. The file-based import process reads the data in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables.

The Define File-Based Data Import Setup and Maintenance task list includes the tasks that are required to configure the import objects, to create source-file mappings, and to schedule the import activities. You submit file-based import activities for each import object. When you’re creating a new contact, you import the Contact object. You must be assigned the Master Data Management Administrator job role to access and submit the import activities for contacts.

### Verifying Your Imported Data

Oracle Sales Cloud Applications provide File-Based Import activity reports, which you can use to verify imported data. You can also access the Customer Center or Partner Center work area to view the contact information that you have imported.

**Related Topics**

- Getting Started with File-Based Import: Documentation Overview
- Extending Oracle Sales Cloud: How It Works

### Contact Import Objects: How They Work Together

You can import all your contact information using the Contact import object. This topic describes this import object and introduces the following:

- Target objects for the Contact import object
Contact Target Import Objects

You use the contact import object to import contact and information. The contact import object is split into separate target import objects for organizing the individual attributes of the contact and the contact’s relationship with other parties.

The target import objects included in the Contact import are grouped into information about the contact and the contact’s relationship with other parties. The contact profile is the target import object containing attributes to import information about the contact. You can have multiple contact relationships associated with a contact. There are multiple target import objects that include attributes to import contacts and their related information.

When updating an existing contact, you must provide the parent reference information of the existing contact. When importing contact profile or contact point information for a contact, you must provide relationship reference information in addition to the parent reference. You must provide this information because a contact can have multiple relationships with an organization, such as employee or board member. When importing information about a contact, you must refer to the specific relationship that you want to import information for. For example, you must specify whether you want to import information for John Smith the employee or John Smith the board member. If you do not include the relationship reference information for a contact relationship, then the import process creates a new relationship.

To update the information for an existing contact or to create a contact record, you can import contact profile information, addresses, and contact points. The following target import objects are for creating and updating contact and contact preference information: PersonProfile, Fax, PrimaryPhone, Url, SalesProfile, Classification, AdditionalName, Relationship, and AdditionalIdentifier.

To update or create a contact point for a contact relationship, use the following target import objects: ContactRelationship, ContactJob, ContactAddress, ContactEmail, ContactPhone, ContactMobile, ContactInstantMessenger, and ContactFax. All contact-point-related entities, such as phone or e-mail, include a child entity that captures the contact preference. For example, the ContactPhoneContactPreference entity captures the contact preference of the primary phone contact method.

Target Import Object Attributes

You must compare the attributes that you want to import with the target object attributes that are available and with their valid values. To evaluate your source data and Oracle Sales Cloud attributes for mapping and validation, you use a reference file. See the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. A reference guide file includes attribute descriptions, default values, and validations performed by the import process. Review the validation for each attribute to determine whether there are functional prerequisites or prerequisite setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can predefine and manage import mappings using the Manage File Import Mappings task, or you can define the mapping when you define the import activity using the Manage File Import Activities task. Both tasks are available in the Setup and Maintenance work area.

Note: If any of the attributes that you want to import do not have an equivalent for the target object attribute, then review the Application Composer extensibility features for the contact object.

Reference Files for Target Import Object Attributes

To access reference files for this object’s target import objects, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File
The following table lists the reference files that are available by target import object.

<table>
<thead>
<tr>
<th>Target Import Object</th>
<th>Description</th>
<th>Reference Guide File Names</th>
</tr>
</thead>
<tbody>
<tr>
<td>PersonProfile</td>
<td>Includes information that identifies a contact, such as first name, last name, date of birth, and so on.</td>
<td>HZ_IMP_PARTIES_T_Reference</td>
</tr>
<tr>
<td>SalesProfile</td>
<td>Includes detailed information about a sales profile. A sales profile is a specific sell-to entity within a given account. An account or contact can have multiple sales profiles.</td>
<td>ZCA_IMP_Sales_Accounts_Reference</td>
</tr>
<tr>
<td>Fax</td>
<td>Contact’s fax number.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>FaxContactPreference</td>
<td>Contact’s preferences about being contacted through Fax.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>Phone</td>
<td>Contact’s phone number. If the contact has multiple phone numbers, then the first phone number is designated as the primary phone number. The attributes of PrimaryPhone, Fax, and URL are the same. However, the value of the attribute ContactPointType is different for each of these contact point related entities.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>PhoneContactPreference</td>
<td>Contact’s preferences about being contacted by phone.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>Email</td>
<td>Contact’s e-mail.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>EmailContactPreference</td>
<td>Contact’s preferences about being contacted through e-mail.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>InstantMessenger</td>
<td>Contact’s instant messenger information.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>Mobile</td>
<td>Contact’s mobile number.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>MobileContactPreference</td>
<td>Contact’s preferences about being contacted through mobile phone.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>Classification</td>
<td>Contact’s classification information. Classification allows you to categorize entities, such as parties, projects, tasks, and orders as hierarchies.</td>
<td>HZ_IMP_CLASSIFICS_T_Reference</td>
</tr>
<tr>
<td>Address</td>
<td>Contact’s address. If the contact has multiple addresses, then the first address is designated as the primary address.</td>
<td>HZ_IMP_PARTYSITES_T_Reference</td>
</tr>
<tr>
<td>Target Import Object</td>
<td>Description</td>
<td>Reference Guide File Names</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------------------------------------</td>
</tr>
<tr>
<td>AddressContactPreference</td>
<td>Contact’s preferences about being contacted at the primary address.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>AdditionalName</td>
<td>Contact’s alternative name.</td>
<td>HZ_IMP_ADDTNLPARTYNAMES_T_Reference</td>
</tr>
<tr>
<td>AdditionalIdentifier</td>
<td>Includes the basic information about an additional identifier for the contact.</td>
<td>HZ_IMP_ADDTNLPARTYIDS_T_Reference</td>
</tr>
<tr>
<td>ContactRelationship</td>
<td>Includes information about a relationship between the contact and other parties.</td>
<td>HZ_IMP_RELSHIPS_T_Reference</td>
</tr>
<tr>
<td>ContactEmail</td>
<td>Contact’s e-mail contact point.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>ContactEmailContactPreference</td>
<td>Contact’s preferences about being contacted through the e-mail contact point.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>ContactFax</td>
<td>Indicates the number of the fax contact point of the contact.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>ContactFaxContactPreference</td>
<td>Contact’s preference about being contacted through the fax contact point.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>ContactInstantMessenger</td>
<td>Contact’s instant messenger contact point.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>ContactMobile</td>
<td>Contact’s mobile contact point.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>ContactMobileContactPreference</td>
<td>Contact’s preferences about being contacted through the mobile phone contact point.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>ContactPhone</td>
<td>Contact’s phone contact point. If the contact has multiple phone numbers, then the first phone number is designated as the primary phone number.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>ContactPhoneContactPreference</td>
<td>Contact’s preferences about being contacted through the phone contact point.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>ContactAddress</td>
<td>Contact’s address contact point. If the contact has multiple addresses, then the first address is designated as the primary address.</td>
<td>HZ_IMP_PARTYSITES_T_Reference</td>
</tr>
<tr>
<td>ContactAddressContactPreference</td>
<td>Contact’s preferences about being contacted at the address contact point.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>ContactJob</td>
<td>Contacts’ job information.</td>
<td>HZ_IMP_CONTACTS_T_Reference</td>
</tr>
</tbody>
</table>
Import Country Structures Using File-Based Import

Importing Country Structures: Explained

This topic explains how to prepare and import country structure data from an external data source using the File-Based Data Import feature. A country structure is a hierarchical grouping of geography types for a country. For example, the geography structure for the United States has the geography type of State at the top, followed by the County, then the City, and finally the Postal Code.

You can use the country structure to set up the following:

- The relationships between geographies within a country
- The types of geographies that you can define for a country

Consider the following questions when importing your data:

- How does your legacy system or source system represent the geography data compared to how the application represents the same data?
- Do you have to configure values in the application to map to your data values?
- Do you have to customize the application to capture additional attributes that are critical to the way you do business?
- What import features are available for importing your business object?
- How do you verify your imported data?

Comparing Business Object Structures

You must understand how your country structure data corresponds with the data in the application so that you can map your legacy data to the data that the application requires. First, you must understand how the application represents the structure of the data for a country structure.

You must import a separate country structure import object for each country. Each of these import objects must contain the geography types that are used in the country’s structure, organized in a hierarchy using geography level numbers. For example, if you’re importing the country structure of Australia, the country structure could be the following: 1: Country, 2: State, 3: County, 4: Town, 5: ZIP.

Import Objects for the Country Structure

To facilitate importing country structures, the application incorporates the structure of the country structure into import objects. The import object for country structures is GeoStructureLevel.

Comparing Business Object Data

Each import object is a collection of attributes that helps to map your data to the application data and to support one-to-many relationships between the structural components that make up the country structure.
You must understand the attribute details of the import objects so that you can prepare your import data. You can use reference guide files that contain attribute descriptions, values that populate attributes by default when you don’t provide values, and validation information for each attribute. The validation information includes the navigation path to the task where you can define values in the application. For example, if you have values in your data that correlate to a choice list in the application, then the validation information for that attribute provides the task name in the Setup and Maintenance work area where you can define your values. For additional information, including a list of reference guide file names and locations that you need to complete this task, see the following table.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Related Import Object Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Country Structure</td>
<td>Country Structure Import Objects: How They Work Together</td>
</tr>
</tbody>
</table>

### Extensible Attributes

If you need to extend the application object to import your legacy or source data, you must use Application Composer to design your object model extensions and to generate the required artifacts to register your extensions and make them available for importing. The corresponding import object is updated with the extensible attributes, which can then be mapped to your source file data. You can use the same source file to import both extensible custom attributes and the standard import object attributes.

### Importing Country Structures Using File-Based Data Import

For the country structure business object, you must use the File-Based Data Import feature. You prepare XML or text source data files in a form that is suitable for a file-based import. The file-based import process reads the data in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables.

The Define File-Based Data Import Setup and Maintenance task list includes the tasks that are required to configure the import objects, to create source-file mappings, and to schedule the import activities. You submit file-based import activities for each import object. When you’re creating a new country structure, you import the Country Structure object. You must be assigned the Master Data Management Administrator job role to access and submit the import activities for country structures.

### Verifying Your Imported Data

You can view the list of import activities from the Manage Import Activities page. You can verify your imported data by clicking the Status column for your import activity.

**Related Topics**

- [Getting Started with File-Based Import: Documentation Overview](#)
- [Extending Oracle Sales Cloud: How It Works](#)

### Country Structure Import Objects: How They Work Together

This topic describes the Country Structure import object. You use the Country Structure import object when you submit a file-based import activity to import your country structure information. This topic introduces the following:

- Target objects for the Country Structure import object
- Target import object attributes
- Reference guide files for target import object attributes
Country Structure Target Import Objects

The Country Structure import object contains one target import object. The target import object organizes the individual attributes of the different aspects of the geography structure. When updating an existing country structure, you must provide the parent reference information of the existing country structure. This reference information connects the imported geography structure to the existing one. Use the ImpGeoStructureLevel target import object to create and update country structure information.

Target Import Object Attributes

You must compare the attributes that you want to import with the target object attributes that are available and with their valid values. To evaluate your source data and Oracle Sales Cloud attributes for mapping and validation, you use a reference file. See the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. A reference guide file includes attribute descriptions, default values, and validations performed by the import process. Review the validation for each attribute to determine whether there are functional prerequisites or prerequisite setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can predefine and manage import mappings using the Manage File Import Mappings task, or you can define the mapping when you define the import activity using the Manage File Import Activities task. Both tasks are available in the Setup and Maintenance work area.

*Note:* If any of the attributes you want to import does not have an equivalent target object attribute, then review the Application Composer extensibility features for country structures.

Reference Files for Target Import Object Attributes

To access reference files for this object’s target import objects, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

For detailed information on importing geographies using file-based import, refer to Document No. 1481758.1, Importing Master Reference Geography Data, on the Oracle Support site.

The following table lists the reference files that are available by target import object.

<table>
<thead>
<tr>
<th>Target Import Object</th>
<th>Description</th>
<th>Reference Guide File Names</th>
</tr>
</thead>
<tbody>
<tr>
<td>ImpGeoStructureLevel</td>
<td>Information that specifies a country’s geography structure.</td>
<td>HZ_IMP_GEO_STRUCTURE_ LEVELS_Reference</td>
</tr>
</tbody>
</table>

Related Topics

- Getting Started with File-Based Import: Documentation Overview
- Extending Oracle Sales Cloud: How It Works

Import Customer Hierarchies Using File-Based Import
Importing Customer Hierarchies: Explained

This topic describes how to import a customer hierarchy and customer hierarchy members. A customer hierarchy captures the hierarchical relationships that an account has with other accounts. A customer hierarchy consists of hierarchy members. A customer hierarchy can be used to process payments from one account and apply them to another account in the same hierarchy. It can also be used to create the revenue roll-up report that rolls up revenue numbers from opportunities for all accounts in a hierarchy.

Consider the following questions when importing customer hierarchy information:

- Are all members of the customer hierarchy in Oracle Sales Cloud?
- Do you have to import additional hierarchy members to complete the hierarchy?
- How does your legacy system or source system represent the customer hierarchy compared to how Oracle Sales Cloud represents the same data?
- Do you have to configure values in Oracle Sales Cloud to map your existing data values to the Customer Hierarchy import object?
- Do you have to customize Oracle Sales Cloud to capture additional attributes that are critical to the way you do business?
- What import features are available for importing your business object?
- How do you verify my imported data?

To import a customer hierarchy, you must:

1. Import accounts that must be a part of the hierarchy.
2. Import a hierarchy definition that includes the hierarchy structure and the root node.
3. Import members from the existing accounts into the hierarchy.

Comparing Business Object Structures

Before you can import your legacy or source system customer hierarchy data, you must first analyze the data and see how it corresponds to the account object structure of Oracle Sales Cloud. You must understand how Oracle Sales Cloud represents the structure of the data for a customer hierarchy.

The customer hierarchy import object structure is hierarchical. At the top level is the customer hierarchy, which must exist before you can import customer hierarchy members. The customer hierarchy and customer hierarchy member contain information about the customer hierarchy and customer hierarchy nodes, such as the hierarchy type, hierarchy code, hierarchy name, and hierarchy version.

Comparing Business Object Data

After you understand the structure of the data, then compare the detailed attribute values of your data with the Oracle Sales Cloud data. Each import object is a collection of attributes that helps to map your data to the Oracle Sales Cloud data and to support one-to-many relationships between the structural components that make up the customer hierarchy.

You must understand the attribute details of the import objects so that you can prepare your import data. Reference files with detailed attribute information are provided in the File Based Data Import for Oracle Sales Cloud guide that is available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm).

Reference files contain attribute descriptions, values that populate attributes by default when you don’t provide values, and validation information for each attribute. The validation information includes the navigation path to the task where you can define values in Oracle Sales Cloud. For example, if you have values in your data that correspond to a choice list, then the validation information for that attribute will provide the task name in the Setup and Maintenance work area where you can define your values.
Extensible Attributes
If you want to extend the import object to import your legacy or source data, then you must use Application Composer to design your object model extensions and to generate the required artifacts to register your extensions and make them available for importing.

The corresponding import object is updated with the extensible attributes, which you can then map to your source file data. You can use the same source file to import both extensible custom attributes and the standard import object attributes.

Importing Customer Hierarchy Using File-Based Data Import
For the customer hierarchy business object, you must use the File-Based Data Import feature. You prepare XML or text source data files, such as CSV, in a form that is suitable for a file-based import. The file-based import process reads the data in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables. The Define File-Based Data Import Setup and Maintenance task list includes the tasks that are required to configure the import objects, to create source file mappings, and to schedule the import activities. You submit file-based import activities for each import object.

An import activity defines the instructions for processing import data, including the source file, import mapping from the source file to the object and attribute, and the import schedule.

When importing customer hierarchy information, you first import the customer hierarchy information and then the customer hierarchy members for the customer hierarchy. When importing customer hierarchy members, you must provide the parent reference information that refers to the customer hierarchy of the member.

Verifying Your Imported Data
Oracle Sales Cloud provides File-Based Import activity reports, which you can use to verify imported data. You can view the list of import activities from the Manage Import Activities page. You can verify the status of the import activity by clicking the Status column for your import activity. Alternatively, you can also navigate to the Manage Hierarchies task from Setup and Maintenance to view the customer hierarchy information that you have imported.

Related Topics
- Party Hierarchy: Explained
- Getting Started with File-Based Import: Documentation Overview
- Extending Oracle Sales Cloud: How It Works

Customer Hierarchy Import Objects: How They Work Together
The customer hierarchy import object imports customer hierarchy and customer hierarchy members. This topic describes the Customer Hierarchy object and introduces the following:

- Target objects for the customer hierarchy import object
- Target import object attributes
- Reference guide files for target import object attributes

Customer Hierarchy Target Import Objects
The target import objects in the customer hierarchy import object are grouped into information about the customer hierarchy and information about the customer hierarchy members. The CustomerHierarchy is the target import object containing attributes to import information about the customer hierarchy, and the CustomerHierarchyMember is the target import object.
containing attributes to import information about the customer hierarchy members. You can have multiple hierarchy members associated with a customer hierarchy.

To import a customer hierarchy, you must do the following:

1. Import customers who must be part of the hierarchy.
2. Import the customer hierarchy definition that includes the hierarchy structure and the root node.
3. Import existing customers as customer hierarchy members into the hierarchy.

Target Import Object Attributes

You must compare the attributes that you want to import with the target object attributes that are available and with their valid values. To evaluate your source data and Oracle Sales Cloud attributes for mapping and validation, you use a reference file. See the File Based Data Import for Oracle Sales Cloud guide. In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. A reference guide file includes attribute descriptions, default values, and validations performed by the import process. Review the validation for each attribute to determine whether there are functional prerequisites or prerequisite setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can predefine and manage import mappings using the Manage File Import Mappings task, or you can define the mapping when you define the import activity using the Manage File Import Activities task. Both tasks are available in the Setup and Maintenance work area.

Note: If any of the attributes that you want to import do not have an equivalent target object attribute, then review the Application Composer extensibility features for the customer hierarchy.

Reference Files for Target Import Object Attributes

To access reference files for this object’s target import objects, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. The following table lists the reference files that are available by target import object.

<table>
<thead>
<tr>
<th>Target Import Object</th>
<th>Description</th>
<th>Reference Guide File Names</th>
</tr>
</thead>
<tbody>
<tr>
<td>CustomerHierarchy</td>
<td>Customer hierarchy definition and root node of the customer hierarchy</td>
<td>HZ_IMP_HIERARCHIES_T_Reference</td>
</tr>
<tr>
<td>CustomerHierarchyMember</td>
<td>Customer hierarchy members</td>
<td>HZ_IMP_HIERARCHY_NODULES_Reference</td>
</tr>
</tbody>
</table>

Related Topics

- Party Hierarchy: Explained
- Getting Started with File-Based Import: Documentation Overview
- Extending Oracle Sales Cloud: How It Works
Importing Customer Hierarchies Using File-Based Import: Quick Start

This topic describes a few key concepts and provides guidance to get you started on importing customer hierarchies. It includes the following information:

- Customer hierarchy overview
- Identifying and associating records
- The minimum data required and the prerequisite steps
- How to access and use reference files to evaluate attributes
- Additional tips

The following table lists the file-based import objects and target objects that you can use to import customer hierarchies.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Target Objects</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer Hierarchy</td>
<td>CustomerHierarchy</td>
</tr>
</tbody>
</table>

Customer Hierarchy Overview

A customer hierarchy captures the hierarchical relationships that a customer has with other customers, and consists of hierarchy members. A customer hierarchy can be used to process payments from one customer and apply them to another customer in the same hierarchy. It can also be used to create the revenue roll-up report that rolls up revenue numbers from opportunities for all customers in a hierarchy.

You can create multiple versions of a customer hierarchy; however, only one hierarchy version can be active on a specific date. You can create a new hierarchy to represent the structure of a new customer. If the customer has made minor changes to its organization, you can create a new version rather than edit the existing one.

Identifying and Associating Records with Each Other

To update an existing customer hierarchy, your source file must include values that enable the import process to identify the existing records. These values will be either a source system and source system reference value combination or an Oracle Fusion record ID.

If the source of your data is a third party or external system, and you intend to import updates to previously imported records from the same system, you can provide the source system code and the unique reference value for the party record in your source system. Oracle Fusion stores a cross-reference between the source system information and Oracle’s record ID. To add a sales account profile to an existing party, you provide the source system code and reference values and the import process uses the value combination to identify the existing party record in Oracle Fusion.

You can set up source systems to identify the source of the data you are importing. Source systems are external sources of data that are used to import data into Oracle Sales Cloud. Source system references are used by the application to create references between source IDs and the Oracle Sales Cloud database IDs. You can configure source systems either by using the Manage Trading Community Source System task or by importing the source system information into Oracle Sales Cloud using the Source System Reference import object.

When the source of your data is not an external system and you want to add a sales account profile to an existing party, you need the Oracle Fusion party record ID in your source file. The import process uses the record ID to identify the existing record. When the source of your data is not an external system and you do not plan to regularly update the data, you do not
need the source system information. To import updates to your existing data, you can export the Oracle Fusion record ID and add it to your source file. The import process uses the record ID to identify the existing record.

**Minimum Data Required and the Prerequisite Steps**

The minimum data that is required to import customer hierarchy information depends on the following:

- The purpose of the import. The data requirements are different when you’re creating a customer hierarchy or when you’re updating a customer hierarchy record.
- The data requirements are different when using source system information or record IDs to identify and associate records.

The values that you provide in your source file may require a setup task or manual steps to ensure that the information passes the import validation. Before you prepare your source file, complete the prerequisite steps to determine the data values, and then include those values in your source file. The following table lists the minimum attributes required to import customer hierarchies based on two possible scenarios for your import, any prerequisite setup or manual steps needed to prepare your data, and the import attributes that you use to map your data to Oracle Fusion. You can optionally include attributes that are available for import in your import file but that aren’t listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Customer Hierarchy Records in the Same Batch</th>
<th>Creating Customer Hierarchy Record for an Existing Party</th>
</tr>
</thead>
<tbody>
<tr>
<td>Language</td>
<td>Indicates the code of the language into which the contents of the translatable columns are translated.</td>
<td>View valid language codes from the Setup and Maintenance, Manage ISO Languages task.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Default is US if not specified.</td>
<td>Required when creating a new hierarchy (tree version).</td>
<td>Required when creating a new hierarchy (tree version).</td>
</tr>
<tr>
<td>SourceLangFlag</td>
<td>This flag indicates whether the language specified by the Language attribute is the source language in which the contents of this record are being created or were originally created.</td>
<td>No setup task required.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Required when creating a new hierarchy (tree version).</td>
<td>Required when creating a new hierarchy (tree version).</td>
</tr>
<tr>
<td>HierarchyType</td>
<td>Unique Identifier for a tree structure.</td>
<td>Tree structure can be created and managed using the Setup and Maintenance, Manage Tree Structures task.</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>HierarchyCode</td>
<td>User-defined identifier for a tree.</td>
<td>If the import object is Employee resource, then HierarchyCode must be 'GLOBAL_ SALES_ MARKETING' else the value is free text.</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>HierarchyName</td>
<td>The name of the tree.</td>
<td>No setup task required.</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Party and Customer Hierarchy Records in the Same Batch</td>
<td>Creating Customer Hierarchy Record for an Existing Party</td>
</tr>
<tr>
<td>----------------------</td>
<td>--------------------------------------------------</td>
<td>--------------------------</td>
<td>-----------------------------------------------------------------</td>
<td>----------------------------------------------------------</td>
</tr>
<tr>
<td>HierarchyVersionName</td>
<td>The name of the tree version.</td>
<td>No setup task required.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Required when creating a hierarchy (tree version).</td>
<td>Required when creating a hierarchy (tree version).</td>
</tr>
<tr>
<td>ActionCode</td>
<td>This attribute specifies to insert or to delete the hierarchy (tree version). If provided, valid values are INSERT, UPDATE, and DELETE.</td>
<td>Use the UPDATE value when updating a record. If deleting an existing record, use the value, DELETE. If inserting a record, use the value INSERT.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>The value DELETE is required if you are deleting an existing hierarchy (tree version).</td>
<td>The value DELETE is required if you are deleting an existing hierarchy (tree version).</td>
</tr>
<tr>
<td>AsOfDate</td>
<td>The date that is used to determine the tree version to delete when ActionCode is provided as DELETE. Only one version is active on any given date.</td>
<td>No setup task required.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>A value is required if deleting an existing hierarchy (tree version).</td>
<td>A value is required if deleting an existing hierarchy (tree version).</td>
</tr>
</tbody>
</table>

### Using Reference Files to Evaluate Attributes

For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

### Additional Tips

Tip: You can explore related topics by entering the following keywords in the Oracle Fusion Help Application search. The Help Application is available from any Oracle Fusion Application page by clicking Help in the top-right corner, and selecting Applications Help.

- Importing customer hierarchies.
- Importing customer hierarchy members.

Tip: If your data is not from a source system, then you can emulate a system by defining a source system and manually generating reference values for your data. For example, if you periodically gather information at sales events or through sales calls, then you might define a source system, using a code of SALES_ACTIVITY. The format of your reference numbers might concatenate the date, salesperson name, and sequence number, such as 01_Jan_2014_Jackson_01.
Importing Customer Hierarchy Member Using File-Based Import: Quick Start

This topic describes a few key concepts and provides guidance to get you started on importing customer hierarchy members. It includes the following information:

- Identifying and associating records
- The minimum data required and the prerequisite steps
- How to access and use reference files to evaluate attributes
- Additional tips

The following table lists the file-based import objects and target objects that you can use to import customer hierarchies.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Target Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer Hierarchy Member</td>
<td>CustomerHierarchyMember</td>
</tr>
</tbody>
</table>

A customer hierarchy captures the hierarchical relationships that a customer has with other customers, and consists of hierarchy members.

Identifying and Associating Records with Each Other

To update an existing customer hierarchy member, your source file must include values that enable the import process to identify the existing records. These values will be either a source system and source system reference value combination or an Oracle Fusion record ID. If the source of your data is a third party or external system, and you intend to import updates to previously imported records from the same system, then you can provide the source system code and the unique reference value for the party record in your source system. Oracle Fusion stores a cross-reference between the source system information and Oracle’s record ID. To add a sales account profile to an existing party, you provide the source system code and reference values and the import process uses the value combination to identify the existing party record in Oracle Fusion.

You can set up source systems to identify the source of the data you are importing. Source systems are external sources of data that are used to import data into Oracle Sales Cloud. Source system references are used by the application to create references between source IDs and the Oracle Sales Cloud database IDs. You can configure source systems either by using the Manage Trading Community Source System task or by importing the source system information into Oracle Sales Cloud using the Source System Reference import object.

When the source of your data is not an external system and you want to add a sales account profile to an existing party, you need the Oracle Fusion party record ID in your source file. The import process uses the record ID to identify the existing record. When the source of your data is not an external system and you do not plan to regularly update the data, you do not need the source system information. To import updates to your existing data, you can export the Oracle Fusion record ID and add it to your source file. The import process uses the record ID to identify the existing record.

Minimum Data Required and the Prerequisite Setup Tasks

The minimum data that is required to import customer hierarchy member information depends on the following:

- The purpose of the import. The data requirements are different when you’re creating both the party and the customer hierarchy member in the same import batch, adding customer hierarchy member to an existing party, or updating a customer hierarchy member record.
• Identifying and associating records. The data requirements are different when you’re using source system information or record IDs to identify and associate records.

Attributes that are available for import in your import file but that aren’t listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Customer Hierarchy Member Records in the Same Batch</th>
<th>Creating Customer Hierarchy Member Record for an Existing Party</th>
</tr>
</thead>
<tbody>
<tr>
<td>ActionCode</td>
<td>This attribute specifies to insert or to delete the hierarchy node (tree node). This attribute is typically used only to delete a record.</td>
<td>Use the UPDATE value when updating a record. If deleting an existing record, use the value, DELETE. If inserting a record, use the value INSERT.</td>
<td>Conditionally required.</td>
<td>A value is required if deleting an existing hierarchy node.</td>
</tr>
<tr>
<td>AsOfDate</td>
<td>When ActionCode is provided as DELETE, AsOfDate determines the tree version from which to delete the hierarchy node. Only one version is active on any given date.</td>
<td>No setup task required.</td>
<td>Conditionally required.</td>
<td>A value is required if deleting an existing hierarchy node.</td>
</tr>
<tr>
<td>HierarchyType</td>
<td>Unique Identifier for a tree structure.</td>
<td>Tree structure can be created and managed using the Setup and Maintenance, Manage Tree Structures task.</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>HierarchyCode</td>
<td>User-defined identifier for a tree.</td>
<td>HierarchyCode must exist in the FND_TREE table, column TREE_CODE or, if the hierarchy is being created in the same batch as this node, then HierarchyCode must exist in HZ_IMP_HIERARCHIES_T table (the interface table for the new hierarchy), column TREE_CODE (the HierarchyCode attribute for a hierarchy import).</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>PartyOrigSystem</td>
<td>The source system code that identifies the original source system of the party that you are inserting as a new node or that exists as the node to be deleted. You provide the source system reference (PartyOrigSystemReference) and the source system</td>
<td>Predefine your source system code as enabled for parties, using the Setup and Maintenance, Manage Trading Community Source Systems task.</td>
<td>Conditionally required.</td>
<td>If deleting an existing node, then identify the node by providing either PartyId or both PartyOrigSystem and PartyOrigSystemReference.</td>
</tr>
</tbody>
</table>


<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Customer Hierarchy Member Records in the Same Batch</th>
<th>Creating Customer Hierarchy Member Record for an Existing Party</th>
</tr>
</thead>
<tbody>
<tr>
<td>PartyOrigSystemReference</td>
<td>The ID that identifies the party that is the parent node of this node in your legacy or external system. The import process uses the source system code and source system reference values to: * find the Oracle Sales Cloud internal ID (ParentPartyId) that uniquely identifies the existing parent node (party record) that was originally imported with the source system reference and source system code values in a prior batch. * find the Oracle Sales Cloud internal ID (ParentPartyId) for the new parent node with the source system reference and source system code values that is imported in the same batch.</td>
<td>Identify the reference value from your source system that was used when you imported the party in a prior batch, or identify the source system reference you will use when importing the party in the same batch.</td>
<td>Conditionally required.</td>
<td>If you’re creating a new node, then identify the parent node by providing either ParentPartyId or both ParPartyOrigSystem and ParPartyOrigSystemReference.</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td></td>
<td></td>
</tr>
<tr>
<td>------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Attribute Description</td>
<td>record that is either being inserted as a new node or is an existing node to be deleted in this batch.</td>
<td>party by exporting the Party object using the Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Creating Party and Customer Hierarchy Member Records in the Same Batch</td>
<td>If deleting an existing node, then identify the node by providing either PartyId or both PartyOrigSystem and PartyOrigSystemReference.</td>
<td>If deleting an existing node, then identify the node by providing either PartyId or both PartyOrigSystem and PartyOrigSystemReference.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ParentPartyId</td>
<td>The unique internal ID for the existing party record that is the parent node of this node.</td>
<td>You can obtain the Parent Party ID for an existing party by exporting the Party object using the Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If you’re creating a new node or deleting an existing node, and the node is not the root node of the hierarchy, then you must identify the parent node by providing either ParentPartyId or both ParPartyOrigSystem and ParPartyOrigSystemReference.</td>
<td>If you’re creating a new node or deleting an existing node, and the node is not the root node of the hierarchy, then you must identify the parent node by providing either ParentPartyId or both ParPartyOrigSystem and ParPartyOrigSystemReference.</td>
<td></td>
</tr>
<tr>
<td>ParentPartyOrigSystem</td>
<td>The source system code that identifies the original source system of the party that is the parent node of this node. You provide the source system reference (ParentPartyOrigSystemReference) and the source system code (ParentPartyOrigSystem) when the source of your data is your legacy or external system. The import process uses the source system code and source system reference values to: * find the Oracle Sales Cloud internal ID (ParentPartyId) that uniquely identifies the existing parent node (party record) that was originally imported with the source system reference and source system code values in a prior batch. * find the Oracle Sales Cloud internal ID (ParentPartyId) for the new parent node with the source system reference and source system code values that is imported in the same batch as this node.</td>
<td>Predefine your source system code as enabled for parent parties, using the Setup and Maintenance, Manage Trading Community Source Systems task.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If you’re creating a new node, then identify the parent node by providing either ParentPartyId or both ParPartyOrigSystem and ParPartyOrigSystemReference.</td>
<td>If you’re creating a new node, then identify the parent node by providing either ParentPartyId or both ParPartyOrigSystem and ParPartyOrigSystemReference.</td>
<td></td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Party and Customer Hierarchy Member Records in the Same Batch</td>
<td>Creating Customer Hierarchy Member Record for an Existing Party</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------</td>
</tr>
<tr>
<td>ParentPartyOrigSystemReference</td>
<td>The ID that identifies the party that is the parent node of this node in your legacy or external system. You provide the source system reference (ParentPartyOrigSystemReference) and the source system code (ParentPartyOrigSystem) when the source of your data is your legacy or external system. The import process uses the source system code and source system reference values to: * find the Oracle Sales Cloud internal ID (ParentPartyId) that uniquely identifies the existing parent node (party record) that was originally imported with the source system reference and source system code values in a prior batch. * find the Oracle Sales Cloud internal ID (ParentPartyId) for the new parent node with the source system reference and source system code values that is imported in the same batch.</td>
<td>Identify the reference value from your source system that was used when you imported the parent party in a prior batch, or identify the source system reference you will use when importing the parent party in the same batch.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
</tbody>
</table>

Using Reference Files to Evaluate Attributes

For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

Additional Tips

Tip: You can explore related topics by entering the following keywords in the Oracle Fusion Help Application search. The Help Application is available from any Oracle Fusion Application page by clicking Help in the top-right corner, and selecting Applications Help.

- Importing customer hierarchies.
- Importing customer hierarchy members.

Tip: If your data is not from a source system, then you can emulate a system by defining a source system and manually generating reference values for your data. For example, if you periodically gather information at sales events or through sales...
calls, then you might define a source system, using a code of SALES_ACTIVITY. The format of your reference numbers might concatenate the date, salesperson name, and sequence number, such as 01_Jan_2014_Jackson_01.

Import Customers Using File-Based Import

Importing Customers: Explained

This topic explains how to prepare and import account data from an external data source into Oracle Sales Cloud using the File-Based Data Import feature. An account is an entity with whom you have a selling relationship. Oracle Sales Cloud refers to sales profile and sales prospects collectively as an Account. An account business object allows you to capture all information about an account, such as the account profile, account contact relationships information, and account contact points.

A sales profile is a specific sell-to entity within a given account. You can create leads and opportunities for sales profiles. An entity with a sales profile can be one of the following:

- Sales Prospect: A prospective entity or person who does not have a sell-to address. This entity is used to define leads.
- Account: A sales profile or sales prospects.
- Legal Entity: A legal entity is a party that can enter into legal contracts or a business relationship. It can be sued if it fails to meet contractual obligations.

You can use the Account import object to import sales profiles or sales prospects of the type organization. You can use the Household import object to import accounts of the type household, and the Contact import object to import accounts of the type person. Consider the following questions when importing your data:

- Did you identify the records that should be imported as sales prospects, and the records that should be imported as sales profiles?
- How does your legacy or source system represent the account information compared to how Oracle Sales Cloud represents the same data?
- Do you have to configure values in Oracle Sales Cloud to map your existing data values to the Account import object?
- Do you have to customize Oracle Sales Cloud to capture attributes that are critical to the way you do business?
- What import features are available for importing your business object?
- How do I verify my imported data?

Comparing Business Object Structures

You must understand how your account data corresponds with the data in Oracle Sales Cloud so that you can map your legacy data to the data that Oracle Sales Cloud requires. First, you must understand how Oracle Sales Cloud represents the structure of the data for an account.

The Account import object structure is hierarchical. The root of the hierarchy is the organization profile, which must exist before you can import lower-level components, such as classifications, contacts, and sales profiles. These child entities
can have other entities as their child entities. This hierarchical structure supports one-to-many relationships between the components that make up the account. The following figure shows the account object and its child entities.

![Account Hierarchy Diagram]

The organization profile contains basic information about the account, such as the customer name and party usage. For each account, you can assign classifications, contacts, sell to addresses, phone details, and additional names. The contact of the account, in turn, includes other child entities that capture information about the contact, such as contact job, contact primary phone, contact primary address, and contact e-mail.

**Note:** All contact entities, such as primary phone or e-mail, include a child entity that captures the contact preference. For example, the Primary Phone Contact Preference entity captures the contact preference of the account for the contact method primary phone.

**Comparing Business Object Data**

Each import object is a collection of attributes that helps to map your data to the Oracle Sales Cloud data and to support one-to-many relationships between the structural components that make up the account.

You must understand the attribute details of the import objects so that you can prepare your import data. You can use reference guide files that contain attribute descriptions, values that populate attributes by default when you do not provide values, and validation information for each attribute. The validation information includes the navigation path to the task where you can define values in Oracle Sales Cloud. For example, if you have values in your data that correlate to a choice list in Oracle Sales Cloud, then the validation information for that attribute provides the task name in the Setup and Maintenance work area where you can define your values. For additional information, including a list of reference guide file names and locations that you need to complete this task, see the topic listed in the following table.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Related Import Object Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account</td>
<td>Account Import Objects : How They Work Together</td>
</tr>
</tbody>
</table>
Extensible Attributes
If you want to extend the Oracle Sales Cloud object to import your legacy or source data, then you must use Application Composer to design your object model extensions and to generate the required artifacts to register your extensions and make them available for importing. The corresponding import object is updated with the extensible attributes, which you can then map to your source file data. You can use the same source file to import both extensible custom attributes and the standard import object attributes.

Importing Accounts Using File-Based Data Import
For the account business object, you must use the File-Based Data Import feature. You prepare XML or text source data files in a form that is suitable for a file-based import. The file-based import process reads the data in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables.

The Define File-Based Data Import Setup and Maintenance task list includes the tasks that are required to configure the import objects, to create source-file mappings, and to schedule the import activities. You submit file-based import activities for each import object. When you’re creating a new account, you import the Account object. You must be assigned the Customer Relationship Management Application Administrator job role to access and submit the import activities for accounts.

When importing account information, you first import the basic account profile information, followed by the child entities for the account. When importing child entities, you must provide the parent reference information for all parent levels for the entity. For example, you first import basic profile details, such as customer name, party type, and party usage. You then import contacts and contact information, such as phone, address, contact points, and fax for the account. You must provide the PartyOrigSystem and PartyOrigSystemReference of the account when importing contacts for the account. PartyOrigSystem is the source system code that identifies the source of the information being imported. PartyOrigSystemReference is unique for each row of data within a single import, and is a combination of PartyOrigSystem and a unique reference. When importing contact information for an account, you must provide the relationship reference information. This information is required because a contact can have multiple relationships with an account.

Verifying Your Imported Data
Oracle Sales Cloud Applications provide File-Based Import activity reports, which you can use to verify imported data. Alternatively, you can also navigate to the Customer Center work area to view the account information that you have imported.

Related Topics
- Getting Started with File-Based Import: Documentation Overview
- Extending Oracle Sales Cloud: How It Works

Account Import Objects: How They Work Together
The Account import object allows you to import accounts, their detailed information, and contacts related to the account. This topic describes the Account import object. It introduces the following:
- Target objects for the Account import objects
Account Target Import Object Concepts

You use the Account import object to import accounts and contacts related to accounts. The Account import object is split into separate target import objects for organizing the individual attributes of the account and account contact.

The target import objects in the Account import object are generally grouped into information about the account and information about the account contact. The organization profile is the target import object containing attributes to import information about the account. You can have multiple contacts associated with an account.

When updating an existing account with additional information, you must provide the parent reference information for the existing account. When importing account contacts or contact information for an account, you must provide relationship reference information in addition to the parent reference. This information is required because a contact can have multiple relationships with an organization, such as an employee or board member. When importing information about a contact you must refer to the specific relationship that you want to import information for. For example, you might want to import information for John Smith the employee or John Smith the board member. If you do not include the reference information for an account-contact relationship, then the import process creates a relationship.

To update the information for an existing account or to create an account record, you can import account profile information, addresses, and contact points, such as a phone and fax. The following target import objects are for creating and updating account information: OrganizationProfile, Fax, Address, PrimaryPhone, Url, SalesAccountProfile, Classification, AdditionalName, Relationship, SellToAddress, and AdditionalIdentifier.

To update or to create an account contact, use the following target import objects: ContactPersonProfile, ContactJob, ContactPrimaryAddress, ContactEmail, ContactPrimaryPhone, ContactMobile, ContactInstantMessenger, and ContactFax. All contact-related entities, such as primary phone or e-mail, include a child entity that captures the contact preference. For example, the Primary Phone Contact Preference entity captures the contact preference of the legal entity for the contact method primary phone.

Target Import Objects Attributes

You must compare the attributes that you want to import with the target object attributes that are available and with their valid values. To evaluate your source data and Oracle Sales Cloud attributes for mapping and validation, you use a reference file. See the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. A reference guide file includes attribute descriptions, default values, and validations performed by the import process. Review the validation for each attribute to determine whether there are functional prerequisites or prerequisite setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can predefine and manage import mappings using the Manage File Import Mappings task, or you can define the mapping when you define the import activity using the Manage File Import Activities task. Both tasks are available in the Setup and Maintenance work area.

**Note:** If any of the attributes you want to import do not have an equivalent target object attribute, then review the Application Composer Extensibility features for the account.

Target Import Objects Attributes Resources

To access reference files for this object’s target import objects, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File
Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

The following table lists the reference files that are available by target import object:

<table>
<thead>
<tr>
<th>Target Import Object</th>
<th>Description</th>
<th>Reference Guide File Names</th>
</tr>
</thead>
<tbody>
<tr>
<td>OrganizationProfile</td>
<td>Includes detailed account information, such as, organization name and organization type.</td>
<td>HZ_IMP_PARTIES_T_Reference</td>
</tr>
<tr>
<td>PrimaryPhone</td>
<td>Account’s primary phone number. If the account has multiple phone numbers, one of the phone numbers is designated as the primary phone number.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>Address</td>
<td>Account’s address.</td>
<td>HZ_IMP_LOCATIONS_T_Reference</td>
</tr>
<tr>
<td>Classification</td>
<td>Account's classification.</td>
<td>HZ_IMP_CLASSIFICS_T_Reference</td>
</tr>
<tr>
<td>AdditionalName</td>
<td>Account’s additional or alternative name.</td>
<td>HZ_IMP_ADDTNLPARTYNAMES_T_Reference</td>
</tr>
<tr>
<td>Fax</td>
<td>Account’s fax number.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>FaxContactPreference</td>
<td>Indicates the account preference about being contacted through Fax.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>Url</td>
<td>Account’s URL.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>SalesProfile</td>
<td>Account’s sales account information. A sales account is a specific sell-to entity within a given account, an account can have multiple sales accounts and sales profiles.</td>
<td>ZCA_IMP_SALES_ACCOUNTS_Reference</td>
</tr>
<tr>
<td>SellToAddress</td>
<td>Account’s addresses and party sites information. If party site usage of an address is not defined, then the import process sets it to Sell-To. If there are multiple addresses without party site usage information, then the import process designates one of the addresses as the Sell-To address.</td>
<td>HZ_IMP_PARTYSITES_T_Reference</td>
</tr>
<tr>
<td>AdditionalIdentifier</td>
<td>Account’s additional identifier information.</td>
<td>HZ_IMP_ADDTNLPARTYIDS_T_Reference</td>
</tr>
<tr>
<td>Relationship</td>
<td>Account’s relationship information. You must enter a relationship code in the RelationshipCode column, when creating a relationship.</td>
<td>HZ_IMP_RELSHIPS_T_Reference</td>
</tr>
<tr>
<td>ContactPersonProfile</td>
<td>Account contact’s person information.</td>
<td>HZ_IMP_RELSHIPS_T_Reference</td>
</tr>
<tr>
<td>Target Import Object</td>
<td>Description</td>
<td>Reference Guide File Names</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------------------------------------------</td>
</tr>
<tr>
<td>ContactAdditionalName</td>
<td>Account contact’s additional name.</td>
<td>HZ_IMP_ADDTNLPARTYNAMES_T_Reference</td>
</tr>
<tr>
<td>ContactEmail</td>
<td>Account contact’s email.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>ContactInstantMessenger</td>
<td>Account contact’s instant messenger or social networking information.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>ContactPrimaryAddress</td>
<td>Account contact’s primary address. If the contact has multiple addresses, one of the addresses is designated as the primary address.</td>
<td>HZ_IMP_PARTYSITES_T_Reference</td>
</tr>
<tr>
<td>ContactFax</td>
<td>Account contact’s fax.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>ContactJob</td>
<td>Account contact’s job information.</td>
<td>HZ_IMP_CONTACTS_T_Reference</td>
</tr>
<tr>
<td>ContactMobile</td>
<td>Account contact’s mobile number.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>ContactPrimaryPhone</td>
<td>Account contact’s primary phone number. If the account has multiple phone numbers, one of the phone numbers is designated as the primary phone number.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>PrimaryPhoneContactPreference</td>
<td>Indicates the account contact preference about being contacted through phone.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>ContactEmailContactPreference</td>
<td>Indicates the account contact preference about being contacted through e-mail.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>ContactFaxContactPreference</td>
<td>Indicates the account contact preference about being contacted through Fax.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>ContactMobileContactPreference</td>
<td>Indicates the account contact preference about being contacted through mobile phone.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>ContactPrimaryAddressContactPreference</td>
<td>Indicates the account contact preference about being contacted at the primary address.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>ContactPrimaryPhoneContactPreference</td>
<td>Indicates the account contact preference about being contacted through the primary phone number.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
</tbody>
</table>

**Note:** Oracle Sales Cloud supports the ContactPersonProfile target object. However, the object is now deprecated. For updating any contact information, use the PersonProfile target object instead.
Import Employee Resources Using File-Based Import

Importing Employee Resources: Explained

This topic explains how to prepare and import employee resource data from an external data source into Oracle Sales Cloud using the File-Based Data Import feature. Employee resources are employees within the deploying company who can be assigned work objects. You must create or import employee resources before you can associate them with resource organizations, resource teams, or work objects. You enter your employee resource information using the Setup and Maintenance work area, Manage Resources task, or you can import data to create new or update existing employee resources.

Consider the following questions when importing your data:

- How does your legacy system or source system represent the employee resource compared to how Oracle Sales Cloud represents the same data?
- Do you have to configure values in Oracle Sales Cloud to map to your data values?
- Do you have to customize Oracle Sales Cloud to capture additional attributes that are critical to the way you do business?
- What import features are available for importing your business object?
- How do you verify your imported data?

Comparing Business Object Structures

You must understand how your employee resource data corresponds with the data in Oracle Sales Cloud so that you can map your legacy data to the data that Oracle Sales Cloud requires. First, you must understand how Oracle Sales Cloud represents the structure of the data for an employee resource.
In Oracle Sales Cloud, one table stores the employee resource definition, and other tables optionally store profile details for that employee resource. Profile details for a resource include information about an employee resource’s organization and team memberships. The following figure shows the structure of the Employee Resource object.

The worker profile contains basic information about the employee resource, such as the employee resource’s primary address and resource profile details.

**Import Objects for the Employee Resource**

To facilitate importing employee resources, Oracle Sales Cloud incorporates the structure of the employee resource into import objects. The import object for the employee resource is Employee Resource.

**Comparing Business Object Data**

Each import object is a collection of attributes that helps to map your data to the Oracle Sales Cloud data and to support one-to-many relationships between the structural components that make up the employee resource.

You must understand the attribute details of the import objects so that you can prepare your import data. You can use the reference guide files that contain attribute descriptions, values that populate attributes by default when you do not provide values, and validation information for each attribute. The validation information includes the navigation path to the task where you can define values in Oracle Sales Cloud. For example, if you have values in your data that correlate to a choice list in Oracle Sales Cloud, then the validation information for that attribute provides the task name in the Setup and Maintenance work area where you can define your values. For additional information, including a list of reference guide file names and locations that you need to complete this task, see the following table.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Related Import Object Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Employee Resource</td>
<td>Employee Resource Import Objects: How They Work Together</td>
</tr>
</tbody>
</table>

**Note:**

You can use the keyword importing employee resources to search for related topics in Oracle Sales Cloud help.

**Extensible Attributes**

If you want to extend the Oracle Sales Cloud object to import your legacy or source data, then you must use Application Composer to design your object model extensions and to generate the required artifacts to register your extensions and
make them available for importing. The corresponding import object is updated with the extensible attributes, which you
can then map to your source file data. You can use the same source file to import both extensible custom attributes and the
standard import object attributes.

**Importing Employee Resources Using File-Based Data Import**

For the employee resource business object, you must use the File-Based Data Import feature. You prepare XML or text
source data files in a form that is suitable for a file-based import. The file-based import process reads the data in your source
file, populates the interface tables according to your mapping, and imports the data into the application destination tables.

The Define File-Based Data Import Setup and Maintenance task list includes the tasks that are required to configure the
import objects, to create source-file mappings, and to schedule the import activities. You submit file-based import activities
for each import object. When you’re creating a new employee resource, you import the Employee Resource object.

**Note:** Employee Resource import is impacted by the error limit settings in the HCM loader. The default error
limit in HCM loader may not always be suitable for Oracle Sales Cloud implementations. It is recommended that
you review the current settings and set the Loader Maximum Errors Allowed parameter to an acceptable value
before launching Employee Resource Import. You can set this parameter in the Setup and Maintenance work
area, in the HCM Configuration for Coexistence task. The value for this parameter must be set depending upon
the batch sizes that you would run. For example, if you run import jobs with 5000 rows at a tolerance level of
20%, then you can set this parameter to 1000. On the other hand, if you run import jobs with 5000 rows at a
tolerance of .5%, then you can set this parameter to 25.

You must be assigned the Master Data Management Administrator job role to access and submit the import activities for
employee resources.

**Note:** The data quality options, such as matching and cleansing, are not relevant for Employee Resource
import. Hence, while importing employee resources using file-based import, ensure that the Customer Data
Management Duplicates LOV is set to blank.

**Verifying Your Imported Data**

Oracle Sales Cloud Applications provide File-Based Import activity reports, which you can use to verify imported data. Users
with the Master Data Management Administrator job role can also navigate to the Manage Resources work area to view the
imported employee resources.

**Related Topics**

- Getting Started with File-Based Import: Documentation Overview
- Extending Oracle Sales Cloud: How It Works

**Employee Resource Import Objects: How They Work Together**

Employee resources are employees to whom you can assign work objects, for example, service agents, sales managers, and
so on. You can import all your employee resource information using the Employee Resource import object.

This topic introduces the following:

- Target objects for the Employee Resource import object
- Target import object attributes
- Reference guide files for target import object attributes
Employee Resource Target Import Objects

You can use the Employee Resource import object to import an employee resource's worker profile information, such as first name, last name, address, and so on. You can also import the employee resource's resource profile information, such as resource organization and resource team memberships. To map the source data in your import file to the target attributes in Oracle Sales Cloud, you must understand how the target objects are related and what attributes are included in each target object.

The target import objects included in the Employee Resource import object are grouped into information about the employee resource's profile and the employee resource's address information. The Worker profile is the target import object containing attributes to import information about the employee resource.

When updating an existing employee resource, you must provide the parent reference information of the existing employee resource. When importing an employee resource's resource team information, you must provide relationship reference information in addition to the parent reference, because an employee resource can belong to more than one team simultaneously.

To update the information for an existing employee resource or to create an employee resource record, you can import employee resource profile information and addresses. The following target import objects are for creating and updating employee resource information: WorkerProfile, ResourceProfile, and PrimaryAddress.

To update or create an employee resource’s resource profile, use the following target import objects: ResourceOrganizationMembership and ResourceTeamMembership.

Target Import Object Attributes

You must compare the attributes that you want to import with the target object attributes that are available and with their valid values. To evaluate your source data and Oracle Sales Cloud attributes for mapping and validation, you use a reference file. See the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. A reference guide file includes attribute descriptions, default values, and validations performed by the import process. Review the validation for each attribute to determine whether there are functional prerequisites or prerequisite setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can predefine and manage import mappings using the Manage File Import Mappings task, or you can define the mapping when you define the import activity using the Manage File Import Activities task. Both tasks are available in the Setup and Maintenance work area.

Note: If any of the attributes you want to import do not have an equivalent target object attribute, then review the Application Composer extensibility features for the employee resource.

Reference Files for Target Import Object Attributes

To access reference files for this object’s target import objects, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

The following table lists the reference files that are available by target import object.
## Related Topics

- Getting Started with File-Based Import: Documentation Overview
- Extending Oracle Sales Cloud: How It Works

### Import Geographies Using File-Based Import

#### Importing Geographies: Explained

This topic explains how to prepare and import geography data from an external data source using the File-Based Data Import feature. A geography is any region with a boundary around it, regardless of its size. It might be a state, a country, a city, a county, or a ward. You must create or import geographies before you can associate them with custom zones and addresses.

**Note:** The application ships with third-party (Nokia) master geography data for multiple countries that can be easily imported. You can import Oracle-licensed Nokia data from Navteq, for those countries where the data is available, such as the U.S. You can import Nokia Geography data using the Manage Geographies task. Search for the country, and select Import Nokia Data from the Actions menu. If the licensed Navteq data is not available for a particular country, then the Import Nokia Data action is disabled. For more information, see Replacing Existing Master Geography Data with Revised Nokia Geography Data: Procedure. If Nokia geography data is not available for a country, then use the information in this chapter to import it using File-Based Data Import.

Consider the following questions when importing your data:

- How does your legacy system or source system represent the geography data compared to how Oracle applications represent the same data?
- Do you have to configure values in the application to map to your data values?
- What import features are available for importing your business object?
- How do you verify your imported data?
Comparing Business Object Structures
You must understand how your geography data corresponds with the data in the application so that you can map your legacy data to the data that the application requires. First, you must understand how the application represents the structure of the data for a geography.

You must import a separate country structure import object for each country. Each of these import objects must contain the geography types that are used in the country’s structure, organized in a hierarchy using geography level numbers. For example, if you are importing the country structure of Australia, the country structure could be the following: 1: Country, 2: State, 3: County, 4: Town, 5: ZIP.

Import Objects for the Geography
To facilitate importing geographies, the application incorporates the structure of the geography into import objects. The import object for the geography is ImpGeography.

Comparing Business Object Data
Each import object is a collection of attributes that helps to map your data to the application data and to support one-to-many relationships between the structural components that make up the geography.

You must understand the attribute details of the import objects so that you can prepare your import data. You can use reference guide files that contain attribute descriptions, values that populate attributes by default when you do not provide values, and validation information for each import object attribute. The validation information includes the navigation path to the task where you can define values in the application. For example, if you have values in your data that correlate to a choice list in the application, then the validation information for that attribute provides the task name in the Setup and Maintenance work area where you can define your values. For additional information, including a list of reference guide file names and locations that you need to complete this task, see the following table.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Related Import Object Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>ImpGeography</td>
<td>Geography Import Objects: How They Work Together</td>
</tr>
</tbody>
</table>

Note:
You can use the keyword importing geographies to search for related topics in Help.

Extensible Attributes
The application doesn’t support extensible attributes for geographies. You can import only data for geography object that already exist by default in the application.

Importing Geographies Using File-Based Data Import
For the geography business object, you must use the File-Based Data Import feature. You prepare XML or text source data files in a form that is suitable for a file-based import. The file-based import process reads the data in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables.

The Define File-Based Data Import Setup and Maintenance task list includes the tasks that are required to configure the import objects, to create source-file mappings, and to schedule the import activities. You submit file-based import activities for each import object. When you’re creating a new geography, you import the Geography object. You must be assigned the Master Data Management Administrator job role to access and submit the import activities for geographies.

When importing geography information, you must provide the parent reference information for all parent levels for the entity.
Verifying Your Imported Data

Oracle applications provide File-Based Import activity reports, which you can use to verify imported data. Users with the Master Data Management Administrator job role can also navigate to the Manage Geographies work area to view the imported geographies.

Related Topics
- Getting Started with File-Based Import: Documentation Overview

Geography Import Objects: How They Work Together

This topic describes the Geography import object. You use the Geography import object to import geography information.

This topic introduces the following:
- Target objects for the Geography import object
- Target import object attributes
- Reference guide files for target import object attributes

Geography Target Import Objects

You can use the Geography import object to import geography hierarchy information to create or update the geography data of a country. To map the source data in your import file to the target attributes in the application, you must understand how the target objects are related and what attributes are included in each target object.

The target import objects in the Geography import object contain information about the geography hierarchy. When updating an existing geography, you must provide the parent reference information of the existing geography, which connects the geography to the country of which it is a part.

Use the ImpGeography target import object to create and update geography information.

Note: Before you import geography data for a country, you must define the country’s geography structure.

Target Import Object Attributes

You must compare the attributes that you want to import with the target object attributes that are available and with their valid values. To evaluate your source data and Oracle Sales Cloud attributes for mapping and validation, you use a reference file. See the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. A reference guide file includes attribute descriptions, default values, and validations performed by the import process. Review the validation for each attribute to determine whether there are functional prerequisites or prerequisite setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can predefine and manage import mappings using the Manage File Import Mappings task, or you can define the mapping when you define the import activity using the Manage File Import Activities task. Both tasks are available in the Setup and Maintenance work area.

Note: If any of the attributes you want to import do not have an equivalent target object attribute, then review the Application Composer extensibility features for geography.
Reference Files for Target Import Object Attributes

To access the reference guide files for the geography’s target import objects, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. For detailed information on importing geographies using file-based import, refer to Document No. 1481758.1, Importing Master Reference Geography Data, on the Oracle Support site.

The following table lists the reference files that are available by target import object.

<table>
<thead>
<tr>
<th>Target Import Object</th>
<th>Description</th>
<th>Attribute Reference Guide File Names</th>
</tr>
</thead>
<tbody>
<tr>
<td>ImpGeography</td>
<td>Contains information that captures a country’s geography hierarchy details, such as geography type, geography code, etc.</td>
<td>HZ_IMP_GEOGRAPHIES_T_Reference</td>
</tr>
</tbody>
</table>

Related Topics

- Getting Started with File-Based Import: Documentation Overview
- Importing Geographies Using File-Based Data Import: Worked Example

Import Households Using File-Based Import

Importing Households: Explained

This topic explains how to prepare and import household data from an external data source into Oracle Sales Cloud using the File-Based Data Import feature.

A household is a party consisting of a collection of persons and organizations. A household, similar to a person or organization, can enter into a business relationship with other parties and can be assigned to opportunities and leads. A household can have members of party type person or organization. You can maintain household information in Oracle Sales Cloud to create leads and opportunities for a household, or capture information about a household to know it better. However, maintaining this information may not let you assign all sales and marketing activities to a household.

Consider the following questions when importing your data:

- How does your legacy or source system represent the household information compared to how Oracle Sales Cloud represents the same data?
- Do you have to configure values in Oracle Sales Cloud to map your existing data values to the Household import object?
- Do you have to customize Oracle Sales Cloud to capture attributes that are critical to the way you do business?
- What import features are available for importing your business object?
- How do I verify my imported data?
Comparing Business Object Structures
You must understand how your household data corresponds with the data in Oracle Sales Cloud so that you can map your legacy data to the data that Oracle Sales Cloud requires. First, you must understand how Oracle Sales Cloud represents the structure of the data for an household.

The Household import object structure is hierarchical. The root of the hierarchy is the group profile, which must exist before you can import lower-level components, such as classifications, contacts, and relationships. These child entities can have other entities as their child entities. This hierarchical structure supports one-to-many relationships between the components that make up the household.

The following figure shows the household import object and its child entities.

![Household Import Object Diagram]

The household profile contains basic information about the household, such as the group name, group type, and the group-party usage. For each household, you can assign classifications, members, relationships, additional identifier, and additional names.

Note: All entities referring to contact information, such as primary phone or e-mail, include a child entity that captures the contact preference. For example, the Phone Contact Preference entity captures the contact preference of the household for the phone contact method.

Comparing Business Object Data
Each import object is a collection of attributes that helps to map your data to the Oracle Sales Cloud data and to support one-to-many relationships between the structural components that make up the household.

You must understand the attribute details of the import objects so that you can prepare your import data. You can use reference guide files that contain attribute descriptions, values that populate attributes by default when you do not provide values, and validation information for each attribute. The validation information includes the navigation path to the task where you can define values in Oracle Sales Cloud. For example, if you have values in your data that correlate to a choice list in Oracle Sales Cloud, then the validation information for that attribute provides the task name in the Setup and Maintenance work area where you can define your values.

Note:
You can use the keyword importing households to search for related topics in Oracle Sales Cloud Help.

**Extensible Attributes**
If you want to extend the Oracle Sales Cloud object to import your legacy or source data, then you must use Application Composer to design your object model extensions and to generate the required artifacts to register your extensions and make them available for importing. The corresponding import object is updated with the extensible attributes, which you can then map to your source file data. You can use the same source file to import both extensible custom attributes and the standard import object attributes.

**Importing Households Using File-Based Data Import**
For the household business object, you must use the File-Based Data Import feature. You prepare XML or text source data files in a form that is suitable for a file-based import. The file-based import process reads the data in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables.

The Define File-Based Data Import Setup and Maintenance task list includes the tasks that are required to configure the import objects, to create source-file mappings, and to schedule the import activities. You submit file-based import activities for each import object. When you’re creating a new household, you import the household object.

You must be assigned the Customer Relationship Management Application Administrator job role to access and submit the import activities for household.

When importing household information, you first import the basic household information, followed by the child entities for the household. When importing child entities, you must provide the parent reference information for all parent levels for the entity. For example, you first import basic profile details, such as name and household type. You then import contacts and contact information, such as phone, address, contact points, and fax for the household. You must provide the PartyOrigSystem and PartyOrigSystemReference of the household when importing contacts for the household. PartyOrigSystem is the source system code that identifies the source of the information being imported.

**Verifying Your Imported Data**
Oracle Sales Cloud Applications provide File-Based Import activity reports, which you can use to verify imported data. Alternatively, you can also navigate to the Customer Center work area to view the household information that you have imported.

**Related Topics**
- Getting Started with File-Based Import: Documentation Overview
- Extending Oracle Sales Cloud: How It Works

**Household Import Objects: How They Work Together**
You use the Household import object to import households, their basic information, and members of the household. This topic describes the Household import object and introduces the following:

- Target objects for the Household import object
- Target import object attributes
- Reference guide files for target import object attributes

**Household Target Import Objects**
The Household import object is split into separate target import objects for organizing the individual attributes for the different aspects of the household. The target import objects in the Household import object are grouped into information about the
household and information about the member. The group profile is the target import object containing attributes to import information about the household. You can have multiple members associated with a household. You can assign only an organization or a person as a member of a household. You cannot assign a household as a member of another household.

When updating an existing household with additional information, you must provide the parent reference information for the existing household. When importing contacts or contact information for a household, you must provide relationship reference information in addition to the parent reference. When importing information about a member, you must refer to the specific relationship that you want to import information for. For example, you might want to import information for John Smith the employee or John Smith the board member. If you do not include the reference information for a relationship, then the import process will create a relationship.

To update the information for an existing household or to create a household record, you can import household profile information, addresses, and contact points, such as phone and fax. The following target import objects are for creating and updating the household information: GroupProfile, Fax, Mobile, Phone, e-mail, InstantMessenger, Relationship, Member, Address, Classification, AdditionalName, and AdditionalIdentifier.

All contact-related entities, such phone or e-mail, include a child entity that captures the contact preference. For example, the Phone Contact Preference entity captures the contact preference of the household for the contact method primary phone. Additionally, the Address import object for a household includes another child entity, AddressPurpose, that captures the purpose of the current household address.

**Target Import Objects Attributes**

You must compare the attributes that you want to import with the target object attributes that are available and with their valid values. To evaluate your source data and Oracle Sales Cloud attributes for mapping and validation, you use a reference file. See the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. A reference guide file includes attribute descriptions, default values, and validations performed by the import process. Review the validation for each attribute to determine whether there are functional prerequisites or prerequisite setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can predefine and manage import mappings using the Manage File Import Mappings task, or you can define the mapping when you define the import activity using the Manage File Import Activities task. Both tasks are available in the Setup and Maintenance work area.

> **Note:** If any of the attributes you want to import do not have an equivalent target object attribute, then review the Application Composer extensibility features for the marketing response.

**Target Import Objects Attributes Resources**

To access reference files for this object’s target import objects, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

The following table lists the reference files that are available by target import object.

<table>
<thead>
<tr>
<th>Target Import Object</th>
<th>Description</th>
<th>Reference Guide File Names</th>
</tr>
</thead>
<tbody>
<tr>
<td>GroupProfile</td>
<td>Household information. The default party usage for a household is SALES_PROSPECT.</td>
<td>HZ_IMP_PARTIES_T_Reference</td>
</tr>
<tr>
<td>Target Import Object</td>
<td>Description</td>
<td>Reference Guide File Names</td>
</tr>
<tr>
<td>----------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>----------------------------------</td>
</tr>
<tr>
<td>Relationship</td>
<td>Relationship between the household and the household member. You must enter a relationship code in the RelationshipCode column, when creating a relationship.</td>
<td>HZ_IMP_RELSHIPS_T</td>
</tr>
<tr>
<td>Member</td>
<td>Household member's information.</td>
<td>HZ_IMP_RELSHIPS_T</td>
</tr>
<tr>
<td>Classification</td>
<td>Household classification. Classification allows you to categorize entities such as parties, projects, tasks, and orders as hierarchies.</td>
<td>HZ_IMP_CLASSIFICS_T</td>
</tr>
<tr>
<td>AdditionalIdentifier</td>
<td>Household's additional identifier information.</td>
<td>HZ_IMP_ADDTNLPARTYID_T</td>
</tr>
<tr>
<td>AdditionalName</td>
<td>Household's alternative name.</td>
<td>HZ_IMP_ADDTNLPARTYNAMES_T</td>
</tr>
<tr>
<td>URL</td>
<td>Household's URL.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>InstantMessenger</td>
<td>Household's instant messenger or social networking information.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>Fax</td>
<td>Household's fax.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>FaxContactPreference</td>
<td>Indicates the household preference about being contacted through fax.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>Mobile</td>
<td>Household's mobile number.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>MobileContactPreference</td>
<td>Indicates the household's preference about being contacted through mobile phone.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>Phone</td>
<td>Primary phone number of the household. If the household has multiple phone numbers, one of the phone numbers is designated as the primary phone number.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>PrimaryPhoneContactPreference</td>
<td>Indicates the household's preference about being contacted through phone.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>Email</td>
<td>Household's e-mail.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>EmailContactPreference</td>
<td>Indicates the household's preference about being contacted through e-mail.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>Address</td>
<td>Household's address. If the household has multiple addresses, one of the addresses is designated as the primary address.</td>
<td>HZ_IMP_LOCATIONS_T HZ_IMP_PARTYSITES_T</td>
</tr>
<tr>
<td>AddressContactPreference</td>
<td>Indicates the household's preference about being contacted at the primary address.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
</tbody>
</table>
Import Legal Entities Using File-Based Import

Importing Legal Entities: Explained

This topic explains how to prepare and import legal entities from an external data source into Oracle Sales Cloud using the File-Based Data Import feature.

A legal entity is a recognized party with rights and responsibilities that is established by legislation. You create a legal entity for each registered company or other entity recognized in law for which you want to record assets, liabilities, expenses and income, pay transaction taxes, or perform intercompany trading. Within Oracle Sales Cloud, a legal entity is a representation of the legal unit of a customer.

Consider the following questions when importing legal entity information:

- How does your legacy or source system structure represent the legal entity information compared to how Oracle Sales Cloud represents the same data?
- How do your legacy or source system legal entities relate to business units, divisions, ledgers, balancing segments, consolidation rules, intercompany transactions, and payroll reporting?
- Do you have to configure values in Oracle Sales Cloud to map to your existing data values to the Legal Entity import object?
- Do you have to customize Oracle Sales Cloud to capture additional attributes that are critical to the way you do business?
- What import features are available for importing your business object?
- How do I verify my imported data?
Comparing Business Object Structures

Before you can import your legacy or source system legal entity data, you must first analyze the data and see how it corresponds with the legal entity object structure of Oracle Sales Cloud. You must understand how Oracle Sales Cloud represents the structure of the data for a legal entity.

The legal entity structure of Oracle Sales Cloud is hierarchical. The organization profile is at the root level of the legal entity. The organization profile must exist before you can import lower-level components, such as e-mail, classification, additional name, primary address, and fax. These child entities can have other entities as their child entities. This hierarchical structure supports one-to-many relationships between the components that make up the legal entity.

The following figure shows the legal entity and its child entities.

![Legal Entity Profile Diagram]

The organization profile contains basic information about the legal entity, such as the legal entity name and party usage. For each legal entity, you can assign classifications, contacts, phone details, and additional names. The contact of the legal entity, in turn, includes other child entities that capture information about the contact, such as contact job, contact primary phone, contact primary address, and contact e-mail.

> **Note:** All contact-related entities, such as primary phone or e-mail, include a child entity that captures the contact preference. For example, the Primary Phone Contact Preference entity captures the contact preference of the legal entity for the contact method primary phone.

Comparing Business Object Data

After you understand the structure of the data, compare the detailed attribute values of your data with the Oracle Sales Cloud data. Each import object is a collection of attributes organized to help you when mapping your data to the Oracle Sales Cloud data and to support one-to-many relationships between the structural components that make up a legal entity.
Chapter 2
Using File-Based Import

You must understand the attribute details of the import objects so that you can prepare your import data. The reference guide files contain descriptions, default values, and validation for each of the attributes. You can use reference guide files that contain attribute descriptions, values that populate attributes by default when you don’t provide values, and validation information for each attribute. The validation information includes the navigation path to the task where you can define values in Oracle Sales Cloud. For example, if you have values in your data that correspond to a choice list, then the validation information for that attribute provides the task name in the Setup and Maintenance work area where you can define your values.

**Extensible Attributes**

If you want to extend the Oracle Sales Cloud object to import your legacy or source data, then you must use Application Composer to design your object model extensions and to generate the required artifacts to register your extensions and make them available for importing. The corresponding import object is updated with the extensible attributes, which you can then map to your source file data. You can use the same source file to import both extensible custom attributes and the standard import object attributes.

**Importing Legal Entities Using File-Based Data Import**

For the legal entity business object, you must use the File-Based Data Import feature. You prepare XML or text source data files in a form that is suitable for a file-based import. The file-based import process reads the data in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables. The Define File-Based Data Import Setup and Maintenance task list includes the tasks that are required to configure the import objects, create source file mappings, and schedule the import activities. You submit file-based import activities for each import object.

An import activity defines the instructions for processing the import data, including the source file, import mapping from the source file to the object and attribute, and the import schedule.

When importing legal entity information, you first import the basic legal entity profile information and then the child entities for the legal entity. When importing child entities, you must provide the parent reference information for all parent levels for the entity. You must provide the PartyOrigSystem and PartyOrigSystemReference of the legal entity when importing contacts for the legal entity. PartyOrigSystem is the source system code that identifies the source of the information being imported. PartyOrigSystemReference is unique for each row of data within a single import, and is a combination of PartyOrigSystem and a unique reference. For example, you first import basic profile details, such as first name, last name, party type, and party usage. You then import contacts and contact information, such as phone, address, contact points, and fax for the legal entity.

> **Note:** When importing contact information for a legal entity, you must provide the relationship reference information. This information is required because a contact can have multiple relationships with a legal entity.

**Verifying Your Imported Data**

You can view the list of import activities from the Manage Import Activities page. You can verify your imported data by clicking the Status column for your import activity. Alternatively, you can also navigate to the Define Legal Entities task list from the Setup and Maintenance work area to view the legal entities that you have imported.

**Related Topics**

- Legal Entities: Explained
- Getting Started with File-Based Import: Documentation Overview
- Extending Oracle Sales Cloud: How It Works
Legal Entity Import Objects: How They Work Together

The legal entity import object allows you to import legal entities, their detailed information, and contacts related to the entity. This topic describes the Legal Entity import object. It introduces the following:

- Target objects for the legal entity import object
- Target import object attributes
- Reference guide files for target import object attributes

Legal Entity Target Import Objects

The Legal Entity import object imports legal entity information and its contacts. The Legal Entity import object is split into separate target import objects for organizing the individual attributes of the legal entity and its contact.

The target import objects in the Legal Entity import object are grouped into information about the legal entity and the contact. The organization profile is the target import object containing attributes to import information about the legal entity. You can have multiple contacts associated with a legal entity. There are multiple target import objects that include attributes to import contacts and their related information.

When updating an existing legal entity with more information, you must provide the parent reference information for the existing legal entity. When importing contacts or contact information for a legal entity, you must provide relationship reference information in addition to the parent reference. This information is required because a contact can have multiple relationships with the legal entity. When importing information about a contact, you must refer to the specific relationship you want to import information for. For example, you might want to import information about John Smith the employee or John Smith the board member. If you do not include the reference information for a contact relationship, then the import process creates a relationship.

To update the information for an existing legal entity or to create a legal entity, you can import legal entity profile information, addresses, and contact points, such as phone and fax. The following target import objects are for creating and updating the corresponding legal entity information: OrganizationProfile, PrimaryAddress, PrimaryPhone, Url, Fax, Classification, AdditionalName, and ContactPersonProfile.

To update or create a contact, use the following target import objects: ContactPersonProfile, ContactJob, ContactPrimaryAddress, ContactEmail, ContactPrimaryPhone, ContactMobile, ContactInstantMessenger, and ContactFax. All contact-related entities, such as primary phone or e-mail, include a child entity that captures the contact preference. For example, the Primary Phone Contact Preference entity captures the contact preference of the legal entity for the contact method primary phone.

Target Import Objects Attributes

You must compare the attributes that you want to import with the target object attributes that are available and with their valid values. To evaluate your source data and Oracle Sales Cloud attributes for mapping and validation, you use a reference file. See the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. A reference guide file includes attribute descriptions, default values, and validations performed by the import process. Review the validation for each attribute to determine whether there are functional prerequisites or prerequisite setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can predefine and manage import mappings using the Manage File Import Mappings task, or you can define the mapping when you define the import activity using Manage File Import Activities task. Both tasks are available in the Setup and Maintenance work area.
Note: If any of the attributes you want to import do not have an equivalent target object attribute, then review the Composer extensibility features for the object.

Reference Files for Target Import Object Attributes

To access reference files for this object’s target import objects, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

The following table lists the reference files that are available by target import object.

<table>
<thead>
<tr>
<th>Target Import Object</th>
<th>Description</th>
<th>Reference File Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>OrganizationProfile</td>
<td>Detailed legal entity information, such as, organization name and organization type.</td>
<td>HZ_IMP_PARTIES_T_Reference</td>
</tr>
<tr>
<td>PrimaryPhone</td>
<td>Primary phone number of the legal entity. If the legal entity has multiple phone numbers, one of the phone numbers is designated as the primary phone number.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>Fax</td>
<td>Fax of the legal entity.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>Url</td>
<td>Url of the legal entity.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>AdditionalName</td>
<td>Alternative name of a legal entity.</td>
<td>HZ_IMP_ADDTNLPARTYNAMES_T_Reference</td>
</tr>
<tr>
<td>Classification</td>
<td>Classification information for a legal entity. Classification allows you to categorize entities such as parties, projects, tasks, and orders as hierarchies.</td>
<td>HZ_IMP_CLASSIFICS_T_Reference</td>
</tr>
<tr>
<td>PrimaryAddress</td>
<td>Primary address of a legal entity. If the contact has multiple addresses, one of the addresses is designated as the primary address.</td>
<td>HZ_IMP_PARTYSITES_T_Reference</td>
</tr>
<tr>
<td>ContactPersonProfile (See note below.)</td>
<td>Detailed information of a legal entity contact, such as, person name, relationship type, and marital status.</td>
<td>HZ_IMP_RELSHIPS_T_Reference</td>
</tr>
<tr>
<td>ContactEmail</td>
<td>E-mail of the legal entity contact.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>ContactFax</td>
<td>Fax of the legal entity contact.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>ContactInstantMessenger</td>
<td>Instant messenger or social networking information of a legal entity contact.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>ContactJob</td>
<td>Job information of a legal entity contact.</td>
<td>HZ_IMP_CONTACTS_T_Reference</td>
</tr>
</tbody>
</table>
## Target Import Object

<table>
<thead>
<tr>
<th>Target Import Object</th>
<th>Description</th>
<th>Reference File Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>ContactMobile</td>
<td>Mobile number of a legal entity’s contact.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>ContactPrimaryAddress</td>
<td>Primary address of a legal entity contact. If the contact has multiple addresses, one of the addresses is designated as the primary address.</td>
<td>HZ_IMP_PARTYSITES_T_Reference</td>
</tr>
<tr>
<td>ContactPrimaryPhone</td>
<td>Primary phone number of the legal entity contact. If the legal entity has multiple phone numbers, one of the phone numbers is designated as the primary phone number.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>ContactAdditionalName</td>
<td>Alternative name of a legal entity contact.</td>
<td>HZ_IMP_ADDTNLPARTYNAMES_T_Reference</td>
</tr>
<tr>
<td>PrimaryPhoneContactPreference</td>
<td>The legal entity’s preference about being contacted through phone.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>MobileContactPreference</td>
<td>The legal entity’s preference about being contacted through mobile.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>FaxContactPreference</td>
<td>The legal entity’s preference about being contacted through fax.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>EmailContactPreference</td>
<td>The legal entity’s preference about being contacted through e-mail.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>ContactPrimaryAddressContactPreference</td>
<td>The legal entity contact’s preference about being contacted at the primary address.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>ContactPrimaryPhoneContactPreference</td>
<td>The legal entity contact’s preference about being contacted through the primary phone number.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>PrimaryAddressContactPreference</td>
<td>The legal entity’s preference about being contacted at the primary address.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>PrimaryPhoneContactPreference</td>
<td>The legal entity’s preference about being contacted through the primary phone number.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
</tbody>
</table>

**Note:** Oracle Sales Cloud supports the ContactPersonProfile target object. However, the object is now deprecated. For updating any contact information, use the PersonProfile target object instead.

**Related Topics**

- Legal Entities: Explained
Import Resource Teams Using File-Based Import

Importing Resource Teams: Explained

This topic explains how to prepare and import resource team data from an external data source into Oracle Sales Cloud using the File-Based Data Import feature. A resource team is a temporary group of resources that is formed to work on work objects. A resource team may contain a resource organization or resources or both. A resource team can’t be hierarchically structured and isn’t intended to implement an organization. You enter your resource team information using the Setup and Maintenance work area, Manage Resource Teams task, or you can import data to create new or update the existing resource teams.

Consider the following questions when importing your data:

• How does your legacy system or source system represent the resource team compared to how Oracle Sales Cloud represents the same data?
• Do you have to configure values in Oracle Sales Cloud to map to your data values?
• Do you have to customize Oracle Sales Cloud to capture additional attributes that are critical to the way you do business?
• What import features are available for importing your business object?
• How do you verify your imported data?

Comparing Business Object Structures

You must understand how your resource team data corresponds with the data in Oracle Sales Cloud so that you can map your legacy data to the data that Oracle Sales Cloud requires. First, you must understand how Oracle Sales Cloud represents the structure of the data for a resource team.

In Oracle Sales Cloud, one table stores the resource team definition and other tables optionally store profile details for that resource team. Profile details for a resource include information about a resource team’s organization and team memberships.

Import Objects for the Resource Team

To facilitate importing resource teams, Oracle Sales Cloud incorporate the structure of the resource team into import objects. The import object for the resource team is Resource Team.

Comparing Business Object Data

Each import object is a collection of attributes that helps to map your data to the Oracle Sales Cloud data and to support one-to-many relationships between the structural components that make up the resource team.

You must understand the attribute details of the import objects so that you can prepare your import data. You can use reference guide files that contain attribute descriptions, values that populate attributes by default when you don’t provide values, and validation information for each attribute. The validation information includes the navigation path to the task where
you can define values in Oracle Sales Cloud. For example, if you have values in your data that correlate to a choice list in Oracle Sales Cloud, then the validation information for that attribute provides the task name in the Setup and Maintenance work area where you can define your values. For additional information, including a list of reference guide file names and locations that you need to complete this task, see the topic listed in the following table.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Related Import Object Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resource Team</td>
<td>Resource Team Import Objects: How They Work Together</td>
</tr>
</tbody>
</table>

💡 Note:
You can use the keyword importing resource teams to search for related topics in Oracle Sales Cloud help.

### Extensible Attributes

If you want to extend the Oracle Sales Cloud object to import your legacy or source data, then you must use Application Composer to design your object model extensions and to generate the required artifacts to register your extensions and make them available for importing. The corresponding import object is updated with the extensible attributes, which you can then map to your source file data. You can use the same source file to import both extensible custom attributes and the standard import object attributes.

### Importing Resource Teams Using File-Based Data Import

For the resource team business object, you must use the File-Based Data Import feature. You prepare XML or text source data files in a form that is suitable for a file-based import. The file-based import process reads the data in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables.

The Define File-Based Data Import Setup and Maintenance task list includes the tasks that are required to configure the import objects, to create source-file mappings, and to schedule the import activities. You submit file-based import activities for each import object. When you’re creating a new contact, you import the Resource Team object. You must be assigned the Master Data Management Administrator job role to access and submit the import activities for resource teams.

### Verifying Your Imported Data

Oracle Sales Cloud Applications provide File-Based Import activity reports, which you can use to verify imported data. Users with the Master Data Management Administrator job role can also navigate to the Manage Resource Teams work area to view the imported resource teams.

**Related Topics**

- Getting Started with File-Based Import: Documentation Overview
- Extending Oracle Sales Cloud: How It Works

### Resource Team Import Objects: How They Work Together

This topic describes the Resource Team import object. You use two main import objects, ResourceTeam and ResourceTeamUsage, when you submit a file-based import activity to import resource team information. This topic introduces the following:

- Target objects for the Resource Team import object
• Target import object attributes
• Reference guide files for target import object attributes

Resource Team Target Import Objects

The target import objects in the Resource Team import object are generally grouped into information about the resource team and its usage. The ResourceTeam target import object contains information that identifies the resource team, such as the team’s name, ID, active dates, and so on. The ResourceTeamUsage target import object contains information that captures the usage of the resource team. To import or update a resource team, use the ResourceTeam target import object. To import or update a resource team’s usage, use the ResourceTeamUsage target import object.

Target Import Objects Attributes

You must compare the attributes that you want to import with the target object attributes that are available and with their valid values. To evaluate your source data and Oracle Sales Cloud attributes for mapping and validation, you use a reference file. See the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. A reference guide file includes attribute descriptions, default values, and validations performed by the import process. Review the validation for each attribute to determine whether there are functional prerequisites or prerequisite setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can predefine and manage import mappings using the Manage File Import Mappings task, or you can define the mapping when you define the import activity using the Manage File Import Activities task. Both tasks are available in the Setup and Maintenance work area.

> **Note:** If any of the attributes you want to import do not have an equivalent target object attribute, then review the Application Composer Extensibility features for the object.

Reference Files for Target Import Object Attributes

To access reference files for this object’s target import objects, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

The following table lists the reference files that are available by target import object.

<table>
<thead>
<tr>
<th>Target Import Object</th>
<th>Description</th>
<th>Reference File Names</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResourceTeam</td>
<td>Captured information that identifies a resource team.</td>
<td>HZ_IMP_TEAMS_Reference</td>
</tr>
<tr>
<td>ResourceTeamUsage</td>
<td>Contains information that indicates the usage of the resource team.</td>
<td>HZ_IMP_TEAM_USAGES_Reference</td>
</tr>
</tbody>
</table>

**Related Topics**

• Getting Started with File-Based Import: Documentation Overview

• Extending Oracle Sales Cloud: How It Works
Import Common Supporting Objects

Importing Additional Party Identifiers: Quick Start

This topic describes how to get you started with importing additional party identifiers. It includes the following information:

- An example of how to identify and associate records
- The minimum data required and the prerequisite tasks
- How to access and use reference files to evaluate attributes
- Additional tips

With an additional party identifier, you can capture other identifiers for an account, contact, or household. For example, a person may have a driving license as the identifier in a record, and have passport as the identifier in another record. You can import new additional identifiers or update existing additional identifier records, using the File-Based Data Import feature. For example, an airline company may have an International Air Transport Association code and the D-U-N-S number as identifiers.

The following table lists the file-based import objects and target objects that you can use to import additional identifiers.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Party Type</th>
<th>Target Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account</td>
<td>Organization</td>
<td>AdditionalIdentifier</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contact</td>
<td>Person</td>
<td>AdditionalIdentifier</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Household</td>
<td>Household</td>
<td>AdditionalIdentifier</td>
</tr>
</tbody>
</table>

Identifying and Associating Records with Each Other

To add additional identifiers to a party or to update existing additional identifiers using file-based import, your source file must contain information about the organization, person, or household party to which the additional identifier belongs.

If you’re creating the party and the additional identifier in the same import activity, then group the party and additional identifier data together in your source file. The import process can import the party and then identify the party record so that it can import the associated additional identifiers.

To add an additional identifier to an existing party or to update an existing additional identifier record, your source file must include the values that enable the import process to identify the existing records. These values will be source system and source system reference value combination, Oracle Sales Cloud internal ID, or public unique identifiers, such as business keys or external IDs.

If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file. The file-based data import process stores a cross-reference between the source system information and Oracle Sales Cloud internal ID. The import process uses the combination of source system and source system reference value to identify the existing record.
Minimum Data Required and the Prerequisite Setup Tasks

The minimum data that is required to import additional identifier information depends on the following:

- The purpose of the import. The data requirements are different when you’re creating both the party and the additional identifier in the same import batch, adding new additional identifiers to an existing party, or updating an existing additional identifier record.
- Identifying and associating records. In some cases, you can choose which attributes you want to provide to the import process when identifying and associating records.

The values that you provide in your source file may require a setup task or manual step to ensure that the information passes the import validation. Before you prepare your source file, complete the prerequisite steps to determine the data values and then include those values in your source file.

The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create or update an additional party identifier record. You can optionally include attributes that are available for import in your import file but that aren’t listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Additional Party Identifier Records in the Same Batch</th>
<th>Creating Additional Party Identifier Record for an Existing Party</th>
</tr>
</thead>
<tbody>
<tr>
<td>AdditionalPartyIdOrigSystem</td>
<td>The code representing the source system for the additional identifier.</td>
<td>When you predefine your source system code in the Setup and Maintenance, Manage Trading Community Source Systems task, select the Enabled for Trading Community Members check box.</td>
<td>Conditionally required</td>
<td>Conditionally required</td>
</tr>
<tr>
<td>AdditionalPartyIdOrigSystemReference</td>
<td>The reference number or text representing the source system unique ID for the additional identifier.</td>
<td>Identify or generate reference values for each additional party identifier record. The reference value must be unique for all additional party identifiers for the source system.</td>
<td>Conditionally required</td>
<td>Conditionally required</td>
</tr>
<tr>
<td>ObjectKey</td>
<td>The Oracle Sales Cloud internal ID for the additional identifier.</td>
<td>Identify the additional party identifier value by exporting the Additional Party identifier object, using the Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>The attribute is neither required nor conditionally required.</td>
</tr>
</tbody>
</table>
### Attribute | Description | Prerequisite Setup Task | Creating Party and Additional Party Identifier Records in the Same Batch | Creating Additional Party Identifier Record for an Existing Party
---|---|---|---|---
PartyId | The Oracle Sales Cloud internal ID for the party (account, contact, household, or legal entity) to which the additional identifier belongs. | Identify the party ID value by exporting the Party object using Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task. | The attribute is neither required nor conditionally required. | Conditionally required. Provide reference information to identify the existing party. The reference information can be:
- PartyId (Oracle Sales Cloud internal ID)
- PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.

PartyOrigSystem | The code representing the source system for the party (account, contact, household, or legal entity) to which the additional identifier belongs. | Identify the source system code that was used when you imported the party in a prior batch, or identify the source system code that you’ll use when importing the party in the same batch as this additional party identifier. | Required | Conditionally required. Provide reference information to identify the existing party. The reference information can be:
- PartyId (Oracle Sales Cloud internal ID)
- PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.

PartyOrigSystem Reference | The reference number or text representing the source system unique ID for the party (account, contact, household, or legal entity) to which the additional identifier belongs. | Identify the reference value from your source system that was used when you imported the party in a prior batch, or identify the source system reference that you’ll use when importing the party in the same batch as this additional party identifier. | Required | Conditionally required. Provide reference information to identify the existing party. The reference information can be:
- PartyId (Oracle Sales Cloud record ID)
### Attribute | Description | Prerequisite Setup Task | Creating Party and Additional Party Identifier Records in the Same Batch | Creating Additional Party Identifier Record for an Existing Party
--- | --- | --- | --- | ---
AdditionalPartyIdentifierType | The identifier type to categorize the identifier. | Identify additional identifier types for the party type using the Setup and Maintenance, Manage Additional Identifier Types task. | Required | Required
AdditionalPartyIdentifierValue | The additional identifier value for a party. | No prerequisite tasks | Required | Required
IssuingAuthorityName | The name of the authority that issued the additional identifier. | No prerequisite tasks | Required, if you are not providing the Issue Authority Party ID. | The attribute is neither required nor conditionally required.
IssuingAuthorityPartyId | The party ID of the issuing authority. | Identify the party ID value by exporting the Party object using Setup and Maintenance, Manage Additional Identifier Types task. | Required, if you are not providing the Issue Authority Party Name. | The attribute is neither required nor conditionally required.

**Note:** It is recommended that you include the issue date and expiry date for identifiers that become invalid after the expiry date, such as passport and driving license.
The following figure shows a sample attribute mapping to import a new additional identifier for an existing organization. In this scenario, the origin system and origin system reference information are used to identify the existing party and the new additional identifier.

The following figure shows a sample attribute mapping to update an existing additional identifier. In this scenario, the origin system reference information is used to identify the additional identifier and update its description.

Using Reference Files to Evaluate Attributes

For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

Additional Tips

Tip: You can explore related topics by entering the following keywords in the Oracle Sales Cloud Help Application search.

- Importing contacts
- Importing accounts
- Importing households

Tip: The Identifier Type is defined for particular party types and for uniqueness. For example, when importing an organization profile, the identifier type must allow assignment to the organization party type. You can also specify the value of an identifier type must be unique. For example, you can define that the passport number listed for each person must be unique.
Tip: If your data isn’t from a source system, then you can emulate a system by defining a source system and manually generating reference values for your data. For example, if you periodically gather information at sales events or through sales calls, then you might define a source system, using a code of SALES_ACTIVITY. The format of your reference values might concatenate the date, salesperson name, and sequence number, such as 01_Jan_2014_Jackson_01.

Related Topics

• Identifier Types: Explained

Importing Additional Party Names: Quick Start

This topic describes how to get you started with importing additional party names. It includes the following information:

• An example of how to identify and associate records
• The minimum data required and the prerequisite tasks
• How to access and use reference files to evaluate attributes
• Additional tips

With an additional party name, you can capture other names for an account, contact, household, or legal entity. For example, a person may have a maiden name, and an organization may also be known by a brand name. You can import new additional names or update existing additional name records, using the File-Based Data Import feature. The following table lists the file-based import objects and target objects. You can import additional names, using these file-based objects.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Party Type</th>
<th>Target Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contact</td>
<td>Person</td>
<td>AdditionalName</td>
</tr>
<tr>
<td>Account</td>
<td>Organization</td>
<td>AdditionalName</td>
</tr>
<tr>
<td>Contact</td>
<td>Person (a contact for an account)</td>
<td>AdditionalName</td>
</tr>
<tr>
<td>Household</td>
<td>Group</td>
<td>AdditionalName</td>
</tr>
<tr>
<td>Legal Entity</td>
<td>Organization</td>
<td>AdditionalName</td>
</tr>
<tr>
<td>Legal Entity</td>
<td>Person (a contact for a legal entity)</td>
<td>ContactAdditionalName</td>
</tr>
</tbody>
</table>

Identifying and Associating Records with Each Other

To add additional names to a party or to update existing additional party name assignments using file-based import, your source file must contain information about the organization, person, or household party to which the additional name is being assigned. You map your source file data to the organization, person, or group profile target object and to the additional party name target object.

If you’re creating the party and the additional party name assignment in the same import activity, then group the party and additional party name data together in your source file. The import process can import the party, and then identify the party record so that it can import the associated additional party name.
To add an additional party name to an existing party, or to update an existing additional party name record, your source file must include values that enable the import process to identify the existing records. These values will be either source system and source system reference value combination, Oracle Sales Cloud internal ID, or public unique identifiers, such as business keys or external IDs.

If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file. The file-based data import process stores a cross-reference between the source system information and Oracle Sales Cloud internal ID. The import process uses the combination of source system and source system reference value to identify the existing record.

**Minimum Data Required and the Prerequisite Setup Tasks**

The minimum data that is required to import additional name information depends on the following:

- The purpose of the import. The data requirements are different when you’re creating both the party and the additional party name in the same import batch, adding new additional names to an existing party, or updating an additional party name record.

- Organization and group names compared with person names. A variety of components that make up a person’s name aren’t used for organization and group names.

- Identifying and associating records. In some cases, you can choose which attributes you want to provide to the import process when identifying and associating records.

The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create or update an additional party name record. You can optionally include attributes that are available for import in your import file but that aren’t listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Additional Party Name Records in the Same Batch</th>
<th>Creating Additional Party Name Record for an Existing Party</th>
</tr>
</thead>
</table>
| PartyId            | The Oracle Sales Cloud internal ID for the party (account, contact, household, or legal entity) to which the additional party name belongs. | Identify the party ID value by exporting the Party object using the Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task. | The attribute is neither required nor conditionally required. | Conditionally required.
|                    |                                                                              | Provide reference information to identify the existing party. The reference information can be: |                                                                                  |                                                             |
|                    |                                                                              | • PartyId (Oracle Sales Cloud internal ID).                                                   |                                                                                  |                                                             |
|                    |                                                                              | • PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile |                                                                                  |                                                             |
| PartyOrigSystem    | The code representing the source system for the party (account, contact, household, or legal entity) that was used when you imported | Identify the source system code that was used when you imported | Conditionally required. | Conditionally required. |
### Attribute | Description | Prerequisite Setup Task | Creating Party and Additional Party Name Records in the Same Batch | Creating Additional Party Name Record for an Existing Party
--- | --- | --- | --- | ---
AdditionalName | The code representing the source system for the additional name. | When you predefine your source system code in the Setup and Maintenance, Manage | Conditionally required. | Conditionally required. |

The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create or update an additional party name record. You can optionally include attributes that are available for import in your import file but that aren’t listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.
### Attribute | Description | Prerequisite Setup Task | Creating Party and Additional Party Name Records in the Same Batch | Creating Additional Party Name Record for an Existing Party
--- | --- | --- | --- | ---
AdditionalName | The reference number or text representing the source system unique ID for the additional name. | Identify or generate reference values for each additional party name record. The reference value must be unique for all additional party names for the source system. | Conditionally required. | Conditionally required.  
If the source of your data is a third party or external system, and you intend to import updates to previously imported records from the same system, provide the original source system and source system reference values.  
If the source of your data is a third party or external system, and you intend to import updates to previously imported records from the same system, provide the original source system and source system reference values.  
Conditionally required. | Conditionally required.  
If the source of your data is a third party or external system, and you intend to import updates to previously imported records from the same system, provide the original source system and source system reference values.  
Conditionally required.

ObjectKey | The Oracle Sales Cloud internal ID for the additional name. | Identify the additional party name ID value by exporting the Additional Party Name object, using the Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task. | The attribute is neither required nor conditionally required. | The attribute is neither required nor conditionally required.  
| | | | |  

NameType | The name type to categorize the name. | Review or define the name types for each party type (person, organization, or household), using the Setup and Maintenance, Manage Additional Name Types task. | Required | Required

Name | The additional name value for an organization or group. | No prerequisite tasks | Required | Required

The following figure depicts a sample attribute mapping to import an additional name for an existing party. In this scenario, the OrigSystem and OrigSystemReference values are used to identify the party.
Mapping of Source and Target Objects

Using Reference Files to Evaluate Attributes

For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

Additional Tips

Tip: You can explore related topics by entering the following keywords in Oracle Sales Cloud Help Application.

- Importing additional party names
- Importing accounts
- Importing contacts
- Importing households

Tip: If your data isn’t from a source system, then you can emulate a system by defining a source system and manually generating reference values for your data. For example, if you periodically gather information at sales events or through sales calls, then you might define a source system, using a code of SALES_ACTIVITY. The format of your reference numbers might concatenate the date, salesperson name, and sequence number, such as 01_Jan_2014_Jackson_01.

Tip: The Name Type is defined for particular party types. For example, if you’re importing an organization profile, ensure that the name type definition allows assignment to the Organization party type.

Related Topics

- Additional Name Types: Explained
**Importing Addresses: Quick Start**

This topic describes how to get you started with importing addresses. It includes the following information:

- An example of how to identify and associate records
- The minimum data required and the prerequisite tasks
- How to access and use reference files to evaluate attributes
- Additional tips

An address enables you to capture detailed location information for an account, contact, household, employee resource, and legal entity. For example, a person may have residential and official addresses, and an organization may have different billing and shipping addresses. You can import new addresses or update existing address records, using the File-Based Data Import feature.

The following table lists the file-based import objects and target objects that you can use to import addresses.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Party Type</th>
<th>Target Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account</td>
<td>Organization</td>
<td>Address, SellToAddress</td>
</tr>
<tr>
<td>Account</td>
<td>Person (a contact for a account)</td>
<td>Address, ContactPrimaryAddress</td>
</tr>
<tr>
<td>Household</td>
<td>Household</td>
<td>Address</td>
</tr>
<tr>
<td>Legal Entity</td>
<td>Organization</td>
<td>PrimaryAddress</td>
</tr>
<tr>
<td>Legal Entity</td>
<td>Person (a contact for a legal entity)</td>
<td>ContactPrimaryAddress</td>
</tr>
<tr>
<td>Employee Resource</td>
<td>Person</td>
<td>PrimaryAddress</td>
</tr>
<tr>
<td>Contact</td>
<td>Person</td>
<td>Address, ContactAddress</td>
</tr>
<tr>
<td>Lead</td>
<td>Lead Consumer</td>
<td>PersonAddress</td>
</tr>
<tr>
<td>Lead</td>
<td>Lead Contact</td>
<td>ContactPrimaryAddress</td>
</tr>
<tr>
<td>Partner</td>
<td>Organization</td>
<td>PrimaryAddress</td>
</tr>
<tr>
<td>Partner Contact</td>
<td>Person</td>
<td>PrimaryAddress</td>
</tr>
<tr>
<td>Campaign Response</td>
<td>Organization</td>
<td>OrganizationAddress</td>
</tr>
<tr>
<td>Campaign Response</td>
<td>Consumer Respondent</td>
<td>PersonAddress</td>
</tr>
<tr>
<td>Campaign Response</td>
<td>Campaign Response Contact</td>
<td>ContactPrimaryAddress</td>
</tr>
</tbody>
</table>
Identifying and Associating Records with Each Other

To add addresses to a party or to update existing address assignments using file-based import, your source file must contain information about the account, contact, or household party to which the address belongs. You map your source file data to the account, contact, or household profile target object and to the address target object.

If you’re creating the party and the address assignment in the same import activity, then group the party and address data together in your source file. The import process can import the party, and then identify the party record so that it can import the associated address.

To add an address to an existing party, or to update an existing address record, your source file must include the values that enable the import process to identify the existing records. These values will be source system and source system reference value combination, an Oracle Sales Cloud internal ID, or Public Unique IDs, such as business keys or external IDs.

If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system's record in your source file. The file-based data import process stores a cross-reference between the source system information and Oracle Sales Cloud internal ID. The import process uses the combination of source system and source system reference value to identify the existing record.

You can use external IDs or business keys to identify and associate records with each other when you’re importing a smaller set of records. Business keys are a set of attributes that uniquely identify a party record. For example, a person's last name and first name uniquely identify a person. An external ID is a unique record identifier from a system outside of Oracle Sales Cloud. You can use the following business keys or external IDs in Oracle Sales Cloud:

- Organization name for organizations
- First name and last name for persons
- Contact information, which is a combination of e-mail Id, mobile number, IM, and URL.
- Address information, which is a combination of address1, address2, city, and postal code.

You can configure source systems to identify the source of the data that you’re importing. Source systems are external sources of data that are used to import data into Oracle Sales Cloud. Oracle Sales Cloud uses source system references to create references between source IDs and the Oracle Sales Cloud database IDs. You can configure source systems either by using the Manage Trading Community Source System task, or by importing the source system information into Oracle Sales Cloud using the Source System Reference import object.

When the source of your data isn’t an external system and you don’t intend to regularly update the data, you don’t need the source system information. To import updates to your existing data, you can export the Oracle Sales Cloud internal ID and add it to your source file. The import process uses the internal ID to identify the existing record.

Minimum Data Required and the Prerequisite Setup Tasks

The minimum data that is required to import address information depends on the following:

- The purpose of the import. The data requirements are different when you’re creating both the party and the address in the same import batch, adding addresses to an existing party, or updating an address record.
- Identifying and associating records. In some cases, you can choose which attributes you want to provide to the import process when identifying and associating records.

The values that you provide in your source file may require a setup task or manual step to ensure the information passes the import validation. Before you prepare your source file, complete the prerequisite steps to determine the data values, and then include those values in your source file.
The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create or update an address record. You can optionally include attributes that are available for import in your import file but that aren’t listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Adding Address in the Same Batch</th>
<th>Adding Address to an Existing Party</th>
<th>Updating Existing Addresses</th>
</tr>
</thead>
<tbody>
<tr>
<td>PartyId</td>
<td>The Oracle Sales Cloud internal ID for the party (account, contact, household, or legal entity) to which the address belongs.</td>
<td>Identify the party ID value by exporting the Party object using the Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td>PartyOrigSystem</td>
<td>The code representing the source system for the party (account, contact, household, or legal entity) to which the address belongs.</td>
<td>Identify the source system code that was used when you imported the party in a prior batch, or identify the source system code that you’ll use when importing the party in the same batch as this address.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
</tbody>
</table>

- PartyId (Oracle Sales Cloud internal ID).
- PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.
- PartyNumber (public unique identifier for an organization profile), which is PARTY_SITE_NUMBER for address.

- PartyId (Oracle Sales Cloud internal ID).
- PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.
- PartyNumber (public unique identifier for an organization profile), which is PARTY_SITE_NUMBER for address.
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Adding Address in the Same Batch</th>
<th>Adding Address to an Existing Party</th>
<th>Updating Existing Addresses</th>
</tr>
</thead>
<tbody>
<tr>
<td>PartyOrigSystem Reference</td>
<td>The reference number or text representing the source system unique ID for the party (account, contact, household, or legal entity) to which the address belongs.</td>
<td>Identify the reference value from your source system that was used when you imported the party in a prior batch, or identify the source system reference that you’ll use when importing the party in the same batch as this address.</td>
<td>Conditionally required. If the source of your data is an external system, and you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>Conditionally required. Provide reference information to identify the existing party. The reference information can be:</td>
<td>Conditionally required. Provide reference information to identify the existing party. The reference information can be:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyId (Oracle Sales Cloud internal ID).</td>
<td>• PartyId (Oracle Sales Cloud internal ID).</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
<td>• PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Party Number (public unique identifier for an organization profile), which is PARTY_SITE_NUMBER for address.</td>
<td>• Party Number (public unique identifier for an organization profile), which is PARTY_SITE_NUMBER for address.</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Party and Adding Address in the Same Batch</td>
<td>Adding Address to an Existing Party</td>
<td>Updating Existing Addresses</td>
</tr>
<tr>
<td>---------------</td>
<td>--------------------------------------------------</td>
<td>----------------------------------------------------------------</td>
<td>-----------------------------------------------------</td>
<td>------------------------------------</td>
<td>-----------------------------</td>
</tr>
<tr>
<td>Country</td>
<td>The country code component of the postal address.</td>
<td>Identify valid country codes using the Setup and Maintenance, Manage Territories task.</td>
<td>Required</td>
<td>Required</td>
<td>The attribute is neither required nor conditionally required.</td>
</tr>
</tbody>
</table>
| LocationId1   | The unique ID for the existing location record in the Oracle Sales Cloud destination table. | Identify the LOCATION_ID for an existing location by exporting the Location object using the Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task. | The attribute is neither required nor conditionally required. | The attribute is neither required nor conditionally required. | Conditionally required. Provide reference information to identify the existing party. The reference information can be:  
  - ObjectKey (Oracle Sales Cloud internal ID).  
  - LocationOrigSystem1 and LocationOrigSystemReference1 (source system code and source system reference values) provided when importing the organization profile.  
  - Party Number (public unique identifier for an organization profile), which is PARTY_SITE_NUMBER for address. |

<p>| LocationOrigSystem1 | A source system code that identifies the original source system of the location. | Identify the source system code that was used when you imported the location in a prior batch, or identify the source system code you will use | Conditionally required. If the source of your data is an external system, and if you intend to import updates to | Conditionally required. If the source of your data is an external system, and if you intend to import updates to | Conditionally required. Provide reference information to identify the existing party. |</p>
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Adding Address in the Same Batch</th>
<th>Adding Address to an Existing Party</th>
<th>Updating Existing Addresses</th>
</tr>
</thead>
<tbody>
<tr>
<td>LocationOrigSystemReference1</td>
<td>Original System Reference for Location: A source system reference that identifies the unique ID of the location in your legacy or external system.</td>
<td>Identify the source system reference that was used when you imported the location in a prior batch, or identify the source system code you will use when importing the location in the same batch as this party relationship to the location.</td>
<td>Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>Conditionally required. Provide reference information to identify the existing party. The reference information can be:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **ObjectKey**: (Oracle Sales Cloud internal ID).
- **LocationOrigSystem1 and LocationOrigSystemReference1**: (source system code and source system reference values) provided when importing the organization profile.
- **PartyNumber**: (public unique identifier for an organization profile), which is PARTY_SITE_NUMBER for address.
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Adding Address in the Same Batch</th>
<th>Adding Address to an Existing Party</th>
<th>Updating Existing Addresses</th>
</tr>
</thead>
<tbody>
<tr>
<td>TransliterationLang</td>
<td>The language considered as the local language for the location.</td>
<td>Identify valid language codes using the Setup and Maintenance, Manage ISO Languages task.</td>
<td>A value is required if you're providing address elements in a translated language.</td>
<td>A value is required if you're providing address elements in a translated language.</td>
<td>A value is required if you're providing address elements in a translated language.</td>
</tr>
<tr>
<td>Identifying AddressFlag</td>
<td>Indicates that an address is the primary address for a party.</td>
<td>No prerequisite tasks.</td>
<td>A value is required if you're adding or updating an address association and where the import default value or lack of a value in the source file results in no address being identified as the primary address for the party.</td>
<td>A value is required if you're adding or updating an address association and where the import default value or lack of a value in the source file results in no address being identified as the primary address for the party.</td>
<td>A value is required if you're adding or updating an address association and where the import default value or lack of a value in the source file results in no address being identified as the primary address for the party.</td>
</tr>
<tr>
<td>RelationshipId</td>
<td>The unique identifier for the existing relationship record in the Oracle Sales Cloud destination table. The relationship is between two parties, such as an organization and an organization's contact. You provide the relationship if the context of the address association is for a relationship.</td>
<td>Identify the relationship ID for an existing relationship by exporting the Relationship and Organization Contact objects using the Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>A value is required if you're adding an address association to an existing party relationship or updating an existing address association record for a relationship, and you're not providing the relationship's source system reference (RelOrigSystemReference) and the source system code (RelOrigSystem).</td>
<td>A value is required if you're adding an address association to an existing party relationship or updating an existing address association record for a relationship, and you're not providing the relationship's source system reference (RelOrigSystemReference) and the source system code (RelOrigSystem).</td>
<td>A value is required if you're adding an address association to an existing party relationship or updating an existing address association record for a relationship, and you're not providing the relationship's source system reference (RelOrigSystemReference) and the source system code (RelOrigSystem).</td>
</tr>
<tr>
<td>RelOrigSystem</td>
<td>The source system code that identifies the original source system of the party relationship.</td>
<td>No prerequisite tasks.</td>
<td>A value is required if you're adding an address association to an existing party relationship and you're not providing the relationship ID, or if you're importing the relationship in</td>
<td>A value is required if you're adding an address association to an existing party relationship and you're not providing the relationship ID, or if you're importing the relationship in</td>
<td>A value is required if you're adding an address association to an existing party relationship and you're not providing the relationship ID, or if you're importing the relationship in</td>
</tr>
</tbody>
</table>

- Party Number (public unique identifier for an organization profile), which is PARTY_SITE_NUMBER for address.
## Attribute Summary

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Adding Address in the Same Batch</th>
<th>Adding Address to an Existing Party</th>
<th>Updating Existing Addresses</th>
</tr>
</thead>
<tbody>
<tr>
<td>RelOrigSystemReference</td>
<td>The ID that identifies the party relationship in your legacy or external system.</td>
<td>No prerequisite tasks.</td>
<td>A value is required if you're adding an address association to an existing party relationship and you’re not providing the relationship ID, or if you're importing the relationship in the same import batch and you're using the source system reference and source system code to relate the source file data.</td>
<td>A value is required if you’re adding an address association to an existing party relationship and you’re not providing the relationship ID, or if you’re importing the relationship in the same import batch and you’re using the source system reference and source system code to relate the source file data.</td>
<td>A value is required if you’re updating a previously imported address association, where you provided a SiteOrigSystem value, and you’re not providing the address association ID (ObjectKey), or if you’re importing associated objects, such as contact preferences and site uses, and are</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Party and Adding Address in the Same Batch</td>
<td>Adding Address to an Existing Party</td>
<td>Updating Existing Addresses</td>
</tr>
<tr>
<td>--------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-------------------------</td>
<td>---------------------------------------------------</td>
<td>-----------------------------------</td>
<td>-------------------------------</td>
</tr>
<tr>
<td>SiteOrigSystem Reference</td>
<td>A source system reference that identifies the unique ID of the address association (party site) in your legacy or external system.</td>
<td>No prerequisite tasks.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>A value is required if you're updating a previously imported address association, where you provided a SiteOrigSystem value, and you're not providing the address association ID (ObjectKey), or if you're importing associated objects, such as contact preferences and site uses, and are using the source system reference and source system code to relate the source file data.</td>
</tr>
<tr>
<td>ObjectKey</td>
<td>The unique ID for the existing address association record (party site) in the Oracle Sales Cloud destination table.</td>
<td>Identify the party site ID for an existing address association by exporting the Party Site object using the Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>A value is required if you're updating an existing address association and are not providing the source system reference and source system code (SiteOrigSystemReference and SiteOrigSystem).</td>
</tr>
</tbody>
</table>

There are four sets of reference information that you can use to refer to a record. However, you must use only one of them to refer to a record; else, the import process would result in an error. The import process recognizes reference information in this order:

1. Oracle Sales Cloud ID such as PartyId, ObjectKey, etc.
2. Public User ID such as organization name, contact name, etc.
3. Origin system and origin system reference
4. Other reference information such as business keys.
The following figure depicts a sample attribute mapping to import an address for an existing party. In this scenario, the OrigSystem and OrigSystemReference values are used to identify the party.

Using Reference Files to Evaluate Attributes

For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

Additional Tips

Tip: You can explore related topics by entering the following keywords in Oracle Sales Cloud Help Application search. You can access help from any Oracle Sales Cloud page by clicking Help in the top-right corner, and selecting Applications Help.

- Importing addresses
- Importing location information
- Importing contacts
- Importing accounts
- Importing households
- Importing legal entities

Tip: If your data isn’t from a source system, then you can emulate a system by defining a source system and manually generating reference values for your data. For example, if you periodically gather information at sales events or through sales calls, then you might define a source system, using a code of SALES_ACTIVITY. The format of your reference numbers might concatenate the date, salesperson name, and sequence number, such as 01_Jan_2014_Jackson_01.
Importing Address Purposes: Quick Start

This topic describes how to get you started with importing address purposes. It includes the following information:

- An example of how to identify and associate records
- The minimum data required and the prerequisite tasks
- How to access and use reference files to evaluate attributes
- Additional tips

An address purpose enables you to capture functions performed by each address. For example, assign the bill-to address purpose to account sites designated to receive and process bills. Similarly, assign the ship-to address purpose to account sites designated to receive goods purchased by the customer account.

The following table lists the file-based import objects and target objects that you can use to import address purposes.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Party Type</th>
<th>Target Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contact</td>
<td>Person</td>
<td>ContactAddressPurpose</td>
</tr>
<tr>
<td>Account</td>
<td>Organization</td>
<td>AddressPurpose</td>
</tr>
<tr>
<td>Account</td>
<td>Person (a contact for an account)</td>
<td>ContactAddressPurpose</td>
</tr>
<tr>
<td>Household</td>
<td>Household</td>
<td>AddressPurpose</td>
</tr>
<tr>
<td>Household</td>
<td>Person (a contact for a household)</td>
<td>ContactAddressPurpose</td>
</tr>
<tr>
<td>Campaign Response</td>
<td>Consumer Respondent</td>
<td>AddressPurpose</td>
</tr>
<tr>
<td>Campaign Response</td>
<td>Campaign Response Contact</td>
<td>ContactAddressPurpose</td>
</tr>
<tr>
<td>Legal Entity</td>
<td>Organization</td>
<td>AddressPurpose</td>
</tr>
<tr>
<td>Legal Entity</td>
<td>Person (a contact for a legal entity)</td>
<td>ContactAddressPurpose</td>
</tr>
<tr>
<td>Lead</td>
<td>Lead Consumer</td>
<td>AddressPurpose</td>
</tr>
<tr>
<td>Lead</td>
<td>Lead Contact</td>
<td>ContactAddressPurpose</td>
</tr>
<tr>
<td>Partner</td>
<td>Organization</td>
<td>AddressPurpose</td>
</tr>
</tbody>
</table>

Identifying and Associating Records with Each Other

To add address purposes to a party or update existing address purpose assignments using file-based import, your source file must contain information about the address and the account, contact, or household party to which the address purpose
belongs. You map your source file data to the account, contact, or household profile target object and to the address purpose target object.

If you’re creating the address and the address purpose assignment in the same import activity, then group the address and address purpose data together in your source file. The import can import the address, and then identify the address record so that it can import the associated address purpose.

To add an address purpose to an existing address, or to update an existing address purpose record, your source file must include values that enable the import process to identify the existing records. These values will be source system and source system reference value combination, Oracle Sales Cloud internal ID, or public unique identifiers, such as business keys or external IDs.

If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file. The file-based data import process stores a cross-reference between the source system information and Oracle Sales Cloud internal ID. The import process uses the combination of source system and source system reference value to identify the existing record.

You can use external IDs or business keys to identify and associate records with each other when you’re importing a smaller set of records. Business keys are a set of attributes that uniquely identify a party record. For example, a person’s last name and first name uniquely identify a person. An external ID is a unique record identifier from a system outside of Oracle Sales Cloud. You can use the following business keys or external IDs in Oracle Sales Cloud:

- Organization name for organizations
- First name and last name for persons
- Contact information, which is a combination of e-mail Id, mobile number, IM, and URL.
- Address information, which is a combination of address1, address2, city, and postal code.

You can configure source systems to identify the source of the data that you’re importing. Source systems are external sources of data that are used to import data into Oracle Sales Cloud. Oracle Sales Cloud uses source system references to create references between source IDs and the Oracle Sales Cloud database IDs. You can configure source systems either by using the Manage Trading Community Source System task, or by importing the source system information into Oracle Sales Cloud using the Source System Reference import object.

When the source of your data isn’t an external system and you don’t intend to regularly update the data, you don’t need the source system information. To import updates to your existing data, you can export the Oracle Sales Cloud internal ID and add it to your source file. The import process uses the internal ID to identify the existing record.

**Minimum Data Required and the Prerequisite Setup Tasks**

The minimum data that is required to import address purpose information depends on the following:

- The purpose of the import. The data requirements are different when you’re creating both the address and the address purpose in the same import batch, adding address purposes to an existing address, or updating an address purpose record.
- Identifying and associating records. In some cases, you can choose which attributes you want to provide to the import process when identifying and associating records.

The values that you provide in your source file may require a setup task or manual step to ensure the information passes the import validation. Before you prepare your source file, complete the prerequisite steps to determine the data values, and then include those values in your source file.

The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create or update an address purpose record. You can optionally include attributes that are available for import in your import file but that aren’t listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Adding Address Purpose to an Existing Party</th>
<th>Adding Address Purpose to an Existing Party</th>
<th>Updating Existing Address Purposes</th>
</tr>
</thead>
<tbody>
<tr>
<td>PartyId</td>
<td>The Oracle Sales Cloud internal ID for the party (account, contact, household, or legal entity) to which the address belongs.</td>
<td>Identify the party ID value by exporting the Party object using the Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>Required.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• PartyId (Oracle Sales Cloud internal ID).</td>
<td>• PartyId (Oracle Sales Cloud internal ID).</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• PartyOrigSystem and PartyOrigSysteml (source system code and source system reference values) provided when importing the organization profile.</td>
<td>• PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• A business key that is a combination of PARTY_SITE_ID, SITE_USE_TYPE, START_DATE, and END_DATE range.</td>
<td>• A business key that is a combination of PARTY_SITE_ID, SITE_USE_TYPE, START_DATE, and END_DATE range.</td>
<td></td>
</tr>
<tr>
<td>PartyOrigSystem</td>
<td>The code representing the source system for the party (account, contact, household, or legal entity) to which the address belongs.</td>
<td>Identify the source system code that was used when you imported the party in a prior batch, or identify the source system code that you'll use when importing the party in the same batch as this address.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• PartyId (Oracle Sales Cloud internal ID).</td>
<td>• PartyId (Oracle Sales Cloud internal ID).</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• PartyOrigSystem and PartyOrigSysteml (source system code and source system reference values) provided when importing the organization profile.</td>
<td>• PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• A business key that is a combination of PARTY_SITE_ID, SITE_USE_TYPE, START_DATE, and END_DATE range.</td>
<td>• A business key that is a combination of PARTY_SITE_ID, SITE_USE_TYPE, START_DATE, and END_DATE range.</td>
<td></td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Party and Adding Address Purpose in the Same Batch</td>
<td>Adding Address Purpose to an Existing Party</td>
<td>Updating Existing Address Purposes</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>--------------------------</td>
<td>-------------------------------------------------------------</td>
<td>--------------------------------------------</td>
<td>-----------------------------------</td>
</tr>
<tr>
<td>PartyOrigSystem Reference</td>
<td>The reference number or text representing the source system unique ID for the party (account, contact, household, or legal entity) to which the address purpose belongs.</td>
<td>Identify the reference value from your source system that was used when you imported the party in a prior batch, or identify the source system reference that you'll use when importing the party in the same batch as this additional party name.</td>
<td>Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system's record in your source file.</td>
<td>Conditionally required. Provide reference information to identify the existing party. The reference information can be: • PartyId (Oracle Sales Cloud internal ID). • PartyOrigSystem and PartyOrigSystemId (source system code and source system reference values) provided when importing the organization profile. • A business key that is a combination of PARTY_SITE_ID, SITE_USE_TYPE, START_DATE, and END_DATE range.</td>
<td>Conditionally required. Provide reference information to identify the existing party. The reference information can be: • PartyId (Oracle Sales Cloud internal ID). • PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile. • A business key that is a combination of PARTY_SITE_ID, SITE_USE_TYPE, START_DATE, and END_DATE range.</td>
</tr>
<tr>
<td>SiteUseType</td>
<td>The lookup code that represents the site use type using the Setup</td>
<td>Required</td>
<td>Required</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Party and Adding Address Purpose in the Same Batch</td>
<td>Adding Address Purpose to an Existing Party</td>
<td>Updating Existing Address Purposes</td>
</tr>
<tr>
<td>----------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
<td>------------------------------------------</td>
<td>-----------------------------------</td>
</tr>
<tr>
<td>PartySiteId</td>
<td>The unique ID for the existing address association record (party site) in the Oracle Sales Cloud destination table. The party site represents the association of a physical location to an organization, person, or to a relationship between two parties, such as an organization’s contact.</td>
<td>and Maintenance, Manage Trading Community Location Lookups task.</td>
<td>Identify the party site ID for an existing address association by exporting the Party Site object using the Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>Conditionally required.</td>
</tr>
</tbody>
</table>
| SiteOrigSystem       | A source system code that identifies the original source system of the address association (party site). The party site represents the association of a physical location to an organization, person, or to a relationship between two parties, such as an organization’s contact. |                                                                                  | Identify the source system code that was used when you imported the address in a prior batch, or identify the source system code that you’ll use when importing the address in the same batch as this address purpose. | Conditionally required. | Conditionally required. | Provide reference information to identify the existing party. The reference information can be:  
  - ObjectKey (Oracle Sales Cloud internal ID).  
  - SiteuseOrigSystem and SiteuseNameOrigSystemRef (source system code and source system reference values) provided when importing the organization profile.  
  - A business key that is a combination of PARTY_SITE_ID, SITE_USE_TYPE, START_DATE, and END_DATE range. |
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Adding Address Purpose in the Same Batch</th>
<th>Adding Address Purpose to an Existing Party</th>
<th>Updating Existing Address Purposes</th>
</tr>
</thead>
</table>
| SiteOrigSystem Reference   | A source system reference that identifies the unique ID of the address association (party site) in your legacy or external system. The party site represents the association of a physical location to an organization, person, or to a relationship between two parties, such as an organization's contact. | Identify the reference value from your source system that was used when you imported the address in a prior batch, or identify the source system reference that you'll use when importing the address in the same batch as this address purpose. | Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system's record in your source file. | Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system's record in your source file. | Conditionally required. Provide reference information to identify the existing party. The reference information can be:  
- ObjectKey (Oracle Sales Cloud internal ID).  
- SiteOrigSystem and SiteNameOrigSystemRef (source system code and source system reference values) provided when importing the organization profile.  
- A business key that is a combination of PARTY_SITE_ID, SITE_USE_TYPE, START_DATE, and END_DATE range. |
### Attribute Description

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Adding Address Purpose in the Same Batch</th>
<th>Adding Address Purpose to an Existing Party</th>
<th>Updating Existing Address Purposes</th>
</tr>
</thead>
</table>
| **SiteuseOrigSystem**         | A source system code that identifies the original source system of the party’s address purpose. | Identify the source system code that was used when you imported the address purpose in a prior batch, or identify the source system code that you’ll use when importing the address purpose in the same batch as this address purpose. | Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file. | Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file. | Conditionally required. Provide reference information to identify the existing party. The reference information can be:  
  - **ObjectKey** (Oracle Sales Cloud internal ID).  
  - **SiteuseOrigSystem** and **SiteuseNameOrigSystemReference** (source system code and source system reference values) provided when importing the organization profile.  
  - A business key that is a combination of **PARTY_SITE_ID**, **SITE_USE_TYPE**, **START_DATE**, and **END_DATE** range. |
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Adding Address Purpose in the Same Batch</th>
<th>Adding Address Purpose to an Existing Party</th>
<th>Updating Existing Address Purposes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>System's record in your source file.</td>
<td>System's record in your source file.</td>
<td>SiteuseNameOrgSystemRef (source system code and source system reference values) provided when importing the organization profile.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ObjectKey,</td>
<td>The unique ID for the existing address purpose record in the Oracle Sales</td>
<td>No prerequisite tasks.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td>PartySiteUseId</td>
<td>Cloud destination table.</td>
<td></td>
<td></td>
<td></td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• ObjectKey (Oracle Sales Cloud internal ID).</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• SiteuseOrigSystem and SiteuseNameOrgSystemRef (source system code and source system reference values) provided when importing the organization profile.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• A business key that is a combination of PARTY_SITE_ID, SITE_USE_TYPE, START_DATE, and END_DATE range.</td>
</tr>
</tbody>
</table>
There are four sets of reference information that you can use to refer to a record. However, you must use only one of them to refer to a record; else, the import process would result in an error. The import process recognizes reference information in this order:

1. Oracle Sales Cloud ID such as PartyId, ObjectKey, etc.
2. Public User ID such as organization name, contact name, etc.
3. Origin system and origin system reference
4. Other reference information such as business keys.

The following figure depicts a sample attribute mapping to import an address purpose for an existing party. In this scenario, the OrigSystem and OrigSystemReference values are used to identify the party.

Using Reference Files to Evaluate Attributes

For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.
Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

Additional Tips
Tip: You can explore related topics by entering the following keywords in Oracle Sales Cloud Help Application search. You can access help from any Oracle Sales Cloud page by clicking Help in the top-right corner, and selecting Applications Help.

- Importing addresses
- Importing address purposes
- Importing accounts
- Importing contacts
- Importing households

Tip: If your data isn’t from a source system, then you can emulate a system by defining a source system and manually generating reference values for your data. For example, if you periodically gather information at sales events or through sales calls, then you might define a source system, using a code of SALES_ACTIVITY. The format of your reference numbers might concatenate the date, salesperson name, and sequence number, such as 01_Jan_2014_Jackson_01.

Importing Classifications: Quick Start
This topic describes how to get you started with importing classifications. It includes the following information:

- An example of how to identify and associate records
- The minimum data required and the prerequisite tasks
- How to access and use reference files to evaluate attributes
- Additional tips

With classifications you can categorize entities, such as organizations, persons, and households. You define various classification categories, eligibility rules, and the classification values. For example, you define a classification category for parties interested in renewable energy types, including a solar energy classification value. Another classification category represents industries, such as automotive and banking. You can classify a party as interested in solar energy and conducts business in the automotive industry.

The following table lists the file-based import objects and target objects that you can use to import classifications.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Party Type</th>
<th>Target Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account</td>
<td>Organization</td>
<td>Classification</td>
</tr>
<tr>
<td>Legal Entity</td>
<td>Organization</td>
<td>Classification</td>
</tr>
<tr>
<td>Contact</td>
<td>Person</td>
<td>Classification</td>
</tr>
<tr>
<td>Consumer</td>
<td>Organization</td>
<td>Classification</td>
</tr>
<tr>
<td>Household</td>
<td>Household</td>
<td>Classification</td>
</tr>
</tbody>
</table>
Identifying and Associating Records with Each Other

To add classifications to a party or update existing classification assignments using file-based import, your source file must contain information about the organization, person, or household party to which the classification is being assigned. You map your source file data to the organization, person, or group profile target object and to the classification target object.

If you’re creating the party and the classification assignment in the same import activity, then group the party and classification data in your source file. The import process can import the party, and identify the party record so that it can import the associated classifications.

To add a classification to an existing party or to update an existing classification record, your source file must include values that enable the import process to identify the existing records. These values will be either source system and source system reference value combination, Oracle Sales Cloud internal ID, or public unique identifiers, such as business keys or external IDs.

If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file. The file-based data import process stores a cross-reference between the source system information and Oracle Sales Cloud internal ID. The import process uses the combination of source system and source system reference value to identify the existing record.

Minimum Data Required and the Prerequisite Setup Tasks

The minimum data that is required to import classification information depends on the following:

- The purpose of the import. The data requirements are different when you’re creating both the party and the classification in the same import batch, adding classifications to an existing party, or updating a classification record.
- Identifying and associating records. The data requirements are different when you’re using source system information or record IDs to identify and associate records.

The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create or update a classification record. You can optionally include attributes that are available for import in your import file but that aren’t listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Classification Records in the Same Batch</th>
<th>Creating Classification Record for an Existing Party</th>
</tr>
</thead>
<tbody>
<tr>
<td>PartyId</td>
<td>The Oracle Sales Cloud internal ID for the party (account, consumer, contact, household, or legal entity) to which the classification belongs.</td>
<td>Identify the party ID value by exporting the Party object using Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>Conditionally required</td>
</tr>
</tbody>
</table>

Provide reference information to identify the existing party. The reference information can be:

- PartyId (Oracle Sales Cloud internal ID).
- PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing.
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Classification Records in the Same Batch</th>
<th>Creating Classification Record for an Existing Party</th>
</tr>
</thead>
<tbody>
<tr>
<td>PartyOrigSystem</td>
<td>The code representing the source system for the party (account, consumer, contact, household, or legal entity) to which the classification belongs.</td>
<td>Identify the source system code that was used when you imported the party in a prior batch, or identify the source system code you’ll use when importing the party in the same batch as this classification.</td>
<td>Conditionally required</td>
<td>Conditionally required</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyId (Oracle Sales Cloud internal ID).</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Business keys such as class_category and class_code.</td>
</tr>
<tr>
<td>PartyOrigSystem Reference</td>
<td>The reference number or text representing the source system unique ID for the party (account, consumer, contact, household, or legal entity) to which the classification belongs.</td>
<td>Identify the reference value from your source system that was used when you imported the party in a prior batch, or identify the source system reference you’ll use when importing the party in the same batch as this classification.</td>
<td>Conditionally required</td>
<td>Conditionally required</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyId (Oracle Sales Cloud internal ID).</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Business keys such as class_category and class_code.</td>
</tr>
</tbody>
</table>
The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create or update classification record. You can optionally include attributes that are available for import in your import file but that aren't listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Classification Records in the Same Batch</th>
<th>Creating Classification Record for an Existing Party</th>
</tr>
</thead>
<tbody>
<tr>
<td>ClassificationOrigSystem</td>
<td>The code representing the source system for the classification.</td>
<td>When you predefine your source system code in the Setup and Maintenance, Manage Trading Community Source Systems task, select the Enabled for Trading Community Members check box.</td>
<td>Conditionally required</td>
<td>Conditionally required</td>
</tr>
<tr>
<td>ClassificationOrigSystemReference</td>
<td>The reference value for the classification record in the external source system.</td>
<td>Identify or generate reference values for each party classification association record. The reference value must be unique for all party classification associations for the source system.</td>
<td>Conditionally required</td>
<td>Conditionally required</td>
</tr>
<tr>
<td>ObjectKey</td>
<td>The Oracle Sales Cloud internal ID for the party classification association.</td>
<td>Identify the party classification association ID by exporting the Classification Assignment object, using the Setup and Maintenance, Manage Trading Community Source Systems task.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>The attribute is neither required nor conditionally required.</td>
</tr>
<tr>
<td>ClassificationCode</td>
<td>The classification code within classification category.</td>
<td>The code should be valid for the classification category. Review the classification codes valid for each classification category, using the Manage Classification Categories, Setup and Maintenance task.</td>
<td>Required</td>
<td>Required</td>
</tr>
</tbody>
</table>
There are four sets of reference information that you can use to refer to a record. However, only set of reference information is sufficient to identify a record. If you have mentioned multiple reference sets that has conflict, then it results in an error. The import process recognizes reference information in this order:

1. Oracle Sales Cloud ID such as PartyId, ObjectKey, etc.
2. Public User ID such as PartyNumber (also called RegistryID), PartySiteNumber, etc.
3. Origin system and Origin system reference
4. Other reference information such as business keys

The following figure shows a sample attributes mapping to import a new classification for an existing organization. In this scenario, the origin system and origin system reference information identifies the new classification and the existing organization.

The following figure shows a sample attribute mapping to update an existing classification of an existing organization, by providing the origin system and origin system reference information.
Using Reference Files to Evaluate Attributes

For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

Additional Tips

Tip: You can explore related topics by entering the following keywords in the Oracle Sales Cloud Help Application search.

- Importing contacts
- Importing accounts
- Importing households
- Importing legal entities
- Importing classification codes

Review the entity assignment rules for the classification category to verify whether the import object meets the assignment criteria.

If your data isn’t from a source system, then you can emulate a system by defining a source system and manually generating reference values for your data. For example, if you periodically gather information at sales events or through sales calls, then you might define a source system, using a code of SALES_ACTIVITY. The format of your reference values might concatenate the date, salesperson name, and sequence number, such as 01_Jan_2014_Jackson_01.

Related Topics

- Classifications: Explained

Importing Contact Points: Quick Start

This topic describes how to get you started with importing contact points. It includes the following information:

- An overview of contact points
• An example of how to identify and associate records
• The minimum data required and the prerequisites tasks
• How to access and use reference files to evaluate attributes
• Additional Tips

Note: Information for associating addresses with persons and organizations is included in this topic, although contact points don’t formally include addresses.

The following table lists the file-based import objects and target objects that you can use to import contact points directly associated with the person.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Party Type</th>
<th>Target Objects for Importing Contact Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contact</td>
<td>Person</td>
<td>Email, Fax, InstantMessenger, Mobile, Phone, Address</td>
</tr>
<tr>
<td>Account</td>
<td>Organization</td>
<td>PrimaryPhone, Fax, Url, Address</td>
</tr>
<tr>
<td>Household</td>
<td>Organization</td>
<td>E-mail, Fax, InstantMessenger, Mobile, Phone, Address</td>
</tr>
<tr>
<td>Legal Entity</td>
<td>Organization</td>
<td>Fax, PrimaryPhone, PrimaryAddress, Url</td>
</tr>
<tr>
<td>Lead</td>
<td>Organization (lead organization)</td>
<td>OrganizationProfileFax, OrganizationProfilePrimaryPhone, OrganizationAddress, Url</td>
</tr>
<tr>
<td>Lead</td>
<td>Person (lead individual)</td>
<td>Email, PersonProfileFax, InstantMessenger, Mobile, PersonAddress, PersonProfilePrimaryPhone</td>
</tr>
<tr>
<td>Lead</td>
<td>Person (contact at lead organization)</td>
<td>ContactEmail, ContactFax, ContactInstantMessenger, ContactMobile, ContactPrimaryAddress, ContactPrimaryPhone</td>
</tr>
<tr>
<td>Partner</td>
<td>Organization</td>
<td>Fax, PrimaryPhone, PrimaryAddress, Url</td>
</tr>
<tr>
<td>Partner Contact</td>
<td>Person (contact for a partner)</td>
<td>Email, Fax, InstantMessenger, Mobile, PersonPrimaryPhone</td>
</tr>
<tr>
<td>Campaign Response</td>
<td>Organization (respondent organization)</td>
<td>OrganizationProfileFax, OrganizationProfilePrimaryPhone, OrganizationAddress, Url</td>
</tr>
<tr>
<td>Campaign Response</td>
<td>Person (individual respondent)</td>
<td>Email, PersonProfileFax, InstantMessenger, Mobile, PersonProfilePrimaryPhone, PersonAddress</td>
</tr>
<tr>
<td>Campaign Response</td>
<td>Person (contact at respondent organization)</td>
<td>ContactEmail, ContactFax, ContactInstantMessenger,</td>
</tr>
</tbody>
</table>
The following table lists the file-based import objects and target objects that you can use to import contact points for a contact relationship.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Party Type</th>
<th>Target Objects for Importing Contact Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contact</td>
<td>Person (a contact for another root object such as account)</td>
<td>ContactEmail, ContactFax, ContactInstantMessenger, ContactMobile, ContactAddress, ContactPhone</td>
</tr>
<tr>
<td>Account</td>
<td>Person (a contact for an account)</td>
<td>ContactEmail, ContactFax, ContactInstantMessenger, ContactMobile, ContactPrimaryAddress, ContactPrimaryPhone</td>
</tr>
<tr>
<td>Legal Entity</td>
<td>Person (contact for a legal entity)</td>
<td>ContactEmail, ContactFax, ContactInstantMessenger, ContactMobile, ContactPrimaryAddress, ContactPrimaryPhone</td>
</tr>
</tbody>
</table>

**Contact Points Overview**

A contact is an individual who is a customer or a prospect, or a contact for an existing customer or consumer, or a contact that does not yet have an established business association with a customer or consumer. Thus, a contact could be an employee of a customer organization, a person you may have met who could help with your business, or a prospective or current individual customer. The contact object contains information that identifies the contact and offers the contact points of the contact. Contact points can be geographical addresses, phone numbers, e-mail IDs, URLs, messenger IDs, and so on. The contact object also contains contact preference information for the contact.

**Identifying and Associating Records with Each Other**

To add contact to a party or to update existing contact point assignments using file-based import, your source file must contain information about the account, contact, or household party to which the contact point belongs. You map your source file data to the account, contact, or household profile target object, the relationship record that captures the interconnection between the contact and the account.

If you’re creating the party and the contact point assignment in the same import activity, then group the party and contact point data together in your source file. The import process can import the party, and then identify the party record so that it can import the associated contact point.

To add a contact point to an existing party, or to update an existing contact point record, your source file must include values that enable the import process to identify the existing records. These values will be source system and source system reference value combination, an Oracle Sales Cloud internal ID, or public unique identifiers, such as business keys or external IDs.

If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system's record in your source file. The file-based data import process stores a cross-reference between the source system information and Oracle Sales Cloud internal ID. The import process uses the combination of source system and source system reference value to identify the existing record.
You can use external IDs or business keys to identify and associate records with each other when you’re importing a smaller set of records. Business keys are a set of attributes that uniquely identify a party record. For example, a person’s last name and first name uniquely identify a person. An external ID is a unique record identifier from a system outside of Oracle Sales Cloud. You can use the following business keys or external IDs in Oracle Sales Cloud:

- Organization name for organizations
- First name and last name for persons
- Contact information, which is a combination of email Id, mobile number, IM, and URL.
- Address information, which is a combination of address1, address2, city, and postal code.

You can configure source systems to identify the source of the data that you’re importing. Source systems are external sources of data that are used to import data into Oracle Sales Cloud. Oracle Sales Cloud uses source system references to create references between source IDs and the Oracle Sales Cloud database IDs. You can configure source systems either by using the Manage Trading Community Source System task, or by importing the source system information into Oracle Sales Cloud using the Source System Reference import object.

When the source of your data isn’t an external system and you don’t intend to regularly update the data, you don’t need the source system information. To import updates to your existing data, you can export the Oracle Sales Cloud internal ID and add it to your source file. The import process uses the internal ID to identify the existing record.

**Minimum Data Required and the Prerequisite Setup Tasks**

The minimum data that is required to import contact point information depends on the following:

- The purpose of the import. The data requirements are different when you’re creating both the party and the contact point in the same import batch, adding contact points to an existing party, or updating a contact point record.
- Identifying and associating records. In some cases, you can choose which attributes you want to provide to the import process when identifying and associating records.

The values that you provide in your source file may require a setup task or manual step to ensure the information passes the import validation. Before you prepare your source file, complete the prerequisite steps to determine the data values, and then include those values in your source file.

The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create or update a contact point data record. You can optionally include attributes that are available for import in your import file but that aren’t listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Adding Contact Points in the Same Batch</th>
<th>Adding Contact Point Records to an Existing Party</th>
<th>Updating Contact Point Records</th>
</tr>
</thead>
</table>
| PartyId   | The Oracle Sales Cloud internal ID for the party (account, contact, household, partner, or legal entity) to which the contact point belongs. | Identify the party ID value by exporting the Party object using Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task. | The attribute is neither required nor conditionally required. | Conditionally required. | Provide reference information to identify the existing party. The reference information can be:  
- PartyId (Oracle Sales Cloud internal ID). |

| PartyId   | The Oracle Sales Cloud internal ID for the party (account, contact, household, partner, or legal entity) to which the contact point belongs. | Identify the party ID value by exporting the Party object using Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task. | The attribute is neither required nor conditionally required. | Conditionally required. | Provide reference information to identify the existing party. The reference information can be:  
- PartyId (Oracle Sales Cloud internal ID). |

| PartyId   | The Oracle Sales Cloud internal ID for the party (account, contact, household, partner, or legal entity) to which the contact point belongs. | Identify the party ID value by exporting the Party object using Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task. | The attribute is neither required nor conditionally required. | Conditionally required. | Provide reference information to identify the existing party. The reference information can be:  
- PartyId (Oracle Sales Cloud internal ID). |

| PartyId   | The Oracle Sales Cloud internal ID for the party (account, contact, household, partner, or legal entity) to which the contact point belongs. | Identify the party ID value by exporting the Party object using Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task. | The attribute is neither required nor conditionally required. | Conditionally required. | Provide reference information to identify the existing party. The reference information can be:  
- PartyId (Oracle Sales Cloud internal ID). |
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Adding Contact Points in the Same Batch</th>
<th>Adding Contact Point Records to an Existing Party</th>
<th>Updating Contact Point Records</th>
</tr>
</thead>
<tbody>
<tr>
<td>PartyOrigSystem</td>
<td>The code representing the source system for the party (account, contact, household, partner, or legal entity) to which the contact point belongs.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Identify the source system code that was used when you imported the party in a prior batch, or identify the source system code that you will use when importing the party in the same batch as this contact point.</td>
<td>Conditionally required.</td>
<td>If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Party and Adding Contact Points in the Same Batch</td>
<td>Adding Contact Point Records to an Existing Party</td>
<td>Updating Contact Point Records</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>--------------------------</td>
<td>-----------------------------------------------------------</td>
<td>--------------------------------------------------</td>
<td>---------------------------------</td>
</tr>
<tr>
<td>PartyOrigSystemReference</td>
<td>The reference number or text representing the source system unique ID for the party (account, contact, household, partner, or legal entity) to which the contact point belongs.</td>
<td>Identify the reference value from your source system that was used when you imported the party in a prior batch, or identify the source system reference that you will use when importing the party in the same batch as this contact point.</td>
<td>Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>Conditionally required. Provide reference information to identify the existing party. The reference information can be: &lt;ul&gt;&lt;li&gt;PartyId (Oracle Sales Cloud internal ID).&lt;/li&gt;&lt;li&gt;PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.&lt;/li&gt;&lt;li&gt;PartyNumber (public unique identifier for an organization profile), which is CONTACT_NUMBER for contact.&lt;/li&gt;&lt;/ul&gt;</td>
<td>Conditionally required. Provide reference information to identify the existing party. The reference information can be: &lt;ul&gt;&lt;li&gt;PartyId (Oracle Sales Cloud internal ID).&lt;/li&gt;&lt;li&gt;PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.&lt;/li&gt;&lt;li&gt;PartyNumber (public unique identifier for an organization profile), which is CONTACT_NUMBER for contact.&lt;/li&gt;&lt;/ul&gt;</td>
</tr>
<tr>
<td>ObjectKey</td>
<td>The unique ID for the existing contact point record in the Oracle Sales Cloud destination table.</td>
<td>Identify the contact point ID for an existing contact point by exporting the contact point object using the Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>Conditionally required. Provide reference information to identify the existing party. The reference information can be: &lt;ul&gt;&lt;li&gt;ObjectKey (Oracle Sales Cloud contact object ID)&lt;/li&gt;&lt;/ul&gt;</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Party and Adding Contact Points in the Same Batch</td>
<td>Adding Contact Point Records to an Existing Party</td>
<td>Updating Contact Point Records</td>
</tr>
<tr>
<td>--------------------</td>
<td>------------------------------------------</td>
<td>------------------------------------------------------------------------------------------</td>
<td>---------------------------------------------------------------</td>
<td>--------------------------------------------------</td>
<td>-------------------------------</td>
</tr>
<tr>
<td>ContactPointType</td>
<td>Type of the contact point like phone, email and so on.</td>
<td>Identify valid lookup codes for the COMMUNICATION_TYPE lookup using the Setup and Maintenance, Manage Contact Point Lookups task.</td>
<td>Required</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>CpOrigSystem</td>
<td>A source system code that identifies the original source system of the contact point.</td>
<td>Identify the source system code that was used when you imported the contact point in a prior batch, or identify the source system code that you'll use when importing the contact point in the same batch as this contact point contact preference.</td>
<td>Conditionally required.</td>
<td>If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>Conditionally required.</td>
</tr>
</tbody>
</table>

Provide reference information to identify the existing party. The reference information can be:

- **ObjectKey** (Oracle Sales Cloud internal ID).
- **CpOrigSystem** and **CpOrigSystemReference** (source system code and source system reference values) provided when importing the organization profile.
- **Party Number** (public unique identifier for an organization profile), which is CONTACT_NUMBER for contact.
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Adding Contact Points in the Same Batch</th>
<th>Adding Contact Point Records to an Existing Party</th>
<th>Updating Contact Point Records</th>
</tr>
</thead>
<tbody>
<tr>
<td>CpOrigSystemReference</td>
<td>A source system reference that identifies the unique ID of the contact point in your legacy or external system.</td>
<td>Identify the reference value from your source system that was used when you imported the contact point in a prior batch, or identify the source system reference that you’ll use when importing the contact point in the same batch as this contact point contact preference.</td>
<td>Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>Conditionally required. Provide reference information to identify the existing party. The reference information can be: Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
</tr>
</tbody>
</table>

- **CpOrigSystemReference:** A source system reference that identifies the unique ID of the contact point in your legacy or external system.
- **CpOrigSystem and CpOrigSystemReference:** Source system code and source system reference values provided when importing the organization profile.
- **Party Number:** (public unique identifier for an organization profile), which is CONTACT_NUMBER for contact.
There are four sets of reference information that you can use to refer to a record. However, you must use only one of them to refer to a record; else, the import process would result in an error. The import process recognizes reference information in this order:

1. Oracle Sales Cloud ID such as PartyId, ObjectKey, and so on.
2. Public User ID such as organization name, contact name, and so on.
3. Origin system and origin system reference
4. Other reference information such as business keys.

The following figure depicts a sample attribute mapping to import a contact point for an existing party. In this scenario, the OrigSystem and OrigSystemReference values identify the party.

Using Reference Files to Evaluate Attributes
For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

Additional Tips
Tip: You can explore related topics by entering the following keywords in Oracle Sales Cloud Help Application search. You can access help from any Oracle Sales Cloud page by clicking Help in the top-right corner, and selecting Applications Help.

- Importing contacts
- Importing accounts
- Importing households
- Importing legal entities
- Importing employee resources

Tip: If your data isn’t from a source system, then you can emulate a system by defining a source system and manually generating reference values for your data. For example, if you periodically gather information at sales events or through sales
calls, then you might define a source system, using a code of SALES_ACTIVITY. The format of your reference numbers might concatenate the date, salesperson name, and sequence number, such as 01_Jan_2014_Jackson_01.

### Importing Contact Point Preferences: Quick Start

This topic describes how to get you started with importing contact point preferences. It includes the following information:

- An overview of contact point preferences
- An example of how to identify and associate records
- The minimum data required and the prerequisites tasks
- How to access and use reference files to evaluate attributes
- Additional Tips

> **Note:** Information for associating addresses with persons and organizations is included in this topic, although contact points don’t formally include addresses.

The following table lists the file-based import objects and target objects that you can use to import contact point directly associated with the person.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Party Type</th>
<th>Target Objects for Importing Contact Point Preferences</th>
</tr>
</thead>
</table>
| Contact       | Person                             | EmailContactPreference, FaxContactPreference, and so on.  
|               |                                    | InstantMessenger has no contact point preference.     |
| Account       | Organization                       | PrimaryPhoneContactPreference, FaxContactPreference, AddressContactPreference  
|               |                                    | Url has no contact point preference.                  |
| Household     | Organization                       | EmailContactPreference, FaxContactPreference, and so on.  
|               |                                    | InstantMessenger and Url have no contact point preference. |
| Legal Entity  | Organization                       | FaxContactPreference, PrimaryPhoneContactPreference, PrimaryAddressContactPreference  
|               |                                    | Url has no contact point preference.                  |
| Lead (organization) | Organization (lead organization) | No contact point preferences                       |
| Lead (individual) | Person (lead individual)         | No contact point preferences                       |
| Lead (contact at lead organization) | Person (contact at lead organization) | No contact point preferences |
Using File-Based Import

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Party Type</th>
<th>Target Objects for Importing Contact Point Preferences</th>
</tr>
</thead>
<tbody>
<tr>
<td>Partner</td>
<td>Organization</td>
<td>FaxContactPreference, PrimaryPhoneContactPreference, PrimaryAddressContactPreference</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Url has no contact point preference.</td>
</tr>
<tr>
<td>Partner Contact</td>
<td>Person (contact for a partner)</td>
<td>EmailContactPreference, FaxContactPreference, and so on.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>InstantMessenger has no contact point preference.</td>
</tr>
<tr>
<td>Campaign Response</td>
<td>Organization (respondent organization)</td>
<td>No contact point preferences.</td>
</tr>
<tr>
<td>Campaign Response</td>
<td>Person (individual respondent)</td>
<td>No contact point preferences.</td>
</tr>
<tr>
<td>Campaign Response</td>
<td>Person (contact at respondent organization)</td>
<td>No contact point preferences.</td>
</tr>
</tbody>
</table>

The following table lists the file-based import objects and target objects that you can use to import contact point preferences directly associated with a contact relationship.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Party Type</th>
<th>Target Objects for Importing Contact Point Preferences</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contact</td>
<td>Person (a contact for another root object such as account)</td>
<td>ContactEmailContactPreference, ContactFaxContactPreference, and so on.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ContactInstantMessenger has no contact point preference.</td>
</tr>
<tr>
<td>Account</td>
<td>Person (a contact for an account)</td>
<td>ContactEmailContactPreference, ContactFaxContactPreference, and so on.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ContactInstantMessenger has no contact point preference.</td>
</tr>
<tr>
<td>Legal Entity</td>
<td>Person (contact for a legal entity)</td>
<td>ContactEmailContactPreference, ContactFaxContactPreference, and so on.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ContactInstantMessenger has no contact point preference.</td>
</tr>
</tbody>
</table>

**Contact Point Preferences Overview**

Every contact has contact points, such as phone numbers, faxes, e-mail IDs, and so on, and each contact point can have associated contact point preferences. Contact point preferences indicate whether a contact point can be used to contact a person. For example, Consider Pinnacle Flowers and Gifts, a customer of Vision Enterprises. At Pinnacle Flowers and Gifts, the designated contact person for the company must not be contacted by phone, and the customer contact for the logistics service that Vision Enterprises offers does not want to be contacted using faxes. Pinnacle Flowers and Gifts has assigned...
phone and fax numbers to most of its contacts and has made these numbers available to its vendors, but the designated contacts have specific preferences about how they do and do not want to be contacted. Vision Enterprises must store these preferences to ensure that their employees do not reach out to customer contacts using contact points that the contacts do not want to be used.

Identifying and Associating Records with Each Other

To add contact point preferences to a party or to update existing contact point preference assignments using file-based import, your source file must contain information about the account, contact, or household party to which the contact point preference belongs. You map your source file data to the account, contact, or household profile target object, the relationship record that captures the interconnection between the contact point and the contact point preference target object.

If you’re creating the party and contact point preference assignment in the same import activity, then group the party and contact point preference data together in your source file. The import process can import the party, and then identify the party record so that it can import the associated contact point preference. To add a contact point preference to an existing party, or to update an existing contact point preference record, your source file must include values that enable the import process to identify the existing records.

If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file. The file-based data import process stores a cross-reference between the source system information and Oracle Sales Cloud internal ID. The import process uses the combination of source system and source system reference value to identify the existing record.

You can configure source systems to identify the source of the data that you’re importing. Source systems are external sources of data that are used to import data into Oracle Sales Cloud. Oracle Sales Cloud uses source system references to create references between source IDs and the Oracle Sales Cloud database IDs. You can configure source systems either by using the Manage Trading Community Source System task, or by importing the source system information into Oracle Sales Cloud using the Source System Reference import object.

When the source of your data isn’t an external system and you don’t intend to regularly update the data, you don’t need the source system information. To import updates to your existing data, you can export the Oracle Sales Cloud internal ID and add it to your source file. The import process uses the internal ID to identify the existing record.

Minimum Data Required and the Prerequisite Setup Tasks

The minimum data that is required to import contact point information depends on the following:

- The purpose of the import. The data requirements are different when you’re creating both the party and contact point preference in the same import batch, adding contact point preferences to an existing party, or updating a contact preference record.
- Identifying and associating records. In some cases, you can choose which attributes you want to provide to the import process when identifying and associating records.

The values that you provide in your source file may require a setup task or manual step to ensure the information passes the import validation. Before you prepare your source file, complete the prerequisite steps to determine the data values, and then include those values in your source file.

The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create or update a contact point preference data record. You can optionally include attributes that are available for import in your import file but that aren’t listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Adding Contact Point Preferences in the Same Batch</th>
<th>Adding Contact Point Preference Records to an Existing Party</th>
<th>Updating Contact Point Preference Records</th>
</tr>
</thead>
<tbody>
<tr>
<td>PartyId</td>
<td>The Oracle Sales Cloud internal ID for the party (account, contact, household, partner, or legal entity) to which the contact point preference belongs.</td>
<td>Identify the party ID value by exporting the Party object using Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyId (Oracle Sales Cloud internal ID).</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyOrigSystem and PartyOrigSysteml (source system code and source system reference values) provided when importing the organization profile.</td>
<td></td>
</tr>
<tr>
<td>PartyOrigSystem</td>
<td>The code representing the source system for the party (account, contact, household, partner, or legal entity) to which the contact point preference belongs.</td>
<td>Identify the source system code that was used when you imported the party in a prior batch, or identify the source system code that you will use when importing the party in the same batch as this contact point preference.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyId (Oracle Sales Cloud internal ID).</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyOrigSystem and PartyOrigSysteml (source system code and source system reference values) provided when importing the organization profile.</td>
<td></td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Party and Adding Contact Point Preferences in the Same Batch</td>
<td>Adding Contact Point Preference Records to an Existing Party</td>
<td>Updating Contact Point Preference Records</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>--------------------------</td>
<td>-----------------------------------------------------------------------</td>
<td>-------------------------------------------------------------</td>
<td>---------------------------------------------</td>
</tr>
<tr>
<td>PartyOrigSystemReference</td>
<td>The reference number or text representing the source system unique ID for the party (account, contact, household, partner, or legal entity) to which the contact point preference belongs.</td>
<td>Identify the reference value from your source system that was used when you imported the party in a prior batch, or identify the source system reference that you will use when importing the party in the same batch as this contact point preference.</td>
<td>Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system's record in your source file.</td>
<td>Conditionally required. Provide reference information to identify the existing party. The reference information can be: • PartyId (Oracle Sales Cloud internal ID). • PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
<td>Conditionally required. Provide reference information to identify the existing party. The reference information can be: • PartyId (Oracle Sales Cloud internal ID). • PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
</tr>
<tr>
<td>ContactType</td>
<td>The method of contact that should or should not be made, such as call or mail.</td>
<td>Identify the contact type from the Setup and Maintenance, Manage Trading Community Common Lookups task.</td>
<td>Required</td>
<td>Required</td>
<td>The attribute is neither required nor conditionally required.</td>
</tr>
<tr>
<td>PreferenceCode</td>
<td>The lookup code that represents the contact method preference phrase, such as Do Use, Opt In, Do not use.</td>
<td>Identify the preference code from the Setup and Maintenance, Manage Trading Community Common Lookups task.</td>
<td>Required</td>
<td>Required</td>
<td>The attribute is neither required nor conditionally required.</td>
</tr>
<tr>
<td>PreferenceStartDate</td>
<td>The date from which a contact preference is valid. The contact preference period start date must be less than or equal to the contact preference period end date.</td>
<td>The contact preference period start date must be less than or equal to the contact preference period end date.</td>
<td>Required</td>
<td>Required</td>
<td>The attribute is neither required nor conditionally required.</td>
</tr>
<tr>
<td>RequestedBy</td>
<td>The lookup code that represents the source of the contact</td>
<td>Identify lookup codes for the source of the contact</td>
<td>Required</td>
<td>Required</td>
<td>The attribute is neither required</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Party and Adding Contact Point Preferences in the Same Batch</td>
<td>Adding Contact Point Preference Records to an Existing Party</td>
<td>Updating Contact Point Preference Records</td>
</tr>
<tr>
<td>----------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------</td>
<td>----------------------------------------------------------</td>
<td>------------------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>preference request from the Setup and Maintenance, Manage Trading Community Common Lookups task.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ContactPrefOrigSysRef</td>
<td>The source system reference that identifies the unique ID of the contact preference in your legacy or external system.</td>
<td>No prerequisite tasks.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ContactPrefOrigSystem</td>
<td>A source system code that identifies the original source system of the contact preference.</td>
<td>No prerequisite tasks.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ObjectKey (Oracle Sales Cloud internal ID).</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AdditionalPartyId and AdditionalPartyIdOrigSystem (source system code and source system reference values) provided when importing the organization profile.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ObjectKey (Oracle Sales Cloud internal ID).</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AdditionalPartyIdOrigSystem (source system code and source system reference values) provided when importing the organization profile.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Party and Adding Contact Point Preferences in the Same Batch</td>
<td>Adding Contact Point Preference Records to an Existing Party</td>
<td>Updating Contact Point Preference Records</td>
</tr>
<tr>
<td>-----------------</td>
<td>------------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------</td>
<td>------------------------------------------</td>
</tr>
<tr>
<td>ObjectKey</td>
<td>The unique ID of the existing contact preference record in the Oracle Sales Cloud destination table.</td>
<td>Identify the contact preference ID of an existing contact preference by exporting the Contact Preference object using the Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td>CpOrigSystem</td>
<td>A source system code that identifies the original source system of the contact point.</td>
<td>Identify the source system code that was used when you imported the contact point in a prior batch, or identify the source system code that you’ll use when importing the contact point in the same batch as this contact point contact preference.</td>
<td>Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>Conditionally required.</td>
</tr>
</tbody>
</table>

Provide reference information to identify the existing party. The reference information can be:

- **ObjectKey** (Oracle Sales Cloud internal ID).
- **CpOrigSystem** and **CpOrigSystemReference** (source system code and source system reference values) provided when importing the organization profile.
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Adding Contact Point Preferences in the Same Batch</th>
<th>Adding Contact Point Preference Records to an Existing Party</th>
<th>Updating Contact Point Preference Records</th>
</tr>
</thead>
<tbody>
<tr>
<td>CpOrigSystemReference</td>
<td>A source system reference that identifies the unique ID of the contact point in your legacy or external system.</td>
<td>Identify the reference value from your source system that was used when you imported the contact point in a prior batch, or identify the source system reference that you’ll use when importing the contact point in the same batch as this contact point contact preference.</td>
<td>Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>Conditionally required. Provide reference information to identify the existing party. The reference information can be:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>- ObjectKey (Oracle Sales Cloud internal ID).</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>- CpOrigSystem and CpOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
</tr>
</tbody>
</table>
The following figure depicts a sample attribute mapping to import a contact point preference for an existing party. In this scenario, the OrigSystem and OrigSystemReference values identify the party.

Using Reference Files to Evaluate Attributes
For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

Additional Tips
Tip: You can explore related topics by entering the following keywords in Oracle Sales Cloud Help Application search. You can access help from any Oracle Sales Cloud page by clicking Help in the top-right corner, and selecting Applications Help.

- Importing contacts
- Importing accounts
- Importing households
- Importing legal entities
• Importing employee resources

Tip: If your data isn’t from a source system, then you can emulate a system by defining a source system and manually generating reference values for your data. For example, if you periodically gather information at sales events or through sales calls, then you might define a source system, using a code of SALES_ACTIVITY. The format of your reference numbers might concatenate the date, salesperson name, and sequence number, such as 01_Jan_2014_Jackson_01.

Importing Contact Jobs: Quick Start

This topic describes how to get you started with importing contact jobs. It includes the following information:

• An example of how to identify and associate records
• The minimum data required and the prerequisite tasks
• How to access and use reference files to evaluate attributes
• Additional tips

A contact job enables you to capture information related to the work that a contact does. The target object contains attributes that records data such as the name and code of the department where the contact works, the job title of the contact, and so on.

The following table lists the file-based import objects and target objects that you can use to import contact jobs.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Party Type</th>
<th>Target Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contact</td>
<td>Person</td>
<td>ContactJob</td>
</tr>
<tr>
<td>Account</td>
<td>Person (a contact for a account)</td>
<td>ContactJob</td>
</tr>
<tr>
<td>Lead</td>
<td>Lead Contact</td>
<td>ContactJob</td>
</tr>
<tr>
<td>Legal Entity</td>
<td>Person (a contact for a legal entity)</td>
<td>ContactJob</td>
</tr>
<tr>
<td>Partner</td>
<td>Partner Contact</td>
<td>Job</td>
</tr>
<tr>
<td>Response</td>
<td>Response Contact</td>
<td>ContactJob</td>
</tr>
</tbody>
</table>

Identifying and Associating Records with Each Other

To add contact jobs to a party or to update existing contact job assignments using file-based import, your source file must contain information about the account, contact, or household party to which the contact job belongs. You map your source file data to the account, contact, or household profile target object and to the contact job target object.

If you’re creating the party and the contact job assignment in the same import activity, then group the party and contact job data together in your source file. The import process can import the party, and then identify the party record so that it can import the associated contact job.

To add a contact job to an existing party, or to update an existing contact job record, your source file must include values that enable the import process to identify the existing records.
If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system's record in your source file. The file-based data import process stores a cross-reference between the source system information and Oracle Sales Cloud internal ID. The import process uses the combination of source system and source system reference value to identify the existing record.

You can configure source systems to identify the source of the data that you’re importing. Source systems are external sources of data that are used to import data into Oracle Sales Cloud. Oracle Sales Cloud uses source system references to create references between source IDs and the Oracle Sales Cloud database IDs. You can configure source systems either by using the Manage Trading Community Source System task, or by importing the source system information into Oracle Sales Cloud using the Source System Reference import object.

When the source of your data isn’t an external system and you don’t intend to regularly update the data, you don’t need the source system information. To import updates to your existing data, you can export the Oracle Sales Cloud internal ID and add it to your source file. The import process uses the internal ID to identify the existing record.

Minimum Data Required and the Prerequisite Setup Tasks
The minimum data that is required to import contact job information depends on the following:

- The purpose of the import. The data requirements are different when you’re creating both the party and the contact job in the same import batch, adding contact jobs to an existing party, or updating a contact job record.
- Identifying and associating records. In some cases, you can choose which attributes you want to provide to the import process when identifying and associating records.

The values that you provide in your source file may require a setup task or manual step to ensure the information passes the import validation. Before you prepare your source file, complete the prerequisite steps to determine the data values, and then include those values in your source file.

The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create or update a contact job record. You can optionally include attributes that are available for import in your import file but that aren’t listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Adding contact job in the Same Batch</th>
<th>Adding contact job to an Existing Party</th>
<th>Updating Existing Contact Jobs</th>
</tr>
</thead>
<tbody>
<tr>
<td>PartyId</td>
<td>The Oracle Sales Cloud internal ID for the party (account, contact, household, or legal entity) to which the address belongs.</td>
<td>Identify the party ID value by exporting the Party object using the Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>Required</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>To identify the party, provide either the internal ID (PartyId) or both the source system code (PartyOrigSystem) and source system reference value (PartyOrigSystemReference) that were provided in a prior import of the party.</td>
<td>To identify the party, provide either the internal ID (PartyId) or both the source system code (PartyOrigSystem) and source system reference value (PartyOrigSystemReference) that were provided in a prior import of the party.</td>
</tr>
<tr>
<td>PartyOrigSystem</td>
<td>The code representing the source system for the party (account, contact, household, or legal entity)</td>
<td>Identify the source system code that was used when you imported the party in a prior batch, or</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>To identify the party, provide</td>
<td>To identify the party, provide</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Party and Adding contact job in the Same Batch</td>
<td>Adding contact job to an Existing Party</td>
<td>Updating Existing Contact Jobs</td>
</tr>
<tr>
<td>--------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------------</td>
<td>--------------------------------------------------------</td>
<td>----------------------------------------</td>
<td>---------------------------------</td>
</tr>
<tr>
<td>PartyOrigSystemReference</td>
<td>The reference number or text representing the source system unique ID for the party (account, contact, household, or legal entity) to which the address belongs.</td>
<td>Identify the reference value from your source system that was used when you imported the party in a prior batch, or identify the source system reference that you’ll use when importing the party in the same batch as this address.</td>
<td>Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>Conditionally required. To identify the party, provide either the internal ID (PartyId) or both the source system code (PartyOrigSystem) and source system reference value (PartyOrigSystemReference) that were provided in a prior import of the party.</td>
<td>Conditionally required. To identify the party, provide either the internal ID (PartyId) or both the source system code (PartyOrigSystem) and source system reference value (PartyOrigSystemReference) that were provided in a prior import of the party.</td>
</tr>
<tr>
<td>RelationshipId</td>
<td>The unique ID for the existing contact’s relationship with the organization or person record in the destination table.</td>
<td>Identify the Relationship ID for an existing contact’s relationship by exporting the Relationship object using the Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>Conditionally required. To identify the existing contact job record, provide either the internal ID (ObjectKey) or both the source system code (RelOrigSystem) and source system reference value (RelOrigSystemReference) that were provided in a prior import of the party.</td>
</tr>
<tr>
<td>RelOrigSystem</td>
<td>The source system code that identifies the original source system of the contact’s relationship with</td>
<td>No prerequisite tasks.</td>
<td>Conditionally required. If the source of your data is an external system, and if you intend to</td>
<td>Conditionally required. If the source of your data is an external system, and if you intend to</td>
<td>Conditionally required. To identify the existing contact job record, provide either the internal</td>
</tr>
</tbody>
</table>

**Attrribute** | **Description** | **Prerequisite Setup Task** | **Creating Party and Adding contact job in the Same Batch** | **Adding contact job to an Existing Party** | **Updating Existing Contact Jobs** |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>PartyOrigSystemReference</td>
<td>The reference number or text representing the source system unique ID for the party (account, contact, household, or legal entity) to which the address belongs.</td>
<td>Identify the reference value from your source system that was used when you imported the party in a prior batch, or identify the source system reference that you’ll use when importing the party in the same batch as this address.</td>
<td>Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>Conditionally required. To identify the party, provide either the internal ID (PartyId) or both the source system code (PartyOrigSystem) and source system reference value (PartyOrigSystemReference) that were provided in a prior import of the party.</td>
<td>Conditionally required. To identify the party, provide either the internal ID (PartyId) or both the source system code (PartyOrigSystem) and source system reference value (PartyOrigSystemReference) that were provided in a prior import of the party.</td>
</tr>
</tbody>
</table>

**RelationshipId** | **Description** | **Prerequisite Setup Task** | **Creating Party and Adding contact job in the Same Batch** | **Adding contact job to an Existing Party** | **Updating Existing Contact Jobs** |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>RelationshipId</td>
<td>The unique ID for the existing contact’s relationship with the organization or person record in the destination table.</td>
<td>Identify the Relationship ID for an existing contact’s relationship by exporting the Relationship object using the Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>Conditionally required. To identify the existing contact job record, provide either the internal ID (ObjectKey) or both the source system code (RelOrigSystem) and source system reference value (RelOrigSystemReference) that were provided in a prior import of the party.</td>
</tr>
</tbody>
</table>

**RelOrigSystem** | **Description** | **Prerequisite Setup Task** | **Creating Party and Adding contact job in the Same Batch** | **Adding contact job to an Existing Party** | **Updating Existing Contact Jobs** |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>RelOrigSystem</td>
<td>The source system code that identifies the original source system of the contact’s relationship with</td>
<td>No prerequisite tasks.</td>
<td>Conditionally required. If the source of your data is an external system, and if you intend to</td>
<td>Conditionally required. If the source of your data is an external system, and if you intend to</td>
<td>Conditionally required. To identify the existing contact job record, provide either the internal</td>
</tr>
</tbody>
</table>
### Attribute | Description | Prerequisite Setup Task | Creating Party and Adding contact job in the Same Batch | Adding contact job to an Existing Party | Updating Existing Contact Jobs
--- | --- | --- | --- | --- | ---
ReOrigSystemReference | The ID that identifies the contact’s relationship with the organization or person in your legacy or external system. | No prerequisite tasks. | Conditionally required. | Conditionally required. | Conditionally required.

If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.

To identify the existing contact job record, provide either the internal ID (ObjectKey) or both the source system code (RelOrigSystem) and source system reference value (RelOrigSystemReference) that were provided in a prior import of the party.

ContactNumber | The contact job’s unique public identifier value. | Obtain the contact number for an existing contact using the Customers work area. | The attribute is neither required nor conditionally required. | The attribute is neither required nor conditionally required. | Conditionally required.

Required if you are updating a contact job and are not providing the object key.

ObjectKey | The unique ID for the existing contact job record in the destination table. | Identify the contact ID for an existing contact by exporting the Organization Contact object using the Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task. | The attribute is neither required nor conditionally required. | The attribute is neither required nor conditionally required. | Conditionally required.

Required if you are updating a contact job and are not providing the contact number.

---

The following figure depicts a sample attribute mapping to import a contact job for an existing party. In this scenario, the OrigSystem and OrigSystemReference values identify the party.
Using Reference Files to Evaluate Attributes

For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

Additional Tips

Tip: You can explore related topics by entering the following keywords in Oracle Sales Cloud Help Application search. You can access help from any Oracle Sales Cloud page by clicking Help in the top-right corner, and selecting Applications Help.

- Importing contact jobs
- Importing accounts
- Importing contacts
- Importing groups

Tip: If your data isn’t from a source system, then you can emulate a system by defining a source system and manually generating reference values for your data. For example, if you periodically gather information at sales events or through sales calls, then you might define a source system, using a code of SALES_ACTIVITY. The format of your reference numbers might concatenate the date, salesperson name, and sequence number, such as 01_Jan_2014_Jackson_01.

Importing Country Structures Using File-Based Import: Quick Start

This topic describes how to get you started with importing country structures. It includes the following information:

- An example of how to identify and associate records
- The minimum data required and the prerequisite tasks
- How to access and use reference files to evaluate attributes
- Additional tips

With the country structure of a country you can define which geography types are part of the country, and how the geography types are hierarchically related within the country. For example, you can create geography types called State, City, and Postal Code. Then you can rank the State geography type as the highest level within the country, the City as the second level, and the Postal Code as the lowest level within the country structure.

The following table lists the file-based import objects and target objects that you can use to import country structures.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Target Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Country Structure</td>
<td>ImpGeoStructureLevel</td>
</tr>
</tbody>
</table>

Identifying and Associating Records with Each Other

To add country structures to a country, or to update existing country structure values using file-based import, your source file must contain information about the country to which the country structure is being assigned. Also, you need to specify the level number at which the geography types you are importing need to be placed. All countries are, by default, placed at Level
1. Create the country structure by increasing the level number as you go down the country structure. Thus, geography types such as states, provinces, and so on, can be placed at Level 2; districts or counties at Level 3, and so on.

To add an additional identifier to an existing party or to update an existing additional identifier record, your source file must include values that enable the import process to identify the existing records. These values will be either source system and source system reference value combination, Oracle Sales Cloud internal ID, or public unique identifiers, such as business keys or external IDs.

If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system's record in your source file. The file-based data import process stores a cross-reference between the source system information and Oracle Sales Cloud internal ID. The import process uses the combination of source system and source system reference value to identify the existing record.

### Minimum Data Required and the Prerequisite Setup Tasks

The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create or update a country structure record. You can optionally include attributes that are available for import in your import file but that aren't listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Country Structure Record for an Existing Country</th>
</tr>
</thead>
<tbody>
<tr>
<td>CountryCode</td>
<td>The code representing the country to which the country structure belongs.</td>
<td>Identify country codes using the Setup and Maintenance, Manage Geographies task.</td>
<td>Required</td>
</tr>
<tr>
<td></td>
<td>Oracle Sales Cloud uses two-letter ISO country codes to refer to countries.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GeographyType</td>
<td>A code used for internal reference by Oracle Sales Cloud at an administrative level. For example, this administrative code may be COUNTRY, STATE, COUNTY, CITY, etc. You must include either the LevelNumber or GeographyType field to define the hierarchical structure of your data.</td>
<td>No prerequisite tasks</td>
<td>Required</td>
</tr>
<tr>
<td>LevelNumber</td>
<td>The level number for the geography type in the hierarchy. For example, a country has the level of 1 in the hierarchy because it is the top level.</td>
<td>No prerequisite tasks</td>
<td>Required</td>
</tr>
</tbody>
</table>
The following image depicts a sample attribute mapping to import a country structure.

![Map Fields](image)

**Using Reference Files to Evaluate Attributes**

For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

**Additional Tips**

Tip: You can explore related topics by entering the following keywords in Oracle Sales Cloud Help Application search.

- Importing country structures

Tip: For sample country structures and detailed steps for importing country structures, see article 1341174.1 on My Oracle Support.

**Related Topics**

- Geography Structure, Hierarchy, and Validation: How They Fit Together

**Importing Workers, Users, and Resources Using File-Based Data Import: Quick Start**

This topic describes how to get you started with importing workers, users, and resources. It includes the following information:

- An overview of workers, users, and resources
- An example of how to identify and associate records
- The minimum data required and the prerequisite tasks
- How to access and use reference files to evaluate attributes
- Additional tips
The following table lists the file-based import objects and target objects that you can use to import workers, users, and resources.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Target Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Employee Resource</td>
<td>WorkerProfile, ResourceProfile, ResourceOrganizationMembership, ResourceTeamMembership, and PrimaryAddress</td>
</tr>
</tbody>
</table>

Workers, Users, and Resources Overview

The information of a newly hired employee, a contingent worker, or a partner contact is stored in Human Capital Management (HCM) as a worker. The employees, contingent workers, and partner contacts who can be assigned work to accomplish business objectives, such as performing Oracle Sales Cloud related tasks, are called resources.

You can import the worker information using the WorkerProfile target import object. This target import object allows import of the basic person information, such as first name, last name, email address, business unit, and legal entity.

The worker information from HCM is imported into Oracle Sales Cloud as a resource. For example, a worker in HCM can be imported into Oracle Sales Cloud as a sales representative. You can import the resource information using the ResourceProfile target import object. This target object allows import of the basic resource information, such as resource role code, resource role type attributes.

A resource role denotes the function of a resource in an enterprise from the perspective of the deploying company. Resource roles are used to not only describe who a resource is in the enterprise, but also what specific role the resource performs within the context of an organization or team. A resource can have only one role in an resource organization during a given period.

Resource role types organize roles into logical groups. For example, the Partner resource role type defines a set of partner-specific roles, such as partner sales representative and partner sales manager. You can use the Sales resource role type and the Marketing resource role type to categorize the appropriate sales and marketing roles for internal employees or contingent worker resources.

You can define roles that can be assigned to resources within resource organizations using the Setup and Maintenance, Manage Resource Roles task. When you assign specific roles to resources, they automatically receive access to specific business functions, locations, applications, and data within the company.

A resource team is a group of resources formed to accomplish a particular task. A resource team may comprise resource organizations, resources, or both. Resource organizations represent the internal organization and structure of the deploying company. Resource organizations are hierarchically structured, and reporting relationships are derived from the organization hierarchy that is created. You can import the resource organization and resource team membership information using the ResourceOrganizationMembership and ResourceTeamMembership target objects, respectively. For more information about resource team, see the Define Resource Team Information chapter in the Oracle Sales Cloud Implementing Customer Data Management guide.

Identifying and Associating Records with Each Other

To add resource details such as resource role code and resource role type to an existing employee record or to update existing employee record using file-based import, your source file must contain information about the employee.

When you add the resource details or update the resource details to an existing employee using file-based import, group the worker and resource information together in your source file. The import process is then able to first import the worker data, and then identify the employee record to import the associated resource details.
Minimum Data Required and the Prerequisite Setup Tasks

The minimum data that is required to import worker and resource profile information depends on the following:

- The purpose of the import. The data requirements are different when you’re importing the worker data and resource data in the same import batch, or adding resource details to an existing employee record.
- Identifying and associating records. In some cases, you can choose which attributes you want to provide to the import process when identifying and associating records.
- The values that you provide in your source file may require a setup task or manual step to ensure the information passes the import validation. Before you prepare your source file, complete the prerequisite steps to determine the data values, and then include those values in your source file. As part of the prerequisite setup tasks, ensure that the Business Unit, Legal Entity, Resources Roles, and Resource Role Mappings are already created.

The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create resource details or update an existing employee record with the resource details. You can optionally include attributes that are available for import in your import file but that aren't listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Employee Resource Record and Updating Employee Resource Record in the Same Batch</th>
</tr>
</thead>
<tbody>
<tr>
<td>WorkerProfile_BusinessUnit</td>
<td>The business unit name to which the resource is associated.</td>
<td>Identify business unit names using the Setup and Maintenance, Manage Business Unit task.</td>
<td>Required</td>
</tr>
<tr>
<td>WorkerProfile_LegalEntity</td>
<td>The legal entity name of the employer.</td>
<td>Identify legal entity names using the Setup and Maintenance, Manage Legal Entity task.</td>
<td>Required</td>
</tr>
<tr>
<td>WorkerProfile_FirstName</td>
<td>The worker’s first name.</td>
<td>No prerequisite tasks</td>
<td>Required</td>
</tr>
<tr>
<td>WorkerProfile_LastName</td>
<td>The worker’s last name.</td>
<td>No prerequisite tasks</td>
<td>Required</td>
</tr>
<tr>
<td>WorkerProfile_EmailAddress</td>
<td>The primary work email ID of the worker.</td>
<td>No prerequisite tasks</td>
<td>Required</td>
</tr>
<tr>
<td>WorkerProfile_ManagerEmail</td>
<td>The primary work email ID of the worker’s manager.</td>
<td>No prerequisite tasks</td>
<td>Conditionally required. If you want to identify an employee in the resource hierarchy, then you must mention the manager email address.</td>
</tr>
<tr>
<td>WorkerProfile_PersonType</td>
<td>Describes the person type, for example, EMP for employees, CWK for contingent workers.</td>
<td>No prerequisite tasks</td>
<td>Conditionally required. If the source of your data is a contingent worker, then you must mention the person type. The default value of the person type is EMP.</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Employee Resource Record and Updating Employee Resource Record in the Same Batch</td>
</tr>
<tr>
<td>-----------------------------------------------</td>
<td>----------------------------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ResourceProfile__RoleCode</td>
<td>The value for the resource role code, which is predefined in the Manage Resource Roles task.</td>
<td>Identify the employee resource role code by reviewing the employee resource role types</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td>and codes in the Setup and Maintenance, Manage Resource Roles task.</td>
<td>and codes in the Setup and Maintenance, Manage Resource Roles task.</td>
<td>If you want to identify an employee as a resource, then you must mention the resource</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>role code.</td>
</tr>
<tr>
<td>ResourceProfile__RoleTypeCode</td>
<td>The value for the resource role type, which is predefined in the Manage Resource Roles task.</td>
<td>Identify the employee resource role type code by reviewing the employee resource role types</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td>and codes in the Setup and Maintenance, Manage Resource Roles task.</td>
<td>and codes in the Setup and Maintenance, Manage Resource Roles task.</td>
<td>If you want to identify an employee as a resource, then you must mention the resource</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>role type code.</td>
</tr>
<tr>
<td>ResourceOrganizationMembership__TreeCode</td>
<td>The code for the organization hierarchy that the resource is a member of. For employee</td>
<td>No prerequisite tasks</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td>resources, the value is GLOBAL__SALES__MARKETING. For partner contacts (members), the value</td>
<td></td>
<td>If you want to associate a resource to an organization hierarchy to which the resource</td>
</tr>
<tr>
<td></td>
<td>is the code for the partner organization hierarchy.</td>
<td></td>
<td>belongs.</td>
</tr>
<tr>
<td>ResourceOrganizationMembership__ParentTreeCode</td>
<td>The code for the parent of the resource’s organization within the organization hierarchy.</td>
<td>No prerequisite tasks</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td>For employee resources, the value is GLOBAL__SALES__MARKETING. For partner contacts (members),</td>
<td></td>
<td>If you want to associate a resource to an organization hierarchy, then you must</td>
</tr>
<tr>
<td></td>
<td>the value is the code for the partner organization hierarchy.</td>
<td></td>
<td>mention the parent tree code of the resource's organization.</td>
</tr>
<tr>
<td>ResourceOrganizationMembership__OrganizationUsage</td>
<td>The resource organization usage, which is SALESRESOURCE_-ORG for sales,</td>
<td>No prerequisite tasks</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td>PARTNER__RESOURCE_-ORGANIZATION for partners, and MARKETING__RESOURCE_-ORGANIZATION for</td>
<td></td>
<td>If you want to associate a resource to an organization hierarchy, then you must</td>
</tr>
<tr>
<td></td>
<td>marketing.</td>
<td></td>
<td>mention the resource organization usage.</td>
</tr>
<tr>
<td>ResourceOrganizationMembership__OrganizationName</td>
<td>The organization name of the resource.</td>
<td>Identify the organization name of the resource by navigating to the Resource Directory.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If the organization name that you use in the source file doesn’t already exist in the</td>
<td>If you want to associate a resource to an organization hierarchy, then you must</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Resource Directory, then it is created during the employee resource import process.</td>
<td>mention the organization name of the resource.</td>
</tr>
<tr>
<td>ResourceOrganizationMembership__ParentOrganizationName</td>
<td>The organization name of the resource’s manager.</td>
<td>Identify the organization name of the resource’s manager by</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Attribute Mapping for Employee Resource Import

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Employee Resource Record and Updating Employee Resource Record in the Same Batch</th>
</tr>
</thead>
</table>
| Navigating to the Resource Directory.         |             | If you want to associate a resource to an organization hierarchy, then you must mention the organization name of the resource’s manager.

#### Note:
To view the imported workers in the Manage Users task, you must run the Update Person Search Keywords ESS job from the Scheduled Processes work area. You can also choose to run this job automatically by setting the Enable Keyword Crawler parameter to Y in the Manage HCM Configuration for Coexistence task. The crawler updates the person-search keywords during the import process for every worker.

The following figure depicts a sample attribute mapping to import an employee resource.

#### Using Reference Files to Evaluate Attributes
For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

#### Additional Tips
Tip: You can explore related topics by entering the following keywords in Oracle Sales Cloud Help Application search. You can access help from any Oracle Sales Cloud page by clicking Help in the top-right corner, and selecting Applications Help.

- Importing users
- Importing resource teams
• Importing employee resources

Related Topics

• Importing Resource Teams and Resource Team Usages Using File-Based Import: Quick Start

Importing Organization Profiles: Quick Start

This topic describes how to get you started with importing organization profiles.

It includes the following information:

• An example of how to identify and associate records
• The minimum data required and the prerequisites steps
• How to access and use reference files to evaluate attributes
• Additional tips

With organization profile, you can capture organization information for an account, legal entity, and partner. The organization profile contains information for the organization, such as the stock symbol, number of employees, and the CEO’s name. You can import new organization profiles or update existing organization profile records, using the File-Based Data Import feature. The following table lists the file-based import objects and target objects that you can use to import organization profiles. You can import organization profiles, using these file-based objects.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Party Type</th>
<th>Target Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account</td>
<td>Organization</td>
<td>OrganizationProfile</td>
</tr>
<tr>
<td>Legal Entity</td>
<td>Organization</td>
<td>OrganizationProfile</td>
</tr>
<tr>
<td>Lead</td>
<td>Organization</td>
<td>OrganizationProfile</td>
</tr>
<tr>
<td>Marketing Campaign Response</td>
<td>Organization</td>
<td>OrganizationProfile</td>
</tr>
<tr>
<td>Partner</td>
<td>Organization</td>
<td>PartnerInterface</td>
</tr>
</tbody>
</table>

Identifying and Associating Records with Each Other

To add organization profiles or to update existing organization profiles using file-based import, your source file must contain information about the organizations you want to create or update.

To update an existing organization profile, your source file must include the values that enable the import process to identify the existing records. These values will be source system and source system reference value combination, Oracle Sales Cloud internal ID, or public unique identifiers, such as business keys or external IDs.

If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file. The file-based data import process stores a cross-reference between the source system information and
Oracle Sales Cloud internal ID. The import process uses the combination of source system and source system reference value to identify the existing record.

You can use external IDs or business keys to identify and associate records with each other when you’re importing a smaller set of records. Business keys are a set of attributes that uniquely identify a party record. For example, a person’s last name and first name uniquely identify a person. An external ID is a unique record identifier from a system outside of Oracle Sales Cloud. You can use the following business keys or external IDs in Oracle Sales Cloud:

- Organization name for organizations
- First name and last name for persons
- Contact information, which is a combination of email Id, mobile number, IM, and URL.
- Address information, which is a combination of address1, address2, city, and postal code.

You can configure source systems to identify the source of the data that you’re importing. Source systems are external sources of data that are used to import data into Oracle Sales Cloud. Oracle Sales Cloud uses source system references to create references between source IDs and the Oracle Sales Cloud database IDs. You can configure source systems either by using the Manage Trading Community Source System task, or by importing the source system information into Oracle Sales Cloud using the Source System Reference import object.

When the source of your data isn’t an external system and you don’t intend to regularly update the data, you don’t need the source system information. To import updates to your existing data, you can export the Oracle Sales Cloud internal ID and add it to your source file. The import process uses the internal ID to identify the existing record.

Minimum Data Required and the Prerequisite Steps

The minimum data that is required to import organization profile information depends on the following:

- The purpose of the import. The data requirements are different when you’re creating a new organization profile or updating an existing organization profile record.
- Identifying and associating records. In some cases, you can choose which attributes you want to provide to the import process when identifying and associating records.

The values that you provide in your source file may require a setup task or manual step to ensure that the information passes the import validation. Before you prepare your source file, complete the prerequisite steps to determine the data values and then include those values in your source file.

The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create or update an organization profile record. You can optionally include attributes that are available for import in your import file but that aren’t listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Create an Organization Profile</th>
<th>Update an Existing Organization Profile</th>
</tr>
</thead>
<tbody>
<tr>
<td>PartyOrigSystem</td>
<td>The code representing the external source system.</td>
<td>Predefine your source system code as enabled for Trading Community members, using the Setup and Maintenance, Manage Trading Community Source Systems task.</td>
<td>Conditionally required.</td>
<td>Conditionally required</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
<td></td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• PartyId (Oracle Sales Cloud record ID).</td>
<td></td>
<td>• PartyId (Oracle Sales Cloud record ID).</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Create an Organization Profile</td>
<td>Update an Existing Organization Profile</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------------</td>
<td>-------------------------------</td>
<td>----------------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Conditionally required</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Conditionally required</td>
</tr>
<tr>
<td>PartyOrigSystemReference</td>
<td>The reference number or text representing the source system unique ID for the organization.</td>
<td>Identify the reference value from your source system.</td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Conditionally required</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Conditionally required</td>
</tr>
<tr>
<td>OrganizationName</td>
<td>The name of the organization.</td>
<td>No prerequisite tasks</td>
<td>Required</td>
<td>The attribute is neither required nor conditionally required.</td>
</tr>
<tr>
<td>ObjectKey or PartyId</td>
<td>The Oracle Sales Cloud record ID for the organization.</td>
<td>Identify the PartyId by exporting the Party object using the Setup and Maintenance,</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>Conditionally required</td>
</tr>
</tbody>
</table>

- **PartyOrigSystem and PartyOrigSystemReference** (source system code and source system reference values) provided when importing the organization profile.
- **Party Number** (public unique identifier for an organization profile).
- A business key such as organization name.

- **PartyOrigSystem and PartyOrigSystemReference** (source system code and source system reference values) provided when importing the organization profile.
- **Party Number** (public unique identifier for an organization profile).
- A business key such as organization name.
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Create an Organization Profile</th>
<th>Update an Existing Organization Profile</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Manage Bulk Data</strong></td>
<td><strong>Export, Schedule Export Processes</strong> task.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Type</strong></td>
<td>The organization type.</td>
<td>No prerequisite tasks</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>The attribute is neither required nor conditionally required.</td>
</tr>
<tr>
<td></td>
<td>The type is set to Account by default for new organization profiles.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Provide reference information to identify the existing party. The reference information can be:

- **PartyId** (Oracle Sales Cloud record ID).
- **PartyOrigSystem** and **PartyOrigSystemReference** (source system code and source system reference values) provided when importing the organization profile.
- **Party Number** (public unique identifier for an organization profile).
- A business key such as organization name.

The following figure shows a sample attribute mapping to import a new organization. In this scenario, the origin system and origin system reference information are used to identify the organization.

The following figure shows a sample attribute mapping to update an existing organization. In this example, the origin source system and source system reference are used to identify an organization and update the CEO name.
Using Reference Files to Evaluate Attributes

For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

Additional Tips

Tip: You can explore related topics by entering the following keywords in the Oracle Sales Cloud Help Application search.

- Importing sales leads
- Importing legal entities
- Importing contacts
- Importing marketing campaign responses
- Importing partners

Tip: If your data isn’t from a source system, then you can emulate a system by defining a source system and manually generating reference values for your data. For example, if you periodically gather information at sales events or through sales calls, then you might define a source system, using a code of SALES_ACTIVITY. The format of your reference values might concatenate the date, salesperson name, and sequence number, such as 01_Jan_2014_Jackson_01.

Importing Person Profiles: Quick Start

This topic describes how to get you started with importing person profiles. It includes the following information:

- An example of how to identify and associate records
- The minimum data required and the prerequisites steps
- How to access and use reference files to evaluate attributes
- Additional tips

A person party profile enables you to import contacts, and household members. For example, you can use person profiles to import contact information. A person profile contains various attributes, such as first name, last name, and so on. You can import new person profiles or update existing person profile records using the File-Based Data Import feature. The following table lists the file-based import objects and target objects that you can use to import person profiles.
The following table lists the file-based import objects and target objects that you can use to import person profiles.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Target Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contact</td>
<td>PersonProfile</td>
</tr>
<tr>
<td>Account (Account Contact)</td>
<td>PersonProfile, ContactPersonProfile (See note below.)</td>
</tr>
<tr>
<td>Household (Customer Contact)</td>
<td>PersonProfile, ContactPersonProfile (See note below.)</td>
</tr>
<tr>
<td>Legal Entity (Legal Entity Contact)</td>
<td>PersonProfile, ContactPersonProfile (See note below.)</td>
</tr>
<tr>
<td>Employee Resource</td>
<td>PersonProfile</td>
</tr>
<tr>
<td>Lead (Lead Contact)</td>
<td>PersonProfile, ContactPersonProfile (See note below.)</td>
</tr>
<tr>
<td>Partner Contact</td>
<td>PersonProfile</td>
</tr>
<tr>
<td>Marketing Campaign Response (Contact Respondent)</td>
<td>PersonProfile</td>
</tr>
<tr>
<td>Marketing Campaign Response (Marketing Campaign Response Contact)</td>
<td>PersonProfile, ContactPersonProfile (See note below.)</td>
</tr>
</tbody>
</table>

Note: Oracle Sales Cloud supports the ContactPersonProfile target object. However, the object is now deprecated. For updating any contact information, use the PersonProfile target object instead.

Identifying and Associating Records with Each Other

To add person profiles or to update existing person profiles using file-based import, your source file must contain information about the persons.

To update an existing person profile, your source file must include the values that enable the import process to identify the existing records. These values will be source system and source system reference value combination, Oracle Sales Cloud internal ID, or public unique identifiers, such as business keys or external IDs.

If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system's record in your source file. The file-based data import process stores a cross-reference between the source system information and Oracle Sales Cloud internal ID. The import process uses the combination of source system and source system reference value to identify the existing record.

You can use external IDs or business keys to identify and associate records with each other when you’re importing a smaller set of records. Business keys are a set of attributes that uniquely identify a party record. For example, a person’s last name and first name uniquely identify a person. An external ID is a unique record identifier from a system outside of Oracle Sales Cloud. You can use the following business keys or external IDs in Oracle Sales Cloud:

- First name and last name for persons
- Contact information, which is a combination of email Id, mobile number, IM, and URL.
• Address information, which is a combination of address1, address2, city, and postal code.

You can configure source systems to identify the source of the data that you’re importing. Source systems are external sources of data that are used to import data into Oracle Sales Cloud. Oracle Sales Cloud uses source system references to create references between source IDs and the Oracle Sales Cloud database IDs. You can configure source systems either by using the Manage Trading Community Source System task, or by importing the source system information into Oracle Sales Cloud using the Source System Reference import object.

When the source of your data isn’t an external system and you don’t intend to regularly update the data, you don’t need the source system information. To import updates to your existing data, you can export the Oracle Sales Cloud internal ID and add it to your source file. The import process uses the internal ID to identify the existing record.

Minimum Data Required and the Prerequisite Steps

The minimum data that is required to import person profile information depends on the following:

• The purpose of the import. The data requirements are different when you’re creating the person profile or updating an existing person profile record.

• Identifying and associating records. In some cases, you can choose which attributes you want to provide to the import process when identifying and associating records.

The values that you provide in your source file may require a setup task or manual step to ensure that the information passes the import validation. Before you prepare your source file, complete the prerequisite steps to determine the data values and then include those values in your source file.

The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create or update a person profile record. You can optionally include attributes that are available for import in your import file but that aren’t listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Create Party and Person Profiles</th>
<th>Update Person Profile Records</th>
</tr>
</thead>
<tbody>
<tr>
<td>ObjectKey or PartyId</td>
<td>The Oracle Sales Cloud record ID for the party (account, contact, household, or legal entity) to which the person profile belongs.</td>
<td>Identify the party ID value by exporting the Party object using the Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>Conditionally required.</td>
</tr>
</tbody>
</table>

Provide reference information to identify the existing party. The reference information can be:

• PartyId (Oracle Sales Cloud record ID).

• PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.

• Party Number (public unique identifier for an organization profile).
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Create and Person Profiles</th>
<th>Update Person Profile Records</th>
</tr>
</thead>
<tbody>
<tr>
<td>PartyOrigSystem</td>
<td>The code representing the external source system for the party (account, contact, household, or legal entity) to which the person profile belongs.</td>
<td>Predefine your source system code as enabled for parties, using the <strong>Setup and Maintenance, Manage Trading Community Source Systems</strong> task.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td>PartyOrigSystemReference</td>
<td>The reference number or text representing the source system unique ID for the party (account, contact, household, or legal entity) to which the person profile belongs.</td>
<td>Identify the reference value from your source system.</td>
<td>Conditionally required.</td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyId (Oracle Sales Cloud record ID).</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Party Number (public unique identifier for an organization profile).</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• A business key such as organization name.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PartyOrigSystemReference</td>
<td>The reference number or text representing the source system unique ID for the party (account, contact, household, or legal entity) to which the person profile belongs.</td>
<td>Identify the reference value from your source system.</td>
<td>Conditionally required.</td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyId (Oracle Sales Cloud record ID).</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Party Number (public unique identifier for an organization profile).</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• A business key such as person name.</td>
</tr>
</tbody>
</table>
### Attribute Details

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Create Party and Person Profiles</th>
<th>Update Person Profile Records</th>
</tr>
</thead>
<tbody>
<tr>
<td>PersonFirstName</td>
<td>First name of a person party.</td>
<td>No prerequisite tasks</td>
<td>Required</td>
<td>The attribute is neither required nor conditionally required.</td>
</tr>
<tr>
<td>PersonLastName</td>
<td>Last name of a person party.</td>
<td>No prerequisite tasks</td>
<td>Required</td>
<td>The attribute is neither required nor conditionally required.</td>
</tr>
</tbody>
</table>

The following figure depicts a sample attribute mapping to import a new contact. In this scenario, the OrigSystem and OrigSystem values identify the party.
The following figure depicts a sample attribute mapping to update a contact. In this scenario, the OrigSystem and OrigSystem values identify the party and update the person's middle name.

Using Reference Files to Evaluate Attributes

For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

Additional Tips

Tip: You can explore related topics by entering the following keywords in the Oracle Sales Cloud Help Application search.

- Importing person profiles
- Importing contacts
- Importing accounts
- Importing households
- Importing legal entities
- Importing employee resources

Tip: If your data isn’t from a source system, then you can emulate a system by defining a source system and manually generating reference values for your data. For example, if you periodically gather information at sales events or through sales calls, then you might define a source system, using a code of SALES_ACTIVITY. The format of your reference values might concatenate the date, salesperson name, and sequence number, such as 01_Jan_2014_Jackson_01.

Importing Relationships: Quick Start

This topic describes how to get you started with importing relationships.

It includes the following information:

- An example of how to identify and associate records
• The minimum data required and the prerequisite tasks
• How to access and use reference files to evaluate attributes
• Additional tips

The following table lists the file-based import objects and target objects that you can use to import relationships.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Party Type</th>
<th>Target Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account</td>
<td>Organization</td>
<td>Relationship</td>
</tr>
<tr>
<td>Account</td>
<td>Person (a contact for an account)</td>
<td>ContactPersonProfile (See note below.)</td>
</tr>
<tr>
<td>Household</td>
<td>Household</td>
<td>Relationship</td>
</tr>
<tr>
<td>Legal Entity</td>
<td>Person (a contact for a legal entity)</td>
<td>ContactPersonProfile (See note below.)</td>
</tr>
<tr>
<td>Contact</td>
<td>Person</td>
<td>ContactRelationship (See note below.)</td>
</tr>
<tr>
<td>Lead</td>
<td>Lead Contact</td>
<td>ContactPersonProfile (See note below.)</td>
</tr>
<tr>
<td>Partner Contact</td>
<td>Partner Contact</td>
<td>Partner</td>
</tr>
<tr>
<td>Campaign Response</td>
<td>Campaign Response Contact</td>
<td>ContactPersonProfile (See note below.)</td>
</tr>
</tbody>
</table>

With a relationship, you can capture the interconnection between parties such as contacts, households, and so on. For example, you can capture the relationship between an account and the account’s contact person by interconnecting the contact person’s party ID with the account’s organization party ID using the CONTACT relationship type. Similarly, you can associate one or more persons with a household, or family, using the relationship type MEMBER. You can import new relationships or update existing relationship records, using the File-Based Data Import feature.

Identifying and Associating Records with Each Other

To add relationships to a party or update existing relationship assignments using file-based import, your source file must contain data that identifies the subject and object of a relationship. For instance, if you’re creating an ‘is an employee of’ relationship between a person party and an organization, you need to provide reference information that identifies both the subject, the person party, and the object, the organization, parties.

If you’re creating the party and the relationship assignment in the same import activity, then group the party and relationship data together in your source file. The import process can import the party, and then identify the party record so that it can import the associated relationship.

To add an relationship to an existing party, or to update an existing relationship record, your source file must include values that enable the import process to identify the existing records. These values will be either source system and source system reference value combination, Oracle Sales Cloud internal ID, or public unique identifiers, such as business keys or external IDs.

If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file. The file-based data import process stores a cross-reference between the source system information and
Oracle Sales Cloud internal ID. The import process uses the combination of source system and source system reference value to identify the existing record.

**Minimum Data Required and the Prerequisite Setup Tasks**

The minimum that is required to import relationship information depends on the following:

- The purpose of the import. The data requirements are different when you're creating both the party and the relationship in the same import batch, adding relationships to an existing party, or updating an relationship record.
- Identifying and associating records. In some cases, you can choose which attributes you want to provide to the import process when identifying and associating records.

The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create or update an additional party name record. You can optionally include attributes that are available for import in your import file but that aren’t listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Relationship Records in the Same Batch</th>
<th>Creating Relationship Record for an Existing Party</th>
</tr>
</thead>
<tbody>
<tr>
<td>PartyId</td>
<td>The Oracle Sales Cloud internal ID for the party (account, contact, household, or legal entity) to which the relationship belongs.</td>
<td>Identify the party ID value by exporting the Party object using the Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td>PartyOrigSystem</td>
<td>The code representing the source system for the party (account, contact, household, or legal entity) to which the relationship belongs.</td>
<td>Identify the source system code that was used when you imported the party in a prior batch, or identify the source system code that you’ll use when importing the party in the same batch as this relationship.</td>
<td>If the source of your data is a third party or external system, and you intend to import updates to previously imported records from the same system, provide the original source system and source system reference values.</td>
<td>Conditionally required</td>
</tr>
</tbody>
</table>
# Using File-Based Import

The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create or update a relationship record. You can optionally include attributes that are available for import in your import file but that aren’t listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Relationship Records in the Same Batch</th>
<th>Creating Relationship Record for an Existing Party</th>
</tr>
</thead>
<tbody>
<tr>
<td>PartyOrigSystemReference</td>
<td>The reference number or text representing the source system unique ID for the party (account, contact, household, or legal entity) to which the relationship belongs.</td>
<td>Identify the reference value from your source system that was used when you imported the party in a prior batch, or identify the source system reference that you’ll use when importing the party in the same batch as this relationship.</td>
<td>Conditionally required. If the source of your data is a third party or external system, and you intend to import updates to previously imported records from the same system, provide the original source system and source system reference values.</td>
<td>Conditionally required. Provide reference information to identify the existing party. The reference information can be:</td>
</tr>
<tr>
<td>RelationshipCode</td>
<td>Describes the directional connection between the object and subject relationship. For example, Employer of verses Employee of.</td>
<td>You can view the relationship type code definition from the Setup and Maintenance, Manage Relationship Types task.</td>
<td>Conditionally required. If the source of your data is a third party or external system, and you intend to import updates</td>
<td>Conditionally required. If the source of your data is a third party or external system, and you intend to import updates</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Party and Relationship Records in the Same Batch</td>
<td>Creating Relationship Record for an Existing Party</td>
</tr>
<tr>
<td>--------------------</td>
<td>--------------------------------------------------</td>
<td>-------------------------</td>
<td>----------------------------------------------------------</td>
<td>---------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>to previously imported records from the same system, provide the original source system and source system reference values.</td>
<td>to previously imported records from the same system, provide the original source system and source system reference values.</td>
</tr>
<tr>
<td>RelationshipType</td>
<td>Describes the relationship, such as Contact or Membership.</td>
<td>Required</td>
<td>Required</td>
<td>Requirement</td>
</tr>
<tr>
<td>RelOrigSystem</td>
<td>A source system code that identifies the original source system of the relationship between the object and the subject parties.</td>
<td>No prerequisite tasks</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td>RelOrigSystemReference</td>
<td>A source system reference that identifies the original reference ID of the relationship between the object and the subject parties in the source system.</td>
<td>No prerequisite tasks</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td>SubOrigSystem</td>
<td>A source system code that identifies the original source system of the subject party of the relationship.</td>
<td>No prerequisite tasks</td>
<td>This attribute is neither required nor conditionally required</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td>SubOrigSystemReference</td>
<td>A reference ID for the subject of the relationship. If importing from another system, use the unique reference</td>
<td>No prerequisite tasks</td>
<td>This attribute is neither required nor conditionally required</td>
<td>Conditionally required.</td>
</tr>
</tbody>
</table>
There are four sets of reference information that you can use to refer to a record. However, only set of reference information is sufficient to identify a record. If you have mentioned multiple reference sets that has conflict, then it results in an error. The import process recognizes reference information in this order:

1. Oracle Sales Cloud ID such as PartyId, ObjectKey, etc.
2. Public User ID such as PartyNumber (also called RegistryID), PartySiteNumber, etc.
3. Origin system and Origin system reference
4. Other reference information such as business keys

The following figure depicts a sample attribute mapping to import a relationship between two parties. While importing relationships, use the OrigSystem and OrigSystemReference values to identify one party and the SubOrigSystem and SubOrigSystem values to identify the second party.

Using Reference Files to Evaluate Attributes
For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.
Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

Additional Tips

Tip: You can explore related topics by entering the following keywords in Oracle Sales Cloud Help Application search.

- Importing relationships
- Importing contacts
- Importing accounts
- Importing households
- Importing legal entities

Tip: If your data isn’t from a source system, then you can emulate a system by defining a source system and manually generating reference values for your data. For example, if you periodically gather information at sales events or through sales calls, then you might define a source system, using a code of SALES_ACTIVITY. The format of your reference numbers might concatenate the date, salesperson name, and sequence number, such as 01_Jan_2014_Jackson_01.

Importing Sales Profiles: Quick Start

This topic describes how to get you started with importing sales profiles. It includes the following information:

- Identifying and associating records
- The minimum data required and the prerequisites steps
- How to access and use reference files to evaluate attributes
- Additional tips

A sales profile enables you to capture information about a selling relationship with an account or consumer. When a party has one sell-to address, it ceases to be a sales prospect and becomes a sales profile. You can import sales profiles or update existing sales profiles, using the File-Based Data Import feature. The sales profile includes attributes that can be used for assigning territories and providing account ownership details. You must have an organization or person with a sell-to address defined before you can import sales profiles.

The following table lists the file-based import objects and target objects that you can use to import sales profiles.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Party Type</th>
<th>Target Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contact</td>
<td>Person</td>
<td>PersonProfile, SalesProfile</td>
</tr>
<tr>
<td>Account</td>
<td>Organization</td>
<td>OrganizationProfile, SalesProfile</td>
</tr>
<tr>
<td>Account</td>
<td>Household</td>
<td>GroupProfile, SalesProfile</td>
</tr>
</tbody>
</table>

Identifying and Associating Records with Each Other

To add a sales profile to an account or to update an existing sales profile using file-based import, your source file must contain information about the customer organization or consumer person party to which the sales profile belongs. You map your source file data to the organization or person profile target object and to the sales profile target object.
If you’re creating the party and the sales profile in the same import activity, then group the party and sales profile data together in your source file. The import process can import the party and then identify the party record so that it can import the associated sales profile.

To add sales profile to an existing party or to update an existing sales profile record, your source file must include the values that enable the import process to identify the existing records. These values are source system and source system reference value combination, Oracle Sales Cloud internal ID, or public unique identifiers, such as business keys or external IDs.

If the source of your data is an external application, and if you intend to import updates to previously imported records from the external application, then you can provide the source system code and the unique reference value for the source system’s record in your source file. The file-based data import process stores a cross-reference between the source system information and Oracle Sales Cloud internal ID. The import process uses the combination of source system and source system reference value to identify the existing record.

You can use external IDs or business keys to identify and associate records with each other when you’re importing a smaller set of records. Business keys are a set of attributes that uniquely identify a party record. For example, a person’s last name and first name uniquely identify a person. An external ID is a unique record identifier from an application outside of Oracle Sales Cloud. You can use the following business keys or external IDs in Oracle Sales Cloud:

- Organization name for organizations
- First name and last name for persons
- Contact information, which is a combination of e-mail Id, mobile number, IM, and URL.
- Address information, which is a combination of address1, address2, city, and postal code.

You can configure source systems to identify the source of the data that you’re importing. Source systems are external sources of data that are used to import data into Oracle Sales Cloud. Oracle Sales Cloud uses source system references to create references between source IDs and the Oracle Sales Cloud database IDs. You can configure source systems either by using the Manage Trading Community Source System task, or by importing the source system information into Oracle Sales Cloud using the Source System Reference import object.

When the source of your data isn’t an external application and you don’t intend to regularly update the data, you don’t need the source system information. To import updates to your existing data, you can export the Oracle Sales Cloud internal ID and add it to your source file. The import process uses the internal ID to identify the existing record.

**Minimum Data Required and the Prerequisite Steps**

The minimum data that is required to import sales profile information depends on the following:

- The purpose of the import. The data requirements are different when you’re creating both the party and the sales profile in the same import batch, adding new sales profile to an existing party, or updating a sales profile record.
- Identifying and associating records. In some cases, you can select which attributes you want to provide to the import process when identifying and associating records.

The values that you provide in your source file may require a setup task or manual step to ensure that the information passes the import validation. Before you prepare your source file, complete the prerequisite steps to determine the data values and then include those values in your source file.

The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create or update a sales profile record. You can optionally include attributes that are available for import in your import file but that aren’t listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Sales Profile Records in the Same Batch</th>
<th>Creating Sales Profile Record for an Existing Party</th>
<th>Updating Sales Profile Record</th>
</tr>
</thead>
<tbody>
<tr>
<td>PartyOrigSystem</td>
<td>The code representing the source system for the party (account or contact) to which the sales profile belongs.</td>
<td>Identify the source system code that was used when you imported the party in a prior batch, or identify the source system code that you use when importing the party in the same batch as this sales profile.</td>
<td>Conditionally required</td>
<td>Provide the source system and source system reference values if the source of your data is a third party or external application and you intend to import updates to previously imported records from the same system.</td>
<td>Conditionally required</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyId (Oracle Sales Cloud record ID).</td>
<td>• PartyId (Oracle Sales Cloud record ID).</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
<td>• PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Party Number (public unique identifier for an organization profile).</td>
<td>• Party Number (public unique identifier for an organization profile).</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• A business key such as organization name.</td>
<td>• A business key such as organization name.</td>
</tr>
<tr>
<td>PartyOrigSystemReferer</td>
<td>The reference number or text representing the source system unique ID for the party (account or contact) to which the sales profile belongs.</td>
<td>Identify the reference value from your source system that was used when you imported the party in a prior batch, or identify the source system reference you use when importing the party in the same batch as this sales profile</td>
<td>Conditionally required</td>
<td>Provide the source system and source system reference values if the source of your data is a third party or external application and you intend to import updates to previously imported records from the same application.</td>
<td>Conditionally required</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyId (Oracle Sales Cloud record ID).</td>
<td>• PartyId (Oracle Sales Cloud record ID).</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyOrigSystem and PartyOrigSystemReference (source</td>
<td>• PartyOrigSystem and PartyOrigSystemReference (source</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Party and Sales Profile Records in the Same Batch</td>
<td>Creating Sales Profile Record for an Existing Party</td>
<td>Updating Sales Profile Record</td>
</tr>
<tr>
<td>-----------------</td>
<td>------------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------------</td>
<td>----------------------------------------------------------</td>
<td>--------------------------------------------------</td>
<td>----------------------------------</td>
</tr>
<tr>
<td>PartyId</td>
<td>The Oracle Sales Cloud record ID for the party (account or contact) to which the sales profile belongs.</td>
<td>Identify the party ID value by exporting the Party object using Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>Conditionally required.</td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>system code and source system reference values provided when importing the organization profile.</td>
<td>PartyId (Oracle Sales Cloud record ID).</td>
<td>PartyId (Oracle Sales Cloud record ID).</td>
<td>PartyId (Oracle Sales Cloud record ID).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
<td>PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
<td>PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
<td>PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
</tr>
</tbody>
</table>

• PartyId (public unique identifier for an organization profile).
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Sales Profile Records in the Same Batch</th>
<th>Creating Sales Profile Record for an Existing Party</th>
<th>Updating Sales Profile Record</th>
</tr>
</thead>
<tbody>
<tr>
<td>AccountDirectorId</td>
<td>The unique party ID of the resource assigned to manage the sales profile.</td>
<td>Identify the party ID value of the account owner by exporting the Party object using Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>Conditionally required. By default, the user importing the account is designated as the account owner. To specify a different account owner, include one of the reference information for the account owner in the import file: • Party ID of the account owner in the AccountDirectorId column. • Origin system and origin system reference of the account owner in the OWNER_ORIG_S AND OWNER_ORIG_S columns. • Party Number of the account owner in the OWNER_PARTY_ column.</td>
<td>Conditionally required. By default, the user importing the account is designated as the account owner. To specify a different account owner, include one of the reference information for the account owner in the import file: • Party ID of the account owner in the AccountDirectorId column. • Origin system and origin system reference of the account owner in the OWNER_ORIG_S AND OWNER_ORIG_S columns. • Party Number of the account owner in the OWNER_PARTY_ column.</td>
<td>Conditionally required. By default, the user importing the account is designated as the account owner. To specify a different account owner, include one of the reference information for the account owner in the import file: • Party ID of the account owner in the AccountDirectorId column. • Origin system and origin system reference of the account owner in the OWNER_ORIG_S AND OWNER_ORIG_S columns. • Party Number of the account owner in the OWNER_PARTY_ column.</td>
</tr>
</tbody>
</table>

**Note:** During import, the attribute Type is set to Account for all new accounts.
The following figure shows a sample attribute mapping to import a new sales profile for an existing party. In this scenario, the origin system and the origin system reference information are used to identify the existing party and the new sales profile.

**Using Reference Files to Evaluate Attributes**

For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

**Additional Tips**

Tip: You can explore related topics by entering the following keywords in the Oracle Sales Cloud Help Application search.

- Importing contacts
- Importing accounts

Tip: If your data isn’t from a source system, then you can emulate a system by defining a source system and manually generating reference values for your data. For example, if you periodically gather information at sales events or through sales calls, then you might define a source system, using a code of SALES_ACTIVITY. The format of your reference values might concatenate the date, salesperson name, and sequence number, such as 01_Jan_2014_Jackson_01.
3 Using File-Based Export

Overview of File-Based Data Export

File-Based Data Export: Explained

File-Based Data Export lets you export a set of data from Oracle Sales Cloud. You can select a parent object and a set of associated child objects, and output a ZIP file containing a separate file for each object. The output data sets contain only those parent records which match the specified view criteria on the parent/child object and their associated child records for the selected objects. File-Based Data Export supports the multi-CSV file format. This file format is also supported by File-Based Data Import. You can use a combination of File-Based Data Export and File-Based Data Import to perform round-trip export/import.

How Does File-Based Data Export Differ From Bulk Export

File-Based Data Export (FBDE) differs from Bulk Export in both the set of data exported as well as the format of the data that is output. While Bulk Export supports the ability to export data at either the parent or child level, FBDE lets you configure an export operation in which the output data set contains only the child records that are associated to the parent records that match the specified filter criteria.

For example: Bulk Export - Account and Account Contact are selected. However the set of child records exported is dependent upon the filter criteria configured at the child level. FBDE - Account and Account Contact are selected. However the sets of child records exported include only those that are associated to a parent record that matches the specified filter criteria at the parent level. FBDE does not support child level filter criteria.

The multiple CSV file format generated by FBDE consists of a ZIP file that contains one or more CSV files. Each CSV file represents a separate object (either parent or child). The records within the child CSV file are associated to a parent record contained within the parent CSV file. Association from child to parent record is accomplished by a foreign key to the parent record. The foreign key for all child records associated to a parent record is the same. The output from an FBDE operation appears as follows: USER_SPECIFIED_FILENAME.ZIP PARENTOBJECTNAME.CSV CHILDOBJECT1.CSV CHILDOBJECT2.CSV CHILDOBJECTn.CSV. You can’t rename the individual CSV files as part of the FBDE configuration.

Considerations for Exporting Data That You Plan to Import into Sales Cloud Applications

You can export data from one Oracle Sales Cloud instance and then import it into another instance. In this case Oracle recommends that you export all child objects and attributes for the selected parent object as doing so ensures that all required attributes are included in the exported data set. The default values in FBDE for options such as Delimiter, time stamp format and Date format all use the same default values as File Based Data Import so Oracle recommends that the default values be used if the data is going to be imported back into an Oracle Sales Cloud instance.

File Based Data Export supports the ability to export the ID value of a record. This ID value is only valid within the source instance and cannot be used on other Oracle Sales Cloud instances; therefore the ID value for a record should be set to Ignore during the Map Fields step of File Based Data Import when importing data into the target Oracle Sales Cloud instance.

You must ensure that all custom objects and attributes exist in the target instance. If a corresponding custom object or attribute haven’t been created in the target instance, then it must be created before data can be successfully imported into the target instance.
Exporting Your Data Objects

To export your data objects you must complete the following steps:

1. Create a File-Based Data Export activity
2. Schedule the activity

Creating a File-Based Data Export Activity

You must navigate to Setup and Maintenance work area and search for the Manage File Export Activities task. You must create a new export activity, provide the parent object, activity name and file name details.

When configuring a file-based data export, you must consider how the exported data are used. If the data is to be consumed by an external application, then you should select the appropriate time and date format as well as delimiter value to avoid the need to edit the data once it has been exported.

Scheduling the Activity

You can schedule the activity immediately or at a later date. Scheduling at a later date has the following options:

- Incremental - For more information, see Configuring Incremental Export
- One Time Scheduling
- Repeating

For incremental export, you must provide the initial date time of interval, start and end date times, repeat frequency and repeat unit. You must review the activity details and configurations and activate the activity to start the export process. The export process generates the ZIP file containing the CSV files for each child object.

Parent-Child References in File-Based Data Export

When data is exported from Oracle Sales Cloud using File Based Data Export, the CSV files containing the child data contain a reference to the parent object to which they are associated. The specific key used may vary between objects. The following table illustrates the keys used to identify the parent object for each child object supported in File Based Data Export.

Supported Objects and Keys

The following table lists the various parent and child objects, corresponding output file names and parent keys:

<table>
<thead>
<tr>
<th>Parent Object</th>
<th>Child Objects</th>
<th>Output File Name</th>
<th>Parent Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>Activity</td>
<td>Activity Assignees</td>
<td>Activity.Assignee.csv</td>
<td>ACTIVITY_ID</td>
</tr>
<tr>
<td>Activity</td>
<td>Activity Contacts</td>
<td>Activity.Contact.csv</td>
<td>ACTIVITY_ID</td>
</tr>
<tr>
<td>Contract Header</td>
<td>Contract Party</td>
<td>Contract.Party.csv</td>
<td>ID</td>
</tr>
<tr>
<td>Contract Header</td>
<td>Contract Party</td>
<td>Contract.Party.csv</td>
<td>MAJOR_VERSION</td>
</tr>
<tr>
<td>Account Profile</td>
<td>Account Additional Name</td>
<td>AdditionalName.csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Parent Object</td>
<td>Child Objects</td>
<td>Output File Name</td>
<td>Parent Key</td>
</tr>
<tr>
<td>----------------------</td>
<td>-----------------------</td>
<td>-------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Account Profile</td>
<td>Account Address</td>
<td>Address.csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Account Profile</td>
<td>Account Address Purpose</td>
<td>AddressPurpose.csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Account Profile</td>
<td>Account Classification</td>
<td>Classification.csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Account Profile</td>
<td>Account Fax</td>
<td>Fax.csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Account Profile</td>
<td>Account Primary Phone</td>
<td>PrimaryPhone.csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Account Profile</td>
<td>Account Relationship</td>
<td>Relationship.csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Account Profile</td>
<td>Account Sales Profile</td>
<td>SalesAccountProfile.csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Account Profile</td>
<td>Account Web</td>
<td>Url.csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Contact Profile</td>
<td>Contact Additional Name</td>
<td>AdditionalName.csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Contact Profile</td>
<td>Contact Address</td>
<td>Address.csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Contact Profile</td>
<td>Contact Address Purpose</td>
<td>AddressPurpose.csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Contact Profile</td>
<td>Contact Classification</td>
<td>Classification.csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Contact Profile</td>
<td>Contact Relationship</td>
<td>ContactRelationship.csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Contact Profile</td>
<td>Contact e-mail</td>
<td>Email.csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Contact Profile</td>
<td>Contact Fax</td>
<td>Fax.csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Contact Profile</td>
<td>Contact Social Network</td>
<td>InstantMessenger.csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Contact Profile</td>
<td>Contact Mobile</td>
<td>Mobile.csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Contact Profile</td>
<td>Contact Phone</td>
<td>Phone.csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Contact Profile</td>
<td>Contact Sales Profile</td>
<td>SalesAccountProfile.csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Lead</td>
<td>Lead TC Member</td>
<td>Lead_ Contact.csv</td>
<td>LEAD_ID</td>
</tr>
<tr>
<td>Lead</td>
<td>Lead Item Assoc</td>
<td>Lead_ Product.csv</td>
<td>LEAD_ID</td>
</tr>
<tr>
<td>Lead</td>
<td>Lead Resource</td>
<td>Lead_ Resource.csv</td>
<td>LEAD_ID</td>
</tr>
<tr>
<td>Parent Object</td>
<td>Child Objects</td>
<td>Output File Name</td>
<td>Parent Key</td>
</tr>
<tr>
<td>-------------------</td>
<td>--------------------</td>
<td>--------------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>Lead</td>
<td>Lead Territory Assoc</td>
<td>Lead_Territories.csv</td>
<td>LEAD_ID</td>
</tr>
<tr>
<td>Opportunity</td>
<td>Opportunity Competitor</td>
<td>OPPORTUNITY_COMPETITOR.csv</td>
<td>OPTY_ID</td>
</tr>
<tr>
<td>Opportunity</td>
<td>Opportunity Contact</td>
<td>OPPORTUNITY_CONTACT.csv</td>
<td>OPTY_ID</td>
</tr>
<tr>
<td>Opportunity</td>
<td>Opportunity Lead</td>
<td>OPPORTUNITY_LEAD.csv</td>
<td>OPTY_ID</td>
</tr>
<tr>
<td>Opportunity</td>
<td>Opportunity Reference</td>
<td>OPPORTUNITY_REFERENCE.csv</td>
<td>OPTY_ID</td>
</tr>
<tr>
<td>Opportunity</td>
<td>Opportunity Resource</td>
<td>OPPORTUNITY_RESOURCE.csv</td>
<td>OPTY_ID</td>
</tr>
<tr>
<td>Opportunity</td>
<td>Opportunity Response</td>
<td>OPPORTUNITY_RESPONSE.csv</td>
<td>OPTY_ID</td>
</tr>
<tr>
<td>Opportunity</td>
<td>Revenue</td>
<td>OPPORTUNITY_REVENUE.csv</td>
<td>OPTY_ID</td>
</tr>
<tr>
<td>Opportunity</td>
<td>Revenue Partner</td>
<td>OPPORTUNITY_REVENUE_PARTNER.csv</td>
<td>OPTY_ID</td>
</tr>
<tr>
<td>Opportunity</td>
<td>Opportunity Source</td>
<td>OPPORTUNITY_SOURCE.csv</td>
<td>OPTY_ID</td>
</tr>
<tr>
<td>Product Group Details</td>
<td>Product Group Translations</td>
<td>ProductGroup.csv</td>
<td>PROD_GROUP_ID</td>
</tr>
<tr>
<td>Product Group Details</td>
<td>Product Group Translations</td>
<td>ProductGroup.csv</td>
<td>REVISION_NUM</td>
</tr>
<tr>
<td>Product Group Details</td>
<td>Product Group Products</td>
<td>ProductGroupItem.csv</td>
<td>PROD_GROUP_ID</td>
</tr>
<tr>
<td>Product Group Details</td>
<td>Product Group Products</td>
<td>ProductGroupItem.csv</td>
<td>REVISION_NUM</td>
</tr>
<tr>
<td>Product Group Details</td>
<td>Product Group Subgroups</td>
<td>ProductGroupRelationship.csv</td>
<td>PROD_GROUP_ID</td>
</tr>
<tr>
<td>Product Group Details</td>
<td>Product Group Subgroups</td>
<td>ProductGroupRelationship.csv</td>
<td>REVISION_NUM</td>
</tr>
<tr>
<td>Sales Resource Quota</td>
<td>Sales Resource Period Quota</td>
<td>Resource_Period_Quota.csv</td>
<td>RESOURCE_QUOTA_ID</td>
</tr>
<tr>
<td>Sales Territory</td>
<td>Sales Territory Coverage</td>
<td>TERR_COVERAGE.csv</td>
<td>TERRITORY_VERSION_ID</td>
</tr>
<tr>
<td>Sales Territory</td>
<td>Sales Territory Line Of Business</td>
<td>TERR_LOB.csv</td>
<td>TERRITORY_VERSION_ID</td>
</tr>
<tr>
<td>Sales Territory</td>
<td>Sales Territory Resource</td>
<td>TERR_RESOURCE.csv</td>
<td>TERRITORY_VERSION_ID</td>
</tr>
<tr>
<td>Parent Object</td>
<td>Child Objects</td>
<td>Output File Name</td>
<td>Parent Key</td>
</tr>
<tr>
<td>---------------</td>
<td>---------------</td>
<td>------------------</td>
<td>---------------------</td>
</tr>
<tr>
<td>Sales Territory Quota</td>
<td>Sales Resource Quota</td>
<td>Resource_ Quota.csv</td>
<td>TERRITORY_ QUOTA_ID</td>
</tr>
<tr>
<td>Sales Territory Quota</td>
<td>Sales Territory Period Quota</td>
<td>Territory_ Period_ Quota.csv</td>
<td>TERRITORY_ QUOTA_ID</td>
</tr>
<tr>
<td>Partner</td>
<td>Partner Address</td>
<td>Address.csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Partner</td>
<td>Partner Fax</td>
<td>Fax.csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Partner</td>
<td>Partner Hierarchy</td>
<td>Hierarchy.csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Partner</td>
<td>Partner Certification</td>
<td>PartnerCertification. csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Partner</td>
<td>Partner Expertise</td>
<td>PartnerExpertise. csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Partner</td>
<td>Partner Dimension Geography</td>
<td>PartnerGeography. csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Partner</td>
<td>Partner Dimension Industry</td>
<td>PartnerIndustry. csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Partner</td>
<td>Partner Interface</td>
<td>PartnerInterface. csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Partner</td>
<td>Partner Dimension Product</td>
<td>PartnerProductSpecialty. csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Partner</td>
<td>Partner Account Resource</td>
<td>PartnerTeam. csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Partner</td>
<td>Partner Dimension Sales Channel</td>
<td>PartnerType. csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Partner</td>
<td>Partner Primary Phone</td>
<td>PrimaryPhone. csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Partner</td>
<td>Partner Web</td>
<td>Url.csv</td>
<td>PARTY_ID</td>
</tr>
<tr>
<td>Program Enrollments</td>
<td>Program Enrollment Contracts</td>
<td>PartnerEnrollmentContract. csv</td>
<td>PROGRAM_ ENROLLMENT_ID</td>
</tr>
<tr>
<td>Program Enrollments</td>
<td>Program Enrollment Participants</td>
<td>PartnerEnrollmentParticipant. csv</td>
<td>PROGRAM_ ENROLLMENT_ID</td>
</tr>
</tbody>
</table>

Using File-Based Data Export to Export Data: Worked Example

This example demonstrates how to export Opportunity object from an instance of Oracle Sales Cloud using File-Based Data Export.
Exporting an Opportunity Object

To export an opportunity object, you must complete the following steps:

1. Navigate to Setup and Maintenance.
2. Search for the Manage File Export Activities task.
3. Click Go to Task.
4. On the Overview page, click Hierarchy Export. Click Create.
5. On the Create Export Process Definition page, enter name of activity, select Opportunity as parent object, and enter the file name.
7. Select the attributes that you want to export, using the Enabled checkbox.
8. You need to create and save a view criteria to start the export process. Create a view criteria, by clicking the Edit Filter Criteria button.
9. In the Edit Filter Criteria window, enter a name and a status for the view criteria. Click Save.
10. Click Next to go to the Create Export Process Definition page.
11. Schedule the export process either immediately or at a later date, by selecting the corresponding radio button.
12. Click Next to go to the review page, where you can either click Back to change any settings, Save and Close the process to run at a later time, or Activate the process.
13. Review the status of your export activity on the Overview page.
14. Click on the generated ZIP file on the exported data file column and save the file to your desktop.
4 Using Round-Trip Export-Import

Exporting and Importing Data Between Oracle Sales Cloud Instances Using Automatic Mapping: Worked Example

This example demonstrates how to export Opportunity object from one instance of Oracle Sales Cloud using File-Based Data Export. It also describes how to import the exported data into another instance of Oracle Sales Cloud using File-Based Data Import without manually mapping the source file and the application attributes.

Exporting Data from Oracle Sales Cloud Using File-Based Data Export

You can export data from Oracle Sales Cloud using File-Based Data Export, using the following steps:

1. Navigate to **Setup and Maintenance**.
2. Search for Manage File Export Activities task. Click **Go to Task**.
3. On the **Overview** page, click the **Create** icon.
4. On the **Create Export Process Definition** page, enter Name of activity, select Opportunity as Parent object, and enter the File Name.
5. Click **Next**. On the **Create Export Process Definition: Review** page. All attributes of the Opportunity object are displayed. Note that all child objects and their attributes are selected by default. Select the attributes that you want to export, using the **Enabled** checkbox.
6. You must create a View Criteria before you can export. Click **Edit Filter Criteria** button to bring up the Edit Filter Criteria dialog box.
7. You can use the View Criteria to filter the records based on certain criteria and save for future use. For instance, you can create a criterion to filter records that were updated after a certain date, or records that were created by a certain user. Add the required fields with conditions, and on the Saved Search drop down, click Personalize. On the **Personalize Saved Searches** dialog box, provide the name for view criteria in the Name field. Click **Apply** and **OK** to save.
8. Click **Next** to go to the **Create Export Process Definition** page.
9. Schedule the export process either immediately or at a later date, by selecting the corresponding radio button.
10. Click **Next** to go to the review page, where you can either click **Back** to change any settings, **Save and Close** the process to run at a later time, or **Activate** the process.
11. Review the status of your export activity on the **Overview** page.
12. Click on the generated ZIP file on the Exported data file column and save the file to your desktop.
Importing Data from Oracle Sales Cloud Using File-Based Data Import

You can import data from Oracle Sales Cloud using File-Based Data Import, using the following steps:

1. Navigate to Setup and Maintenance.
2. Search for Manage File Import Activities task. Click Go to Task.
3. On the Manage Import Activities page, click Create.
4. Select the FileType ‘ZIP File’. Provide a name for the Import Activity in the Name textbox.
5. Select the Object Opportunity from dropdown. If you are trying to perform round trip export-import of a custom object, make sure you have already generated the artifacts required for the importing and exporting of custom fields and objects on the Generate Import and Export Artifacts page.
6. Click Next to get the Map Fields page.

**Note:** You can’t map the Object Id value if you had exported the data from a different Oracle Sales Cloud environment.

7. Click Next and Activate the import process in the Review and Activate page.
8. Review the status of your import activity on the Overview page.

Related Topics

- Using the Round-Trip Feature with the Export-Import Activity: Explained
- Exporting and Importing an Import Mapping Between Oracle Sales Cloud Instances: Procedure

Exporting and Importing Data Between Oracle Sales Cloud Instances Using Automatic Mapping: Procedure

This procedure describes how to export data from one instance of Oracle Sales Cloud using File-Based Data Export. It also describes how to import the exported data into another instance of Oracle Sales Cloud using File-Based Data Import without manually mapping the source file and the application attributes.

Exporting Data from Oracle Sales Cloud Using File-Based Data Export

You can use the following steps to export data from an Oracle Sales Cloud instance using the automatic mapping feature:

1. From the Setup and Maintenance work area, search for the task Manage File Export Activities.
2. On the Overview page, click the Create icon.
3. On the Create Export Process Definition page, enter the name of the activity, select a Parent object, and enter the File Name.
4. Click Next. On the Create Export Process Definition: Review page all attributes of the selected object are displayed. All child objects and their attributes are selected by default. Select the attributes that you want to export, using the Enabled check box.
5. You must create a View Criteria before you can export. Click the **Edit Filter Criteria** button to display the Edit Filter Criteria dialog box.

6. You can use the View Criteria to filter the records based on certain criteria and save them for future use. For instance, you can create a criterion to filter records that were updated after a certain date, or records that were created by a certain user. Add the required fields with conditions, and on the **Saved Search** drop down list, click **Personalize**. On the **Personalize Saved Searches** dialog box, provide the name for the view criteria in the Name field. Click **Apply** and **OK** to save.

7. Click **Next** to go to the **Create Export Process Definition** page.

8. Schedule the export process either immediately or at a later date, by selecting the corresponding radio button.

9. Click **Next** to go to the review page, where you can either click **Back** to change any settings, **Save and Close** the process to run at a later time, or **Activate** the process.

10. Review the status of your export activity on the **Overview** page.

11. Click the generated ZIP file on the Exported data file column and save the file to your desktop.

---

### Importing Data from Oracle Sales Cloud Using File-Based Data Import

You can use the following steps to import data from an Oracle Sales Cloud instance using the automatic mapping feature.

1. From the **Setup and Maintenance** work area, search for the task Manage File Import Activities. Click **Go to Task**.

2. On the **Manage Import Activities** page, click **Create**.

3. Select the **FileType** ZIP File. Browse and select the file that you had generated as a result of export activity. Provide a name for the import activity in the **Name** text box.

   ✍️ **Note:** Don’t alter the name or contents of the ZIP file.

4. Select the **Map Automatically** check box to automatically map the columns in your source file to the application attributes. Otherwise, you must manually map the columns in the data file for both the parent and child attributes.

5. Select the object that you want to import from the drop-down list.

   If you’re trying to perform round trip export-import of a custom object, make sure you have already generated the artifacts required for the importing and exporting of custom fields and objects on the **Generate Import and Export Artifacts** page. The **Generate** button is enabled only if the sandbox is not active. Disable the sandbox to activate the **Generate** button.

6. Click **Next** to display the **Map Fields** page.

   ✍️ **Note:** You can’t map the Object Id value if you had exported the data from a different Oracle Sales Cloud environment. If one or more columns cannot be automatically mapped, then you must either map them manually or select Ignore. Otherwise, the application generates an error.

7. Click **Next** and **Activate** to activate the import process in the **Review and Activate** page.

8. Review the status of your import activity on the **Overview** page.

**Related Topics**

- Exporting and Importing an Import Mapping Between Oracle Sales Cloud Instances: Procedure
- Using the Round-Trip Feature with the Export-Import Activity: Explained
Importing Accounts

Importing Common Supporting Objects: Documentation

Overview

Common supporting objects often support the primary import objects. For example, additional party identifiers support accounts, contacts, and households. Notes support leads and opportunities. For a mapping of primary import objects to the common object import topics that support them, see:

Related Topics

- Importing Common Supporting Objects: Overview

Importing Accounts Using File-Based Import: Explained

This topic explains how to prepare and import account data from an external data source into Oracle Sales Cloud using the File-Based Data Import feature. An account is an entity with whom you have a selling relationship. Oracle Sales Cloud refers to sales profile and sales prospects collectively as an Account. An account business object allows you to capture all information about an account, such as the account profile, account contact relationships information, and account contact points.

A sales profile is a specific sell-to entity within a given account. You can create leads and opportunities for sales profiles. An entity with a sales profile can be one of the following:

- Sales Prospect: A prospective entity or person who does not have a sell-to address. This entity is used to define leads.
- Account: A sales profile or sales prospects.
- Legal Entity: A legal entity is a party that can enter into legal contracts or a business relationship. It can be sued if it fails to meet contractual obligations.

You can use the Account import object to import sales profiles or sales prospects of the type organization. You can use the Household import object to import accounts of the type household, and the Contact import object to import accounts of the type person. Consider the following questions when importing your data:

- Did you identify the records that should be imported as sales prospects, and the records that should be imported as sales profiles?
- How does your legacy or source system represent the account information compared to how Oracle Sales Cloud represents the same data?
- Do you have to configure values in Oracle Sales Cloud to map your existing data values to the Account import object?
- Do you have to customize Oracle Sales Cloud to capture attributes that are critical to the way you do business?
- What import features are available for importing your business object?
- How do I verify my imported data?
Comparing Business Object Structures

You must understand how your account data corresponds with the data in Oracle Sales Cloud so that you can map your legacy data to the data that Oracle Sales Cloud requires. First, you must understand how Oracle Sales Cloud represents the structure of the data for an account.

The Account import object structure is hierarchical. The root of the hierarchy is the organization profile, which must exist before you can import lower-level components, such as classifications, contacts, and sales profiles. These child entities can have other entities as their child entities. This hierarchical structure supports one-to-many relationships between the components that make up the account. The following figure shows the account object and its child entities.

The organization profile contains basic information about the account, such as the customer name and party usage. For each account, you can assign classifications, contacts, sell to addresses, phone details, and additional names. The contact of the account, in turn, includes other child entities that capture information about the contact, such as contact job, contact primary phone, contact primary address, and contact e-mail.

Note: All contact entities, such as primary phone or e-mail, include a child entity that captures the contact preference. For example, the Primary Phone Contact Preference entity captures the contact preference of the account for the contact method primary phone.

Comparing Business Object Data

Each import object is a collection of attributes that helps to map your data to the Oracle Sales Cloud data and to support one-to-many relationships between the structural components that make up the account.

You must understand the attribute details of the import objects so that you can prepare your import data. You can use reference guide files that contain attribute descriptions, values that populate attributes by default when you do not provide
values, and validation information for each attribute. The validation information includes the navigation path to the task where you can define values in Oracle Sales Cloud. For example, if you have values in your data that correlate to a choice list in Oracle Sales Cloud, then the validation information for that attribute provides the task name in the Setup and Maintenance work area where you can define your values. For additional information, including a list of reference guide file names and locations that you need to complete this task, see the topic listed in the following table.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Related Import Object Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account</td>
<td>Account Import Objects : How They Work Together</td>
</tr>
</tbody>
</table>

Note: You can use the keyword importing accounts to search for related topics in Oracle Sales Cloud Help.

Extensible Attributes
If you want to extend the Oracle Sales Cloud object to import your legacy or source data, then you must use Application Composer to design your object model extensions and to generate the required artifacts to register your extensions and make them available for importing. The corresponding import object is updated with the extensible attributes, which you can then map to your source file data. You can use the same source file to import both extensible custom attributes and the standard import object attributes.

Importing Accounts Using File-Based Data Import
For the account business object, you must use the File-Based Data Import feature. You prepare XML or text source data files in a form that is suitable for a file-based import. The file-based import process reads the data in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables.

The Define File-Based Data Import Setup and Maintenance task list includes the tasks that are required to configure the import objects, to create source-file mappings, and to schedule the import activities. You submit file-based import activities for each import object. When you’re creating a new account, you import the Account object. You must be assigned the Customer Relationship Management Application Administrator job role to access and submit the import activities for accounts.

When importing account information, you first import the basic account profile information, followed by the child entities for the account. When importing child entities, you must provide the parent reference information for all parent levels for the entity. For example, you first import basic profile details, such as customer name, party type, and party usage. You then import contacts and contact information, such as phone, address, contact points, and fax for the account. You must provide the PartyOrigSystem and PartyOrigSystemReference of the account when importing contacts for the account. PartyOrigSystem is the source system code that identifies the source of the information being imported. PartyOrigSystemReference is unique for each row of data within a single import, and is a combination of PartyOrigSystem and a unique reference. When importing contact information for an account, you must provide the relationship reference information. This information is required because a contact can have multiple relationships with an account.

Verifying Your Imported Data
Oracle Sales Cloud Applications provide File-Based Import activity reports, which you can use to verify imported data. Alternatively, you can also navigate to the Customer Center work area to view the account information that you have imported.

Related Topics
- File-Based Data Import: How It Works
Account Import Objects: How They Work Together

The Account import object allows you to import accounts, their detailed information, and contacts related to the account. This topic describes the Account import object. It introduces the following:

- Target objects for the Account import objects
- Target import object attribute
- Target import object attribute reference guide files for evaluating and mapping source file data

Account Target Import Object Concepts

You use the Account import object to import accounts and contacts related to accounts. The Account import object is split into separate target import objects for organizing the individual attributes of the account and account contact.

The target import objects in the Account import object are generally grouped into information about the account and information about the account contact. The organization profile is the target import object containing attributes to import information about the account. You can have multiple contacts associated with an account.

When updating an existing account with additional information, you must provide the parent reference information for the existing account. When importing account contacts or contact information for an account, you must provide relationship reference information in addition to the parent reference. This information is required because a contact can have multiple relationships with an organization, such as an employee or board member. When importing information about a contact you must refer to the specific relationship that you want to import information for. For example, you might want to import information for John Smith the employee or John Smith the board member. If you do not include the reference information for an account-contact relationship, then the import process creates a relationship.

To update the information for an existing account or to create an account record, you can import account profile information, addresses, and contact points, such as a phone and fax. The following target import objects are for creating and updating account information: OrganizationProfile, Fax, Address, PrimaryPhone, Url, SalesAccountProfile, Classification, AdditionalName, Relationship, SellerToAddress, and AdditionalIdentifier.

To update or to create an account contact, use the following target import objects: ContactPersonProfile, ContactJob, ContactPrimaryAddress, ContactEmail, ContactPrimaryPhone, ContactMobile, ContactInstantMessenger, and ContactFax. All contact-related entities, such as primary phone or e-mail, include a child entity that captures the contact preference. For example, the Primary Phone Contact Preference entity captures the contact preference of the legal entity for the contact method primary phone.

Target Import Objects Attributes

You must compare the attributes that you want to import with the target object attributes that are available and with their valid values. To evaluate your source data and Oracle Sales Cloud attributes for mapping and validation, you use a reference file. See the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. A reference guide file includes attribute
descriptions, default values, and validations performed by the import process. Review the validation for each attribute to determine whether there are functional prerequisites or prerequisite setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can predefine and manage import mappings using the Manage File Import Mappings task, or you can define the mapping when you define the import activity using the Manage File Import Activities task. Both tasks are available in the Setup and Maintenance work area.

⚠️ **Note:** If any of the attributes you want to import do not have an equivalent target object attribute, then review the Application Composer Extensibility features for the account.

### Target Import Objects Attributes Resources

To access reference files for the object’s target import objects, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

The following table lists the reference files that are available by target import object:

<table>
<thead>
<tr>
<th>Target Import Object</th>
<th>Description</th>
<th>Reference Guide File Names</th>
</tr>
</thead>
<tbody>
<tr>
<td>OrganizationProfile</td>
<td>Includes detailed account information, such as, organization name and organization type.</td>
<td>HZ_IMP_PARTIES_T_Reference</td>
</tr>
<tr>
<td>PrimaryPhone</td>
<td>Account’s primary phone number. If the account has multiple phone numbers, one of the phone numbers is designated as the primary phone number.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>Address</td>
<td>Account’s address.</td>
<td>HZ_IMP_LOCATIONS_T_Reference</td>
</tr>
<tr>
<td>Classification</td>
<td>Account’s classification.</td>
<td>HZ_IMP_CLASSIFICS_T_Reference</td>
</tr>
<tr>
<td>AdditionalName</td>
<td>Account’s additional or alternative name.</td>
<td>HZ_IMP_ADDTNLPARTYNAMES_T_Reference</td>
</tr>
<tr>
<td>Fax</td>
<td>Account’s fax number.</td>
<td>HZ_IMPCONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>FaxContactPreference</td>
<td>Indicates the account preference about being contacted through Fax.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>Url</td>
<td>Account’s URL.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>SalesProfile</td>
<td>Account’s sales account information. A sales account is a specific sell-to entity within a given account, an account can have multiple sales accounts and sales profiles.</td>
<td>ZCA_IMP_SALES_ACCOUNTS_Reference</td>
</tr>
<tr>
<td>SellToAddress</td>
<td>Account’s addresses and party sites information. If party site usage of an address</td>
<td>HZ_IMP_PARTYSITES_T_Reference</td>
</tr>
</tbody>
</table>
## Importing Accounts

<table>
<thead>
<tr>
<th>Target Import Object</th>
<th>Description</th>
<th>Reference Guide File Names</th>
</tr>
</thead>
<tbody>
<tr>
<td>AdditionalIdentifier</td>
<td>Account’s additional identifier information.</td>
<td>HZ_IMP_ADDTNLPARTYIDS_T_Reference</td>
</tr>
<tr>
<td>Relationship</td>
<td>Account’s relationship information. You must enter a relationship code in the RelationshipCode column, when creating a relationship.</td>
<td>HZ_IMP_RELSHIPS_T_Reference</td>
</tr>
<tr>
<td>ContactPersonProfile</td>
<td>Account contact’s person information.</td>
<td>HZ_IMP_RELSHIPS_T_Reference</td>
</tr>
<tr>
<td>ContactAdditionalName</td>
<td>Account contact’s additional name.</td>
<td>HZ_IMP_ADDTNLPARTYNAMES_T_Reference</td>
</tr>
<tr>
<td>ContactEmail</td>
<td>Account contact’s email.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>ContactInstantMessenger</td>
<td>Account contact’s instant messenger or social networking information.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>ContactPrimaryAddress</td>
<td>Account contact’s primary address. If the contact has multiple addresses, one of the addresses is designated as the primary address.</td>
<td>HZ_IMP_PARTYSITES_T_Reference</td>
</tr>
<tr>
<td>ContactFax</td>
<td>Account contact’s fax.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>ContactJob</td>
<td>Account contact’s job information.</td>
<td>HZ_IMP_CONTACTS_T_Reference</td>
</tr>
<tr>
<td>ContactMobile</td>
<td>Account contact’s mobile number.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>ContactPrimaryPhone</td>
<td>Account contact’s primary phone number. If the account has multiple phone numbers, one of the phone numbers is designated as the primary phone number.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>PrimaryPhoneContactPreference</td>
<td>Indicates the account contact preference about being contacted through phone.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>ContactEmailContactPreference</td>
<td>Indicates the account contact preference about being contacted through e-mail.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>ContactFaxContactPreference</td>
<td>Indicates the account contact preference about being contacted through Fax.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>ContactMobileContactPreference</td>
<td>Indicates the account contact preference about being contacted through mobile phone.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
</tbody>
</table>
## Importing Sales Profiles Using File-Based Import: Quick Start

This topic describes how to get you started with importing sales profiles. It includes the following information:

- Identifying and associating records
- The minimum data required and the prerequisites steps
- How to access and use reference files to evaluate attributes
- Additional tips

A sales profile enables you to capture information about a selling relationship with an account or consumer. When a party has one sell-to address, it ceases to be a sales prospect and becomes a sales profile. You can import sales profiles or update existing sales profiles, using the File-Based Data Import feature. The sales profile includes attributes that can be used for assigning territories and providing account ownership details. You must have an organization or person with a sell-to address defined before you can import sales profiles.

The following table lists the file-based import objects and target objects that you can use to import sales profiles.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Party Type</th>
<th>Target Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contact</td>
<td>Person</td>
<td>PersonProfile, SalesProfile</td>
</tr>
</tbody>
</table>

### Target Import Object

<table>
<thead>
<tr>
<th>Description</th>
<th>Reference Guide File Names</th>
</tr>
</thead>
<tbody>
<tr>
<td>Indicates the account contact preference about being contacted at the primary address.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>Indicates the account contact preference about being contacted through the primary phone number.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
</tbody>
</table>

### Related Topics

- File-Based Data Import: How It Works
- Getting Started with File-Based Import: Documentation Overview
- Importing Accounts Using File-Based Import: Explained
- Extending Oracle Sales Cloud: How It Works
- File Based Data Import for Oracle Sales Cloud

**Note:** Oracle Sales Cloud supports the ContactPersonProfile target object. However, the object is now deprecated. For updating any contact information, use the PersonProfile target object instead.
Identifying and Associating Records with Each Other

To add a sales profile to an account or to update an existing sales profile using file-based import, your source file must contain information about the customer organization or consumer person party to which the sales profile belongs. You map your source file data to the organization or person profile target object and to the sales profile target object.

If you’re creating the party and the sales profile in the same import activity, then group the party and sales profile data together in your source file. The import process can import the party and then identify the party record so that it can import the associated sales profile.

To add sales profile to an existing party or to update an existing sales profile record, your source file must include the values that enable the import process to identify the existing records. These values are source system and source system reference value combination, Oracle Sales Cloud internal ID, or public unique identifiers, such as business keys or external IDs.

If the source of your data is an external application, and if you intend to import updates to previously imported records from the external application, then you can provide the source system code and the unique reference value for the source system’s record in your source file. The file-based data import process stores a cross-reference between the source system information and Oracle Sales Cloud internal ID. The import process uses the combination of source system and source system reference value to identify the existing record.

You can use external IDs or business keys to identify and associate records with each other when you’re importing a smaller set of records. Business keys are a set of attributes that uniquely identify a party record. For example, a person’s last name and first name uniquely identify a person. An external ID is a unique record identifier from an application outside of Oracle Sales Cloud. You can use the following business keys or external IDs in Oracle Sales Cloud:

- Organization name for organizations
- First name and last name for persons
- Contact information, which is a combination of e-mail Id, mobile number, IM, and URL.
- Address information, which is a combination of address1, address2, city, and postal code.

You can configure source systems to identify the source of the data that you’re importing. Source systems are external sources of data that are used to import data into Oracle Sales Cloud. Oracle Sales Cloud uses source system references to create references between source IDs and the Oracle Sales Cloud database IDs. You can configure source systems either by using the Manage Trading Community Source System task, or by importing the source system information into Oracle Sales Cloud using the Source System Reference import object.

When the source of your data isn’t an external application and you don’t intend to regularly update the data, you don’t need the source system information. To import updates to your existing data, you can export the Oracle Sales Cloud internal ID and add it to your source file. The import process uses the internal ID to identify the existing record.

Minimum Data Required and the Prerequisite Steps

The minimum data that is required to import sales profile information depends on the following:

- The purpose of the import. The data requirements are different when you’re creating both the party and the sales profile in the same import batch, adding new sales profile to an existing party, or updating a sales profile record.
• Identifying and associating records. In some cases, you can select which attributes you want to provide to the import process when identifying and associating records.

The values that you provide in your source file may require a setup task or manual step to ensure that the information passes the import validation. Before you prepare your source file, complete the prerequisite steps to determine the data values and then include those values in your source file.

The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create or update a sales profile record. You can optionally include attributes that are available for import in your import file but that aren’t listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Sales Profile Records in the Same Batch</th>
<th>Creating Sales Profile Record for an Existing Party</th>
<th>Updating Sales Profile Record</th>
</tr>
</thead>
<tbody>
<tr>
<td>PartyOrigSystem</td>
<td>The code representing the source system for the party (account or contact) to which the sales profile belongs.</td>
<td>Identify the source system code that was used when you imported the party in a prior batch, or identify the source system code that you use when importing the party in the same batch as this sales profile.</td>
<td>Conditionally required</td>
<td></td>
<td>Conditionally required</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Provide the source system and source system reference values if the source of your data is a third party or external application and you intend to import updates to previously imported records from the same system.</td>
<td></td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyId (Oracle Sales Cloud record ID).</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyNumber (public unique identifier for an organization profile).</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• A business key such as organization name.</td>
<td></td>
</tr>
<tr>
<td>PartyOrigSystemReference</td>
<td>The reference number or text representing the source system unique ID for the party.</td>
<td>Identify the reference value from your source system that was used when you imported the party in a prior batch, or identify the reference value from your source system that was used when importing the party in the same batch as this sales profile.</td>
<td>Conditionally required</td>
<td>Provide the source system and source system reference</td>
<td>Conditionally required</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Party and Sales Profile Records in the Same Batch</td>
<td>Creating Sales Profile Record for an Existing Party</td>
<td>Updating Sales Profile Record</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
<td>-------------------------</td>
<td>------------------------------------------------------------</td>
<td>-----------------------------------------------</td>
<td>-------------------------------</td>
</tr>
<tr>
<td>party (account or contact) to which the sales profile belongs.</td>
<td>party in a prior batch, or identify the source system reference you use when importing the party in the same batch as this sales profile</td>
<td>system reference values if the source of your data is a third party or external application and you intend to import updates to previously imported records from the same application.</td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
<td></td>
</tr>
<tr>
<td>PartyId</td>
<td>The Oracle Sales Cloud record ID for the party (account or contact) to which the sales profile belongs.</td>
<td>Identify the party ID value by exporting the Party object using Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>Conditionally required: Provide reference information to identify the existing party. The reference information can be:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Conditionally required: Provide reference information to identify the existing party. The reference information can be:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyId (Oracle Sales Cloud record ID).</td>
<td>• PartyId (Oracle Sales Cloud record ID).</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyOrigSystem and PartyOrigSystemRef (source system code and source system reference values) provided when importing the organization profile.</td>
<td>• PartyOrigSystem and PartyOrigSystemRef (source system code and source system reference values) provided when importing the organization profile.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyNumber (public unique identifier for an organization profile).</td>
<td>• PartyNumber (public unique identifier for an organization profile).</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• A business key such as organization name.</td>
<td>• A business key such as organization name.</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Party and Sales Profile Records in the Same Batch</td>
<td>Creating Sales Profile Record for an Existing Party</td>
<td>Updating Sales Profile Record</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------------</td>
<td>------------------------------------------------------------</td>
<td>-----------------------------------------------------</td>
<td>--------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>By default, the user importing the account is designated as the account owner. To specify a different account owner, include one of the reference information for the account owner in the import file:</td>
<td>By default, the user importing the account is designated as the account owner. To specify a different account owner, include one of the reference information for the account owner in the import file:</td>
<td>By default, the user importing the account is designated as the account owner. To specify a different account owner, include one of the reference information for the account owner in the import file:</td>
</tr>
<tr>
<td>AccountDirectorId</td>
<td>The unique party ID of the resource assigned to manage the sales profile.</td>
<td>Identify the party ID value of the account owner by exporting the Party object using Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Conditionally required.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>By default, the user importing the account is designated as the account owner. To specify a different account owner, include one of the reference information for the account owner in the import file:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Party ID of the account owner in the AccountDirectorId column.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Origin system and origin system reference of the account owner in the OWNER_ORIG_S AND OWNER_ORIG_S columns.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Party Number of the account owner in the OWNER_PARTY_ column.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
During import, the attribute Type is set to Account for all new accounts.

The following figure shows a sample attribute mapping to import a new sales profile for an existing party. In this scenario, the origin system and the origin system reference information are used to identify the existing party and the new sales profile.

Using Reference Files to Evaluate Attributes

For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

Additional Tips

Tip: You can explore related topics by entering the following keywords in the Oracle Sales Cloud Help Application search.

- Importing contacts
- Importing accounts

Tip: If your data isn’t from a source system, then you can emulate a system by defining a source system and manually generating reference values for your data. For example, if you periodically gather information at sales events or through sales calls, then you might define a source system, using a code of SALES_ACTIVITY. The format of your reference values might concatenate the date, salesperson name, and sequence number, such as 01_Jan_2014_Jackson_01.
Importing Contacts

Importing Contacts and Consumers: Important Considerations

The Consumer object is only applicable if you are using release 1.0.7 or an earlier release. All capabilities of the Consumer object are included in the Contact object for releases 1.0.8 and later. If you are upgrading from an earlier release to release 1.0.8 or later, it is recommended that you use the Contact object to update your consumer information using file-based data import in the current release.

Importing Common Supporting Objects: Documentation Overview

Common supporting objects often support the primary import objects. For example, additional party identifiers support accounts, contacts, and households. Notes support leads and opportunities. For a mapping of primary import objects to the common object import topics that support them, see:

Related Topics
• Importing Common Supporting Objects: Overview

Importing Contacts Using File-Based Import: Explained

This topic explains how to prepare and import contact data from an external data source into Oracle Sales Cloud using the File-Based Data Import feature. A contact is an individual who is a customer or a prospect, or a contact for an existing customer or consumer, or a contact that does not yet have an established business association with a customer or consumer. Thus, a contact could be an employee of a customer organization, a person you may have met who could help with your business, or a prospective or current individual customer. The contact object contains information that identifies the contact and offers the contact points of the contact. Contact points can be geographical addresses, phone numbers, e-mail IDs, URLs, messenger IDs, and so on. The contact object also contains contact preference information for the contact. You must create or import contacts before you can associate them with account objects.

Consider the following questions when importing your data:

• How does your legacy system or source system represent the contact compared to how Oracle Sales Cloud represents the same data?
• Do you have to configure values in Oracle Sales Cloud to map to your data values?
• Do you have to customize Oracle Sales Cloud to capture additional attributes that are critical to the way you do business?
• What import features are available for importing your business object?
• How do you verify your imported data?
Comparing Business Object Structures
You must understand how your contact data corresponds with the data in Oracle Sales Cloud so that you can map your legacy data to the data that Oracle Sales Cloud requires. First, you must understand how Oracle Sales Cloud represents the structure of the data for a contact.

In Oracle Sales Cloud, one table stores the contact information, and another table optionally stores contact point details for that contact. A contact point is an identified means to contact a party. Contact points can be phone numbers, e-mail IDs, Web site addresses, or instant messenger IDs.

The contact profile contains basic information about the contact, such as the contact name, party type, and party usage. For each contact, you can assign classifications, phone details, and additional names. If the contact is a consumer, then the contact relationship of the consumer includes other child entities that capture information about the consumer contact, such as contact job, contact phone, contact address, and contact e-mail.

Note: All contact-related entities, such phone or e-mail, include a child entity that captures the contact preference. For example, the Phone Contact Preference entity captures the contact preference of the consumer for the contact method phone.

To facilitate importing contacts, Oracle Sales Cloud incorporates the structure of the contact business object into import objects. The import object for contacts is Contact.

Comparing Business Object Data
Each import object is a collection of attributes that helps to map your data to the Oracle Sales Cloud data and to support one-to-many relationships between the structural components that make up the contact.

You must understand the attribute details of the import objects so that you can prepare your import data. You can use reference guide files that contain attribute descriptions, values that populate attributes by default when you do not provide values, and validation information for each attribute. The validation information includes the navigation path to the task where you can define values in Oracle Sales Cloud. For example, if you have values in your data that correlate to a choice list in Oracle Sales Cloud, then the validation information for that attribute provides the task name in the Setup and Maintenance work area where you can define your values. For additional information, including a list of reference guide file names and locations that you need to complete this task, see the topic listed in the following table.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Related Import Object Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contact</td>
<td>Contact Import Objects: How They Work Together</td>
</tr>
</tbody>
</table>

Note: You can use the keyword importing contacts to search for related topics in Oracle Sales Cloud help.

Extensible Attributes
If you want to extend the Oracle Sales Cloud object to import your legacy or source data, then you must use Application Composer to design your object model extensions and to generate the required artifacts to register your extensions and make them available for importing. The corresponding import object is updated with the extensible attributes, which you can then map to your source file data. You can use the same source file to import both extensible custom attributes and the standard import object attributes.
Importing Contacts Using File-Based Data Import

For the contact business object, you must use the File-Based Data Import feature. You prepare XML or text source data files in a form that is suitable for a file-based import. The file-based import process reads the data in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables.

The Define File-Based Data Import Setup and Maintenance task list includes the tasks that are required to configure the import objects, to create source-file mappings, and to schedule the import activities. You submit file-based import activities for each import object. When you’re creating a new contact, you import the Contact object. You must be assigned the Master Data Management Administrator job role to access and submit the import activities for contacts.

Verifying Your Imported Data

Oracle Sales Cloud Applications provide File-Based Import activity reports, which you can use to verify imported data. You can also access the Customer Center or Partner Center work area to view the contact information that you have imported.

Related Topics

- File-Based Data Import: How It Works
- Getting Started with File-Based Import: Documentation Overview
- Contact Import Objects: How They Work Together
- Extending Oracle Sales Cloud: How It Works

Contact Import Objects: How They Work Together

You can import all your contact information using the Contact import object. This topic describes this import object and introduces the following:

- Target objects for the Contact import object
- Target import object attributes
- Reference guide files for target import object attributes

Contact Target Import Objects

You use the contact import object to import contact and information. The contact import object is split into separate target import objects for organizing the individual attributes of the contact and the contact’s relationship with other parties.

The target import objects included in the Contact import are grouped into information about the contact and the contact’s relationship with other parties. The contact profile is the target import object containing attributes to import information about the contact. You can have multiple contact relationships associated with a contact. There are multiple target import objects that include attributes to import contacts and their related information.

When updating an existing contact, you must provide the parent reference information of the existing contact. When importing contact profile or contact point information for a contact, you must provide relationship reference information in addition to the parent reference. You must provide this information because a contact can have multiple relationships with an organization, such as employee or board member. When importing information about a contact, you must refer to the specific relationship that you want to import information for. For example, you must specify whether you want to import information...
for John Smith the employee or John Smith the board member. If you do not include the reference information for a contact relationship, then the import process creates a new relationship.

To update the information for an existing contact or to create a contact record, you can import contact profile information, addresses, and contact points. The following target import objects are for creating and updating contact and contact preference information: PersonProfile, Fax, PrimaryPhone, Url, SalesProfile, Classification, AdditionalName, Relationship, and AdditionalIdentifier.

To update or create a contact point for a contact relationship, use the following target import objects: ContactRelationship, ContactJob, ContactAddress, ContactEmail, ContactPhone, ContactMobile, ContactInstantMessenger, and ContactFax. All contact-point-related entities, such as phone or e-mail, include a child entity that captures the contact preference. For example, the ContactPhoneContactPreference entity captures the contact preference of the primary phone contact method.

Target Import Object Attributes

You must compare the attributes that you want to import with the target object attributes that are available and with their valid values. To evaluate your source data and Oracle Sales Cloud attributes for mapping and validation, you use a reference file. See the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. A reference guide file includes attribute descriptions, default values, and validations performed by the import process. Review the validation for each attribute to determine whether there are functional prerequisites or prerequisite setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can predefine and manage import mappings using the Manage File Import Mappings task, or you can define the mapping when you define the import activity using the Manage File Import Activities task. Both tasks are available in the Setup and Maintenance work area.

Note: If any of the attributes that you want to import do not have an equivalent for the target object attribute, then review the Application Composer extensibility features for the contact object.

Reference Files for Target Import Object Attributes

To access reference files for this object’s target import objects, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

The following table lists the reference files that are available by target import object.

<table>
<thead>
<tr>
<th>Target Import Object</th>
<th>Description</th>
<th>Reference Guide File Names</th>
</tr>
</thead>
<tbody>
<tr>
<td>PersonProfile</td>
<td>Includes information that identifies a contact, such as first name, last name, date of birth, and so on.</td>
<td>HZ_IMP_PARTIES_T_Reference</td>
</tr>
<tr>
<td>SalesProfile</td>
<td>Includes detailed information about a sales profile. A sales profile is a specific sell-to entity within a given account. An account or contact can have multiple sales profiles.</td>
<td>ZCA_IMP_Sales_Accounts_Reference</td>
</tr>
<tr>
<td>Fax</td>
<td>Contact’s fax number.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>Target Import Object</td>
<td>Description</td>
<td>Reference Guide File Names</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------------------------------------------</td>
</tr>
<tr>
<td>FaxContactPreference</td>
<td>Contact’s preferences about being contacted through Fax.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>Phone</td>
<td>Contact’s phone number. If the contact has multiple phone numbers, then the first phone number is designated as the primary phone number. The attributes of PrimaryPhone, Fax, and URL are the same. However, the value of the attribute ContactPointType is different for each of these contact point related entities.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>PhoneContactPreference</td>
<td>Contact’s preferences about being contacted by phone.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>Email</td>
<td>Contact’s e-mail.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>EmailContactPreference</td>
<td>Contact’s preferences about being contacted through e-mail.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>InstantMessenger</td>
<td>Contact’s instant messenger information.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>Mobile</td>
<td>Contact’s mobile number.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>MobileContactPreference</td>
<td>Contact’s preferences about being contacted through mobile phone.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>Classification</td>
<td>Contact’s classification information. Classification allows you to categorize entities, such as parties, projects, tasks, and orders as hierarchies.</td>
<td>HZ_IMP_CLASSIFICS_T_Reference</td>
</tr>
<tr>
<td>Address</td>
<td>Contact’s address. If the contact has multiple addresses, then the first address is designated as the primary address.</td>
<td>HZ_IMP_PARTYSITES_T_Reference</td>
</tr>
<tr>
<td>AddressContactPreference</td>
<td>Contact’s preferences about being contacted at the primary address.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>AdditionalName</td>
<td>Contact’s alternative name.</td>
<td>HZ_IMP_ADDTNLPARTYNAMES_T_Reference</td>
</tr>
<tr>
<td>AdditionalIdentifier</td>
<td>Includes the basic information about an additional identifier for the contact.</td>
<td>HZ_IMP_ADDTNLPARTYIDS_T_Reference</td>
</tr>
<tr>
<td>ContactRelationship</td>
<td>Includes information about a relationship between the contact and other parties.</td>
<td>HZ_IMP_RELSHIPS_T_Reference</td>
</tr>
<tr>
<td>ContactEmail</td>
<td>Contact’s e-mail contact point.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
</tbody>
</table>
### Target Import Object

<table>
<thead>
<tr>
<th>Target Import Object</th>
<th>Description</th>
<th>Reference Guide File Names</th>
</tr>
</thead>
<tbody>
<tr>
<td>ContactEmailContactPreference</td>
<td>Contact’s preferences about being contacted through the e-mail contact point.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>ContactFax</td>
<td>Indicates the number of the fax contact point of the contact.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>ContactFaxContactPreference</td>
<td>Contact’s preference about being contacted through the fax contact point.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>ContactInstantMessenger</td>
<td>Contact’s instant messenger contact point.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>ContactMobile</td>
<td>Contact’s mobile contact point.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>ContactMobileContactPreference</td>
<td>Contact’s preferences about being contacted through the mobile phone contact point.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>ContactPhone</td>
<td>Contact’s phone contact point. If the contact has multiple phone numbers, then the first phone number is designated as the primary phone number.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>ContactPhoneContactPreference</td>
<td>Contact’s preferences about being contacted through the phone contact point.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>ContactAddress</td>
<td>Contact’s address contact point. If the contact has multiple addresses, then the first address is designated as the primary address.</td>
<td>HZ_IMP_PARTYSITES_T_Reference</td>
</tr>
<tr>
<td>ContactAddressContactPreference</td>
<td>Contact’s preferences about being contacted at the address contact point.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>ContactJob</td>
<td>Contacts’ job information.</td>
<td>HZ_IMP_CONTACTS_T_Reference</td>
</tr>
</tbody>
</table>

### Related Topics

- **File-Based Data Import: How It Works**
- **Getting Started with File-Based Import: Documentation Overview**
- **Importing Contacts Using File-Based Import: Worked Example**
- **Extending Oracle Sales Cloud: How It Works**
Importing Person Profiles Using File-Based Import: Quick Start

This topic describes how to get you started with importing person profiles. It includes the following information:

- An example of how to identify and associate records
- The minimum data required and the prerequisites steps
- How to access and use reference files to evaluate attributes
- Additional tips

A person party profile enables you to import contacts, and household members. For example, you can use person profiles to import contact information. A person profile contains various attributes, such as first name, last name, and so on. You can import new person profiles or update existing person profile records using the File-Based Data Import feature. The following table lists the file-based import objects and target objects that you can use to import person profiles.

The following table lists the file-based import objects and target objects that you can use to import person profiles.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Target Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contact</td>
<td>PersonProfile</td>
</tr>
<tr>
<td>Account (Account Contact)</td>
<td>PersonProfile, ContactPersonProfile (See note below.)</td>
</tr>
<tr>
<td>Household (Customer Contact)</td>
<td>PersonProfile, ContactPersonProfile (See note below.)</td>
</tr>
<tr>
<td>Legal Entity (Legal Entity Contact)</td>
<td>PersonProfile, ContactPersonProfile (See note below.)</td>
</tr>
<tr>
<td>Employee Resource</td>
<td>PersonProfile</td>
</tr>
<tr>
<td>Lead (Lead Contact)</td>
<td>PersonProfile, ContactPersonProfile (See note below.)</td>
</tr>
<tr>
<td>Partner Contact</td>
<td>PersonProfile</td>
</tr>
<tr>
<td>Marketing Campaign Response (Contact Respondent)</td>
<td>PersonProfile</td>
</tr>
<tr>
<td>Marketing Campaign Response (Marketing Campaign Response Contact)</td>
<td>PersonProfile, ContactPersonProfile (See note below.)</td>
</tr>
</tbody>
</table>

**Note:** Oracle Sales Cloud supports the ContactPersonProfile target object. However, the object is now deprecated. For updating any contact information, use the PersonProfile target object instead.
Identifying and Associating Records with Each Other

To add person profiles or to update existing person profiles using file-based import, your source file must contain information about the persons.

To update an existing person profile, your source file must include the values that enable the import process to identify the existing records. These values will be source system and source system reference value combination, Oracle Sales Cloud internal ID, or public unique identifiers, such as business keys or external IDs.

If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system's record in your source file. The file-based data import process stores a cross-reference between the source system information and Oracle Sales Cloud internal ID. The import process uses the combination of source system and source system reference value to identify the existing record.

You can use external IDs or business keys to identify and associate records with each other when you’re importing a smaller set of records. Business keys are a set of attributes that uniquely identify a party record. For example, a person’s last name and first name uniquely identify a person. An external ID is a unique record identifier from a system outside of Oracle Sales Cloud. You can use the following business keys or external IDs in Oracle Sales Cloud:

- First name and last name for persons
- Contact information, which is a combination of email Id, mobile number, IM, and URL.
- Address information, which is a combination of address1, address2, city, and postal code.

You can configure source systems to identify the source of the data that you’re importing. Source systems are external sources of data that are used to import data into Oracle Sales Cloud. Oracle Sales Cloud uses source system references to create references between source IDs and the Oracle Sales Cloud database IDs. You can configure source systems either by using the Manage Trading Community Source System task, or by importing the source system information into Oracle Sales Cloud using the Source System Reference import object.

When the source of your data isn’t an external system and you don’t intend to regularly update the data, you don’t need the source system information. To import updates to your existing data, you can export the Oracle Sales Cloud internal ID and add it to your source file. The import process uses the internal ID to identify the existing record.

Minimum Data Required and the Prerequisite Steps

The minimum data that is required to import person profile information depends on the following:

- The purpose of the import. The data requirements are different when you’re creating the person profile or updating an existing person profile record.
- Identifying and associating records. In some cases, you can choose which attributes you want to provide to the import process when identifying and associating records.

The values that you provide in your source file may require a setup task or manual step to ensure that the information passes the import validation. Before you prepare your source file, complete the prerequisite steps to determine the data values and then include those values in your source file.

The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create or update a person profile record. You can optionally include attributes that are available for import in your import file but that aren’t listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Create Party and Person Profiles</th>
<th>Update Person Profile Records</th>
</tr>
</thead>
</table>
| ObjectKey or PartyId | The Oracle Sales Cloud record ID for the party (account, contact, household, or legal entity) to which the person profile belongs. | Identify the party ID value by exporting the Party object using the **Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes** task. | The attribute is neither required nor conditionally required. | Conditionally required. Provide reference information to identify the existing party. The reference information can be:  
  - PartyId (Oracle Sales Cloud record ID).  
  - PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.  
  - Party Number (public unique identifier for an organization profile).  
  - A business key such as person name. |
| PartyOrigSystem      | The code representing the external source system for the party (account, contact, household, or legal entity) to which the person profile belongs. | Predefine your source system code as enabled for parties, using the **Setup and Maintenance, Manage Trading Community Source Systems** task. | Conditionally required. Provide reference information to identify the existing party. The reference information can be:  
  - PartyId (Oracle Sales Cloud record ID).  
  - PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.  
  - Party Number (public unique identifier for an organization profile).  
  - A business key such as person name. | Conditionally required. Provide reference information to identify the existing party. The reference information can be:  
  - PartyId (Oracle Sales Cloud record ID).  
  - PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.  
  - Party Number (public unique identifier for an organization profile). |
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Create Party and Person Profiles</th>
<th>Update Person Profile Records</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>organization name.</td>
<td></td>
</tr>
<tr>
<td>PartyOrigSystemReference</td>
<td>The reference number or text representing the source system unique ID for the party (account, contact, household, or legal entity) to which the person profile belongs.</td>
<td>Identify the reference value from your source system.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• PartyId (Oracle Sales Cloud record ID).</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Party Number (public unique identifier for an organization profile).</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• A business key such as person name.</td>
<td></td>
</tr>
<tr>
<td>PersonFirstName</td>
<td>First name of a person party.</td>
<td>No prerequisite tasks</td>
<td>Required</td>
<td>The attribute is neither required nor conditionally required.</td>
</tr>
<tr>
<td>PersonLastName</td>
<td>Last name of a person party.</td>
<td>No prerequisite tasks</td>
<td>Required</td>
<td>The attribute is neither required nor conditionally required.</td>
</tr>
</tbody>
</table>
The following figure depicts a sample attribute mapping to import a new contact. In this scenario, the OrigSystem and OrigSystem values identify the party.

![Map Fields](image)

The following figure depicts a sample attribute mapping to update a contact. In this scenario, the OrigSystem and OrigSystem values identify the party and update the person’s middle name.

![Map Fields](image)

Using Reference Files to Evaluate Attributes

For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.
Additional Tips

Tip: You can explore related topics by entering the following keywords in the Oracle Sales Cloud Help Application search.

- Importing person profiles
- Importing contacts
- Importing accounts
- Importing households
- Importing legal entities
- Importing employee resources

Tip: If your data isn’t from a source system, then you can emulate a system by defining a source system and manually generating reference values for your data. For example, if you periodically gather information at sales events or through sales calls, then you might define a source system, using a code of SALES_ACTIVITY. The format of your reference values might concatenate the date, salesperson name, and sequence number, such as 01_Jan_2014_Jackson_01.

Related Topics

- Importing Contacts Using File-Based Import: Explained
- Importing Employee Resources Using File-Based Import: Explained

Importing Contact Points Using File-Based Import: Quick Start

This topic describes how to get you started with importing contact points. It includes the following information:

- An overview of contact points
- An example of how to identify and associate records
- The minimum data required and the prerequisites tasks
- How to access and use reference files to evaluate attributes
- Additional Tips

⚠️ Note: Information for associating addresses with persons and organizations is included in this topic, although contact points don’t formally include addresses.

The following table lists the file-based import objects and target objects that you can use to import contact points directly associated with the person.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Party Type</th>
<th>Target Objects for Importing Contact Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contact</td>
<td>Person</td>
<td>Email, Fax, InstantMessenger, Mobile, Phone, Address</td>
</tr>
<tr>
<td>Account</td>
<td>Organization</td>
<td>PrimaryPhone, Fax, Url, Address</td>
</tr>
</tbody>
</table>
The following table lists the file-based import objects and target objects that you can use to import contact points for a contact relationship.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Party Type</th>
<th>Target Objects for Importing Contact Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>Household</td>
<td>Organization</td>
<td>E-mail, Fax, InstantMessenger, Mobile, Phone, Address, Url</td>
</tr>
<tr>
<td>Legal Entity</td>
<td>Organization</td>
<td>Fax, PrimaryPhone, PrimaryAddress, Url</td>
</tr>
<tr>
<td>Lead</td>
<td>Organization (lead organization)</td>
<td>OrganizationProfileFax, OrganizationProfilePrimaryPhone, OrganizationAddress, Url</td>
</tr>
<tr>
<td>Lead</td>
<td>Person (lead individual)</td>
<td>Email, PersonProfileFax, InstantMessenger, Mobile, PersonAddress, PersonProfilePrimaryPhone</td>
</tr>
<tr>
<td>Lead</td>
<td>Person (contact at lead organization)</td>
<td>ContactEmail, ContactFax, ContactInstantMessenger, ContactMobile, ContactPrimaryAddress, ContactPrimaryPhone</td>
</tr>
<tr>
<td>Partner</td>
<td>Organization</td>
<td>Fax, PrimaryPhone, PrimaryAddress, Url</td>
</tr>
<tr>
<td>Partner Contact</td>
<td>Person (contact for a partner)</td>
<td>Email, Fax, InstantMessenger, Mobile, PrimaryPhone, PrimaryAddress</td>
</tr>
<tr>
<td>Campaign Response</td>
<td>Organization (respondent organization)</td>
<td>OrganizationProfileFax, OrganizationProfilePrimaryPhone, OrganizationAddress, Url</td>
</tr>
<tr>
<td>Campaign Response</td>
<td>Person (individual respondent)</td>
<td>Email, PersonProfileFax, InstantMessenger, Mobile, PersonProfilePrimaryPhone, PersonAddress</td>
</tr>
<tr>
<td>Campaign Response</td>
<td>Person (contact at respondent organization)</td>
<td>ContactEmail, ContactFax, ContactInstantMessenger, ContactMobile, ContactPrimaryAddress, ContactPrimaryPhone</td>
</tr>
<tr>
<td>Contact</td>
<td>Person (a contact for another root object such as account)</td>
<td>ContactEmail, ContactFax, ContactInstantMessenger, ContactMobile, ContactAddress, ContactPhone</td>
</tr>
<tr>
<td>Account</td>
<td>Person (a contact for an account)</td>
<td>ContactEmail, ContactFax, ContactInstantMessenger, ContactMobile, ContactPrimaryAddress, ContactPrimaryPhone</td>
</tr>
</tbody>
</table>
**Contact Points Overview**

A contact is an individual who is a customer or a prospect, or a contact for an existing customer or consumer, or a contact that does not yet have an established business association with a customer or consumer. Thus, a contact could be an employee of a customer organization, a person you may have met who could help with your business, or a prospective or current individual customer. The contact object contains information that identifies the contact and offers the contact points of the contact. Contact points can be geographical addresses, phone numbers, e-mail IDs, URLs, messenger IDs, and so on. The contact object also contains contact preference information for the contact.

**Identifying and Associating Records with Each Other**

To add contact to a party or to update existing contact point assignments using file-based import, your source file must contain information about the account, contact, or household party to which the contact point belongs. You map your source file data to the account, contact, or household profile target object, the relationship record that captures the interconnection between the contact and the account.

If you’re creating the party and the contact point assignment in the same import activity, then group the party and contact point data together in your source file. The import process can import the party, and then identify the party record so that it can import the associated contact point.

To add a contact point to an existing party, or to update an existing contact point record, your source file must include values that enable the import process to identify the existing records. These values will be source system and source system reference value combination, an Oracle Sales Cloud internal ID, or public unique identifiers, such as business keys or external IDs.

If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file. The file-based data import process stores a cross-reference between the source system information and Oracle Sales Cloud internal ID. The import process uses the combination of source system and source system reference value to identify the existing record.

You can use external IDs or business keys to identify and associate records with each other when you’re importing a smaller set of records. Business keys are a set of attributes that uniquely identify a party record. For example, a person’s last name and first name uniquely identify a person. An external ID is a unique record identifier from a system outside of Oracle Sales Cloud. You can use the following business keys or external IDs in Oracle Sales Cloud:

- Organization name for organizations
- First name and last name for persons
- Contact information, which is a combination of email ID, mobile number, IM, and URL.
- Address information, which is a combination of address1, address2, city, and postal code.

You can configure source systems to identify the source of the data that you’re importing. Source systems are external sources of data that are used to import data into Oracle Sales Cloud. Oracle Sales Cloud uses source system references to create references between source IDs and the Oracle Sales Cloud database IDs. You can configure source systems either by using the **Manage Trading Community Source System** task, or by importing the source system information into Oracle Sales Cloud using the Source System Reference import object.
When the source of your data isn't an external system and you don't intend to regularly update the data, you don't need the source system information. To import updates to your existing data, you can export the Oracle Sales Cloud internal ID and add it to your source file. The import process uses the internal ID to identify the existing record.

Minimum Data Required and the Prerequisite Setup Tasks

The minimum data that is required to import contact point information depends on the following:

- The purpose of the import. The data requirements are different when you're creating both the party and the contact point in the same import batch, adding contact points to an existing party, or updating a contact point record.
- Identifying and associating records. In some cases, you can choose which attributes you want to provide to the import process when identifying and associating records.

The values that you provide in your source file may require a setup task or manual step to ensure the information passes the import validation. Before you prepare your source file, complete the prerequisite steps to determine the data values, and then include those values in your source file.

The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create or update a contact point data record. You can optionally include attributes that are available for import in your import file but that aren't listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Adding Contact Points in the Same Batch</th>
<th>Adding Contact Point Records to an Existing Party</th>
<th>Updating Contact Point Records</th>
</tr>
</thead>
<tbody>
<tr>
<td>PartyId</td>
<td>The Oracle Sales Cloud internal ID for the party (account, contact, household, partner, or legal entity) to which the contact point belongs.</td>
<td>Identify the party ID value by exporting the Party object using Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
<td></td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• PartyId (Oracle Sales Cloud internal ID).</td>
<td></td>
<td>• PartyId (Oracle Sales Cloud internal ID).</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
<td></td>
<td>• PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Party Number (public unique identifier)</td>
<td></td>
<td>• Party Number (public unique identifier)</td>
<td></td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Party and Adding Contact Points in the Same Batch</td>
<td>Adding Contact Point Records to an Existing Party</td>
<td>Updating Contact Point Records</td>
</tr>
<tr>
<td>----------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------------</td>
<td>-------------------------------------------------------------</td>
<td>---------------------------------------------------</td>
<td>--------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>for an organization profile), which is CONTACT_NUMBER for contact.</td>
<td>for an organization profile), which is CONTACT_NUMBER for contact.</td>
</tr>
<tr>
<td>PartyOrigSystem</td>
<td>The code representing the source system for the party (account, contact, household, partner, or legal entity) to which the contact point belongs.</td>
<td></td>
<td></td>
<td>Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>Conditionally required. Provide reference information to identify the existing party. The reference information can be: • PartyId (Oracle Sales Cloud internal ID). • PartyOrigSystem and PartyOrigSysteml (source system code and source system reference values) provided when importing the organization profile. • Party Number (public unique identifier for an organization profile), which is CONTACT_NUMBER for contact.</td>
</tr>
<tr>
<td>PartyOrigSystemReference</td>
<td>The reference number or text representing the source system unique ID for the party (account, contact, household, partner, or legal entity) to which the contact point belongs.</td>
<td></td>
<td></td>
<td>Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>Conditionally required. Provide reference information to identify the existing party. The reference information can be: • PartyId (Oracle Sales Cloud internal ID). • PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile. • Party Number (public unique identifier for an organization profile), which is CONTACT_NUMBER for contact.</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Party and Adding Contact Points in the Same Batch</td>
<td>Adding Contact Point Records to an Existing Party</td>
<td>Updating Contact Point Records</td>
</tr>
<tr>
<td>-----------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-------------------------</td>
<td>-----------------------------------------------------------</td>
<td>--------------------------------------------------</td>
<td>--------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>System code and the unique reference value for the source system’s record in your source file.</td>
<td>Cloud internal ID.</td>
<td>Cloud internal ID.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
<td>• PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Party Number (public unique identifier for an organization profile), which is CONTACT_NUMBER for contact.</td>
<td>• Party Number (public unique identifier for an organization profile), which is CONTACT_NUMBER for contact.</td>
</tr>
<tr>
<td>ObjectKey</td>
<td>The unique ID for the existing contact point record in the Oracle Sales Cloud destination table.</td>
<td>Identify the contact point ID for an existing contact point by exporting the contact point object using the Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• ObjectKey (Oracle Sales Cloud internal ID).</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• CpOrigSystem and CpOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Party Number (public unique identifier for an organization profile), which is CONTACT_NUMBER for contact.</td>
<td></td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Party and Adding Contact Point Records in the Same Batch</td>
<td>Adding Contact Point Records to an Existing Party</td>
<td>Updating Contact Point Records</td>
</tr>
<tr>
<td>------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------------</td>
<td>------------------------------------------------------------------</td>
<td>---------------------------------------------------</td>
<td>--------------------------------</td>
</tr>
<tr>
<td>ContactPointType</td>
<td>Type of the contact point like phone, email and so on.</td>
<td>Identify valid lookup codes for the COMMUNICATION_TYPE lookup using the Setup and Maintenance, Manage Contact Point Lockups task.</td>
<td>Required</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>CpOrigSystem</td>
<td>A source system code that identifies the original source system of the contact point.</td>
<td>Identify the source system code that was used when you imported the contact point in a prior batch, or identify the source system code that you'll use when importing the contact point in the same batch as this contact point contact preference.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
</tbody>
</table>

Provide reference information to identify the existing party. The reference information can be:

- **ObjectKey** (Oracle Sales Cloud internal ID).
- **CpOrigSystem** and **CpOrigSystemReference** (source system code and source system reference values) provided when importing the organization profile.
- **Party Number** (public unique identifier for an organization profile), which is CONTACT_NUMBER for contact.
## Attribute | Description | Prerequisite Setup Task | Creating Party and Adding Contact Points in the Same Batch | Adding Contact Point Records to an Existing Party | Updating Contact Point Records
--- | --- | --- | --- | --- | ---
CpOrigSystemReference | A source system reference that identifies the unique ID of the contact point in your legacy or external system. | Identify the reference value from your source system that was used when you imported the contact point in a prior batch, or identify the source system reference that you’ll use when importing the contact point in the same batch as this contact point contact preference. | Conditionally required. | Conditionally required. | Conditionally required.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>CpOrigSystemReference</td>
<td>A source system reference that identifies the unique ID of the contact point in your legacy or external system.</td>
<td>Identify the reference value from your source system that was used when you imported the contact point in a prior batch, or identify the source system reference that you’ll use when importing the contact point in the same batch as this contact point contact preference.</td>
</tr>
</tbody>
</table>

There are four sets of reference information that you can use to refer to a record. However, you must use only one of them to refer to a record; else, the import process would result in an error. The import process recognizes reference information in this order:

1. Oracle Sales Cloud ID such as PartyId, ObjectKey, and so on.
2. Public User ID such as organization name, contact name, and so on.
3. Origin system and origin system reference
4. Other reference information such as business keys.
The following figure depicts a sample attribute mapping to import a contact point for an existing party. In this scenario, the OrigSystem and OrigSystemReference values identify the party.

Using Reference Files to Evaluate Attributes

For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

Additional Tips

Tip: You can explore related topics by entering the following keywords in Oracle Sales Cloud Help Application search. You can access help from any Oracle Sales Cloud page by clicking Help in the top-right corner, and selecting Applications Help.

- Importing contacts
- Importing accounts
- Importing households
- Importing legal entities
- Importing employee resources

Tip: If your data isn’t from a source system, then you can emulate a system by defining a source system and manually generating reference values for your data. For example, if you periodically gather information at sales events or through sales calls, then you might define a source system, using a code of SALES_ACTIVITY. The format of your reference numbers might concatenate the date, salesperson name, and sequence number, such as 01_Jan_2014_Jackson_01.

Related Topics

- Contact Import Objects: How They Work Together
Importing Contact Point Preferences Using File-Based Import: Quick Start

This topic describes how to get you started with importing contact point preferences. It includes the following information:

- An overview of contact point preferences
- An example of how to identify and associate records
- The minimum data required and the prerequisites tasks
- How to access and use reference files to evaluate attributes
- Additional Tips

> **Note:** Information for associating addresses with persons and organizations is included in this topic, although contact points don't formally include addresses.

The following table lists the file-based import objects and target objects that you can use to import contact point directly associated with the person.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Party Type</th>
<th>Target Objects for Importing Contact Point Preferences</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contact</td>
<td>Person</td>
<td>EmailContactPreference, FaxContactPreference, and so on. InstantMessenger has no contact point preference.</td>
</tr>
<tr>
<td>Account</td>
<td>Organization</td>
<td>PrimaryPhoneContactPreference, FaxContactPreference, AddressContactPreference Url has no contact point preference.</td>
</tr>
<tr>
<td>Household</td>
<td>Organization</td>
<td>EmailContactPreference, FaxContactPreference, and so on. InstantMessenger and Url have no contact point preference.</td>
</tr>
<tr>
<td>Legal Entity</td>
<td>Organization</td>
<td>FaxContactPreference, PrimaryPhoneContactPreference, PrimaryAddressContactPreference Url has no contact point preference.</td>
</tr>
</tbody>
</table>
### Importing Contacts

The following table lists the file-based import objects and target objects that you can use to import contact point preferences directly associated with a contact relationship.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Party Type</th>
<th>Target Objects for Importing Contact Point Preferences</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lead</td>
<td>Organization (lead organization)</td>
<td>No contact point preferences</td>
</tr>
<tr>
<td>Lead</td>
<td>Person (lead individual)</td>
<td>No contact point preferences</td>
</tr>
<tr>
<td>Lead</td>
<td>Person (contact at lead organization)</td>
<td>No contact point preferences</td>
</tr>
<tr>
<td>Partner</td>
<td>Organization</td>
<td>FaxContactPreference, PrimaryPhoneContactPreference, PrimaryAddressContactPreference, Url has no contact point preference.</td>
</tr>
<tr>
<td>Partner Contact</td>
<td>Person (contact for a partner)</td>
<td>EmailContactPreference, FaxContactPreference, and so on. InstantMessenger has no contact point preference.</td>
</tr>
<tr>
<td>Campaign Response</td>
<td>Organization (respondent organization)</td>
<td>No contact point preferences</td>
</tr>
<tr>
<td>Campaign Response</td>
<td>Person (individual respondent)</td>
<td>No contact point preferences</td>
</tr>
<tr>
<td>Campaign Response</td>
<td>Person (contact at respondent organization)</td>
<td>No contact point preferences</td>
</tr>
<tr>
<td>Contact</td>
<td>Person (a contact for another root object such as account)</td>
<td>ContactEmailContactPreference, ContactFaxContactPreference, and so on. ContactInstantMessenger has no contact point preference.</td>
</tr>
<tr>
<td>Account</td>
<td>Person (a contact for an account)</td>
<td>ContactEmailContactPreference, ContactFaxContactPreference, and so on. ContactInstantMessenger has no contact point preference.</td>
</tr>
<tr>
<td>Legal Entity</td>
<td>Person (contact for a legal entity)</td>
<td>ContactEmailContactPreference, ContactFaxContactPreference, and so on. ContactInstantMessenger has no contact point preference.</td>
</tr>
</tbody>
</table>
Contact Point Preferences Overview

Every contact has contact points, such as phone numbers, faxes, e-mail IDs, and so on, and each contact point can have associated contact point preferences. Contact point preferences indicate whether a contact point can be used to contact a person. For example, Consider Pinnacle Flowers and Gifts, a customer of Vision Enterprises. At Pinnacle Flowers and Gifts, the designated contact person for the company must not be contacted by phone, and the customer contact for the logistics service that Vision Enterprises offers does not want to be contacted using faxes. Pinnacle Flowers and Gifts has assigned phone and fax numbers to most of its contacts and has made these numbers available to its vendors, but the designated contacts have specific preferences about how they do and do not want to be contacted. Vision Enterprises must store these preferences to ensure that their employees do not reach out to customer contacts using contact points that the contacts do not want to be used.

Identifying and Associating Records with Each Other

To add contact point preferences to a party or to update existing contact point preference assignments using file-based import, your source file must contain information about the account, contact, or household party to which the contact point preference belongs. You map your source file data to the account, contact, or household profile target object, the relationship record that captures the interconnection between the contact point and the contact point preference target object.

If you’re creating the party and contact point preference assignment in the same import activity, then group the party and contact point preference data together in your source file. The import process can import the party, and then identify the party record so that it can import the associated contact point preference. To add a contact point preference to an existing party, or to update an existing contact point preference record, your source file must include values that enable the import process to identify the existing records.

If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file. The file-based data import process stores a cross-reference between the source system information and Oracle Sales Cloud internal ID. The import process uses the combination of source system and source system reference value to identify the existing record.

You can configure source systems to identify the source of the data that you’re importing. Source systems are external sources of data that are used to import data into Oracle Sales Cloud. Oracle Sales Cloud uses source system references to create references between source IDs and the Oracle Sales Cloud database IDs. You can configure source systems either by using the Manage Trading Community Source System task, or by importing the source system information into Oracle Sales Cloud using the Source System Reference import object.

When the source of your data isn’t an external system and you don’t intend to regularly update the data, you don’t need the source system information. To import updates to your existing data, you can export the Oracle Sales Cloud internal ID and add it to your source file. The import process uses the internal ID to identify the existing record.

Minimum Data Required and the Prerequisite Setup Tasks

The minimum data that is required to import contact point information depends on the following:

- The purpose of the import. The data requirements are different when you’re creating both the party and contact point preference in the same import batch, adding contact point preferences to an existing party, or updating a contact preference record.
- Identifying and associating records. In some cases, you can choose which attributes you want to provide to the import process when identifying and associating records.
The values that you provide in your source file may require a setup task or manual step to ensure the information passes the import validation. Before you prepare your source file, complete the prerequisite steps to determine the data values, and then include those values in your source file.

The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create or update a contact point preference data record. You can optionally include attributes that are available for import in your import file but that aren’t listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Adding Contact Point Preferences in the Same Batch</th>
<th>Adding Contact Point Preference Records to an Existing Party</th>
<th>Updating Contact Point Preference Records</th>
</tr>
</thead>
<tbody>
<tr>
<td>PartyId</td>
<td>The Oracle Sales Cloud internal ID for the party (account, contact, household, partner, or legal entity) to which the contact point preference belongs.</td>
<td>Identify the party ID value by exporting the Party object using Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td>PartyOrigSystem</td>
<td>The code representing the source system for the party (account, contact, household, partner, or legal entity) to which the contact point preference belongs.</td>
<td>Identify the source system code that was used when you imported the party in a prior batch, or identify the source system code that you will use when importing the party in the same batch as this contact point preference.</td>
<td>Conditionally required.</td>
<td>If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td>Partyld (Oracle Sales Cloud internal ID).</td>
<td></td>
<td></td>
<td></td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
<td></td>
</tr>
<tr>
<td>PartyOrigSystem and PartyOrigSystemI (source system code and source system reference values) provided when importing the organization profile.</td>
<td></td>
<td></td>
<td></td>
<td>• Partyld (Oracle Sales Cloud internal ID).</td>
<td></td>
</tr>
<tr>
<td>PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
<td></td>
<td></td>
<td></td>
<td>• Partyld (Oracle Sales Cloud internal ID).</td>
<td></td>
</tr>
</tbody>
</table>

206
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Adding Contact Point Preferences in the Same Batch</th>
<th>Adding Contact Point Preference Records to an Existing Party</th>
<th>Updating Contact Point Preference Records</th>
</tr>
</thead>
<tbody>
<tr>
<td>PartyOrigSystemReference</td>
<td>The reference number or text representing the source system unique ID for the party (account, contact, household, partner, or legal entity) to which the contact point preference belongs.</td>
<td>Identify the reference value from your source system that was used when you imported the party in a prior batch, or identify the source system reference that you will use when importing the party in the same batch as this contact point preference.</td>
<td>Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>Conditionally required. Provide reference information to identify the existing party. The reference information can be:</td>
<td>Conditionally required. Provide reference information to identify the existing party. The reference information can be:</td>
</tr>
<tr>
<td>ContactType</td>
<td>The method of contact that should or should not be made, such as call or mail.</td>
<td>Identify the contact type from the Setup and Maintenance, Manage Trading Community Common Lookups task.</td>
<td>Required</td>
<td>Required</td>
<td>The attribute is neither required nor conditionally required.</td>
</tr>
<tr>
<td>PreferenceCode</td>
<td>The lookup code that represents the contact method preference phrase, such as Do Use, Opt In, Do not use.</td>
<td>Identify the preference code from the Setup and Maintenance, Manage Trading Community Common Lookups task.</td>
<td>Required</td>
<td>Required</td>
<td>The attribute is neither required nor conditionally required.</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Party and Adding Contact Point Preferences in the Same Batch</td>
<td>Adding Contact Point Preference Records to an Existing Party</td>
<td>Updating Contact Point Preference Records</td>
</tr>
<tr>
<td>----------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>--------------------------</td>
<td>----------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------</td>
<td>---------------------------------------------</td>
</tr>
<tr>
<td>PreferenceStartDate</td>
<td>The date from which a contact preference is valid.</td>
<td>Required</td>
<td>Required</td>
<td>The attribute is neither required nor conditionally required.</td>
<td></td>
</tr>
<tr>
<td>RequestedBy</td>
<td>The lookup code that represents the source of the contact preference request, such as internal requirement or requested by the party.</td>
<td>Required</td>
<td>Required</td>
<td>The attribute is neither required nor conditionally required.</td>
<td></td>
</tr>
<tr>
<td>ContactPrefOrigSysRef</td>
<td>The source system reference that identifies the unique ID of the contact preference in your legacy or external system.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
<td>Provide reference information to identify the existing party.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>The reference information can be:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• ObjectKey (Oracle Sales Cloud internal ID).</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• AdditionalPartyId and AdditionalPartyId (source system code and source system reference values) provided when importing the organization profile.</td>
<td></td>
</tr>
<tr>
<td>ContactPrefOrigSystem</td>
<td>A source system code that identifies the original source system of the contact preference.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
<td>Provide reference information to identify the existing party.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>The reference information can be:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• ObjectKey (Oracle Sales Cloud internal ID).</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• AdditionalPartyIdOrigSystem and AdditionalPartyIdOrigSystem (source system code and source system reference values) provided when importing the organization profile.</td>
<td></td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Party and Adding Contact Point Preferences in the Same Batch</td>
<td>Adding Contact Point Preference Records to an Existing Party</td>
<td>Updating Contact Point Preference Records</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------</td>
<td>-------------------------------------------</td>
</tr>
<tr>
<td>ObjectKey</td>
<td>The unique ID of the existing contact preference record in the Oracle Sales Cloud destination table.</td>
<td>Identify the contact preference ID of an existing contact preference by exporting the Contact Preference object using the Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>Conditionally required. Provide reference information to identify the existing party. The reference information can be:</td>
</tr>
<tr>
<td>CpOrigSystem</td>
<td>A source system code that identifies the original source system of the contact point.</td>
<td>Identify the source system code that was used when you imported the contact point in a prior batch, or identify the source system code that you’ll use when</td>
<td>Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported</td>
<td>Conditionally required. If the source of your data is an external system, and if you intend to import updates to previously imported</td>
<td>Conditionally required. Provide reference information to identify the existing party. The reference information can be:</td>
</tr>
</tbody>
</table>
## Attribute | Description | Prerequisite Setup Task | Creating Party and Adding Contact Point Preferences in the Same Batch | Adding Contact Point Preference Records to an Existing Party | Updating Contact Point Preference Records
---|---|---|---|---|---
importing the contact point in the same batch as this contact point contact preference. | records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file. | records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file. | party. The reference information can be:
  - **ObjectKey** (Oracle Sales Cloud internal ID).
  - **CpOrigSystem** and **CpOrigSystemReference** (source system code and source system reference values) provided when importing the organization profile.

### CpOrigSystemReference
A source system reference that identifies the unique ID of the contact point in your legacy or external system.

**Identify the reference value from your source system that was used when you imported the contact point in a prior batch, or identify the source system reference that you'll use when importing the contact point in the same batch as this contact point contact preference.**

**Conditionally required.**

If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.

**Conditionally required.**

If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.

**Conditionally required.**

Provide reference information to identify the existing party. The reference information can be:

  - **ObjectKey** (Oracle Sales Cloud internal ID).
  - **CpOrigSystem** and **CpOrigSystemReference** (source system code and source system reference values) provided when importing the organization profile.
The following figure depicts a sample attribute mapping to import a contact point preference for an existing party. In this scenario, the OrigSystem and OrigSystemReference values identify the party.

Using Reference Files to Evaluate Attributes
For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

Additional Tips
Tip: You can explore related topics by entering the following keywords in Oracle Sales Cloud Help Application search. You can access help from any Oracle Sales Cloud page by clicking Help in the top-right corner, and selecting Applications Help.

- Importing contacts
- Importing accounts
- Importing households
• Importing legal entities
• Importing employee resources

Tip: If your data isn’t from a source system, then you can emulate a system by defining a source system and manually generating reference values for your data. For example, if you periodically gather information at sales events or through sales calls, then you might define a source system, using a code of SALES_ACTIVITY. The format of your reference numbers might concatenate the date, salesperson name, and sequence number, such as 01_Jan_2014_Jackson_01.

Related Topics
• Contact Import Objects: How They Work Together
• Account Import Objects: How They Work Together
• Household Import Objects: How They Work Together
• Importing Contact Points Using File-Based Import: Quick Start

Importing Contact Jobs Using File-Based Import: Quick Start

This topic describes how to get you started with importing contact jobs. It includes the following information:
• An example of how to identify and associate records
• The minimum data required and the prerequisite tasks
• How to access and use reference files to evaluate attributes
• Additional tips

A contact job enables you to capture information related to the work that a contact does. The target object contains attributes that records data such as the name and code of the department where the contact works, the job title of the contact, and so on.

The following table lists the file-based import objects and target objects that you can use to import contact jobs.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Party Type</th>
<th>Target Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contact</td>
<td>Person</td>
<td>Contact.Job</td>
</tr>
<tr>
<td>Account</td>
<td>Person (a contact for a account)</td>
<td>Contact.Job</td>
</tr>
<tr>
<td>Lead</td>
<td>Lead Contact</td>
<td>Contact.Job</td>
</tr>
<tr>
<td>Legal Entity</td>
<td>Person (a contact for a legal entity)</td>
<td>Contact.Job</td>
</tr>
<tr>
<td>Partner</td>
<td>Partner Contact</td>
<td>Job</td>
</tr>
<tr>
<td>Response</td>
<td>Response Contact</td>
<td>Contact.Job</td>
</tr>
</tbody>
</table>
Identifying and Associating Records with Each Other

To add contact jobs to a party or to update existing contact job assignments using file-based import, your source file must contain information about the account, contact, or household party to which the contact job belongs. You map your source file data to the account, contact, or household profile target object and to the contact job target object.

If you’re creating the party and the contact job assignment in the same import activity, then group the party and contact job data together in your source file. The import process can import the party, and then identify the party record so that it can import the associated contact job.

To add a contact job to an existing party, or to update an existing contact job record, your source file must include values that enable the import process to identify the existing records.

If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file. The file-based data import process stores a cross-reference between the source system information and Oracle Sales Cloud internal ID. The import process uses the combination of source system and source system reference value to identify the existing record.

You can configure source systems to identify the source of the data that you’re importing. Source systems are external sources of data that are used to import data into Oracle Sales Cloud. Oracle Sales Cloud uses source system references to create references between source IDs and the Oracle Sales Cloud database IDs. You can configure source systems either by using the Manage Trading Community Source System task, or by importing the source system information into Oracle Sales Cloud using the Source System Reference import object.

When the source of your data isn’t an external system and you don’t intend to regularly update the data, you don’t need the source system information. To import updates to your existing data, you can export the Oracle Sales Cloud internal ID and add it to your source file. The import process uses the internal ID to identify the existing record.

Minimum Data Required and the Prerequisite Setup Tasks

The minimum data that is required to import contact job information depends on the following:

- The purpose of the import. The data requirements are different when you’re creating both the party and the contact job in the same import batch, adding contact jobs to an existing party, or updating a contact job record.
- Identifying and associating records. In some cases, you can choose which attributes you want to provide to the import process when identifying and associating records.

The values that you provide in your source file may require a setup task or manual step to ensure the information passes the import validation. Before you prepare your source file, complete the prerequisite steps to determine the data values, and then include those values in your source file.

The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create or update a contact job record. You can optionally include attributes that are available for import in your import file but that aren’t listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Adding contact job in the Same Batch</th>
<th>Adding contact job to an Existing Party</th>
<th>Updating Existing Contact Jobs</th>
</tr>
</thead>
<tbody>
<tr>
<td>PartyId</td>
<td>The Oracle Sales Cloud internal ID for the party (account, contact, household)</td>
<td>Identify the party ID value by exporting the Party object using the Setup</td>
<td>Required</td>
<td>Conditionally required</td>
<td>Conditionally required</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Party and Adding contact job in the Same Batch</td>
<td>Adding contact job to an Existing Party</td>
<td>Updating Existing Contact Jobs</td>
</tr>
<tr>
<td>------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>PartyOrigSystem</td>
<td>The code representing the source system for the party (account, contact, household, or legal entity) to which the address belongs.</td>
<td>and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>To identify the party, provide either the internal ID (PartyId) or both the source system code (PartyOrigSystem) and source system reference value (PartyOrigSystemReference) that were provided in a prior import of the party.</td>
<td>To identify the party, provide either the internal ID (PartyId) or both the source system code (PartyOrigSystem) and source system reference value (PartyOrigSystemReference) that were provided in a prior import of the party.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td>PartyOrigSystemReference</td>
<td>The reference number or text representing the source system unique ID for the party (account, contact, household, or legal entity) to which the address belongs.</td>
<td>Identify the reference value from your source system that was used when you imported the party in a prior batch, or identify the source system reference value that you'll use when importing the party in the same batch as this address.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td>RelationshipId</td>
<td>The unique ID for the existing contact's relationship with the organization or person record in the destination table.</td>
<td>Identify the Relationship ID for an existing contact's relationship by exporting the Relationship object using the Setup and Maintenance,</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
</tbody>
</table>

To identify the party, provide either the internal ID (PartyId) or both the source system code (PartyOrigSystem) and source system reference value (PartyOrigSystemReference) that were provided in a prior import of the party.
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Adding contact job in the Same Batch</th>
<th>Adding contact job to an Existing Party</th>
<th>Updating Existing Contact Jobs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>both the source system code (RelOrigSystem) and source system reference value (RelOrigSystemReference) that were provided in a prior import of the party.</td>
</tr>
<tr>
<td>RelOrigSystem</td>
<td>The source system code that identifies the original source system of the contact’s relationship with the organization or person.</td>
<td>No prerequisite tasks.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>To identify the existing contact job record, provide either the internal ID (ObjectKey) or both the source system code (RelOrigSystem) and source system reference value (RelOrigSystemReference) that were provided in a prior import of the party.</td>
</tr>
<tr>
<td>RelOrigSystemReference</td>
<td>The ID that identifies the contact’s relationship with the organization or person in your legacy or external system.</td>
<td>No prerequisite tasks.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file.</td>
<td>To identify the existing contact job record, provide either the internal ID (ObjectKey) or both the source system code (RelOrigSystem) and source system reference value (RelOrigSystemReference) that were provided in a prior import of the party.</td>
</tr>
<tr>
<td>ContactNumber</td>
<td>The contact job’s unique public identifier value.</td>
<td>Obtain the contact number for an existing contact using the Customers work area.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Required if you are updating a contact job and are not providing the object key.</td>
</tr>
</tbody>
</table>
### Attribute Mapping for Importing Contacts

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Adding contact job in the Same Batch</th>
<th>Adding contact job to an Existing Party</th>
<th>Updating Existing Contact Jobs</th>
</tr>
</thead>
<tbody>
<tr>
<td>ObjectKey</td>
<td>The unique ID for the existing contact job record in the destination table.</td>
<td>Identify the contact ID for an existing contact by exporting the Organization Contact object using the Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>Conditionally required. Required if you are updating a contact job and are not providing the contact number.</td>
</tr>
</tbody>
</table>

The following figure depicts a sample attribute mapping to import a contact job for an existing party. In this scenario, the OrigSystem and OrigSystemReference values identify the party.

### Using Reference Files to Evaluate Attributes

For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

### Additional Tips

Tip: You can explore related topics by entering the following keywords in Oracle Sales Cloud Help Application search. You can access help from any Oracle Sales Cloud page by clicking Help in the top-right corner, and selecting Applications Help.

- Importing contact jobs
- Importing accounts
- Importing contacts
- Importing groups

Tip: If your data isn’t from a source system, then you can emulate a system by defining a source system and manually generating reference values for your data. For example, if you periodically gather information at sales events or through sales calls, then you might define a source system, using a code of SALES_ACTIVITY. The format of your reference numbers might concatenate the date, salesperson name, and sequence number, such as 01_Jan_2014_Jackson_01.
7 Importing Country Structures

Importing Country Structures Using File-Based Import: Explained

This topic explains how to prepare and import country structure data from an external data source using the File-Based Data Import feature. A country structure is a hierarchical grouping of geography types for a country. For example, the geography structure for the United States has the geography type of State at the top, followed by the County, then the City, and finally the Postal Code.

You can use the country structure to set up the following:

- The relationships between geographies within a country
- The types of geographies that you can define for a country

Consider the following questions when importing your data:

- How does your legacy system or source system represent the geography data compared to how the application represents the same data?
- Do you have to configure values in the application to map to your data values?
- Do you have to customize the application to capture additional attributes that are critical to the way you do business?
- What import features are available for importing your business object?
- How do you verify your imported data?

Comparing Business Object Structures

You must understand how your country structure data corresponds with the data in the application so that you can map your legacy data to the data that the application requires. First, you must understand how the application represents the structure of the data for a country structure.

You must import a separate country structure import object for each country. Each of these import objects must contain the geography types that are used in the country’s structure, organized in a hierarchy using geography level numbers. For example, if you’re importing the country structure of Australia, the country structure could be the following: 1: Country, 2: State, 3: County, 4: Town, 5: ZIP.

Import Objects for the Country Structure

To facilitate importing country structures, the application incorporates the structure of the country structure into import objects. The import object for country structures is GeoStructureLevel.

Comparing Business Object Data

Each import object is a collection of attributes that helps to map your data to the application data and to support one-to-many relationships between the structural components that make up the country structure.

You must understand the attribute details of the import objects so that you can prepare your import data. You can use reference guide files that contain attribute descriptions, values that populate attributes by default when you don’t provide
values, and validation information for each attribute. The validation information includes the navigation path to the task where you can define values in the application. For example, if you have values in your data that correlate to a choice list in the application, then the validation information for that attribute provides the task name in the Setup and Maintenance work area where you can define your values. For additional information, including a list of reference guide file names and locations that you need to complete this task, see the following table.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Related Import Object Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Country Structure</td>
<td>Country Structure Import Objects: How They Work Together</td>
</tr>
</tbody>
</table>

**Extensible Attributes**

If you need to extend the application object to import your legacy or source data, you must use Application Composer to design your object model extensions and to generate the required artifacts to register your extensions and make them available for importing. The corresponding import object is updated with the extensible attributes, which can then be mapped to your source file data. You can use the same source file to import both extensible custom attributes and the standard import object attributes.

**Importing Country Structures Using File-Based Data Import**

For the country structure business object, you must use the File-Based Data Import feature. You prepare XML or text source data files in a form that is suitable for a file-based import. The file-based import process reads the data in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables.

The Define File-Based Data Import Setup and Maintenance task list includes the tasks that are required to configure the import objects, to create source-file mappings, and to schedule the import activities. You submit file-based import activities for each import object. When you’re creating a new country structure, you import the Country Structure object. You must be assigned the Master Data Management Administrator job role to access and submit the import activities for country structures.

**Verifying Your Imported Data**

You can view the list of import activities from the Manage Import Activities page. You can verify your imported data by clicking the Status column for your import activity.

**Related Topics**

- File-Based Data Import: How It Works
- Getting Started with File-Based Import: Documentation Overview
- Country Structure Import Objects: How They Work Together
- Extending Oracle Sales Cloud: How It Works
Country Structure Import Objects: How They Work Together

This topic describes the Country Structure import object. You use the Country Structure import object when you submit a file-based import activity to import your country structure information. This topic introduces the following:

- Target objects for the Country Structure import object
- Target import object attributes
- Reference guide files for target import object attributes

Country Structure Target Import Objects

The Country Structure import object contains one target import object. The target import object organizes the individual attributes of the different aspects of the geography structure. When updating an existing country structure, you must provide the parent reference information of the existing country structure. This reference information connects the imported geography structure to the existing one. Use the ImpGeoStructureLevel target import object to create and update country structure information.

Target Import Object Attributes

You must compare the attributes that you want to import with the target object attributes that are available and with their valid values. To evaluate your source data and Oracle Sales Cloud attributes for mapping and validation, you use a reference file. See the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. A reference guide file includes attribute descriptions, default values, and validations performed by the import process. Review the validation for each attribute to determine whether there are functional prerequisites or prerequisite setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can predefine and manage import mappings using the Manage File Import Mappings task, or you can define the mapping when you define the import activity using the Manage File Import Activities task. Both tasks are available in the Setup and Maintenance work area.

Note: If any of the attributes you want to import does not have an equivalent target object attribute, then review the Application Composer extensibility features for country structures.

Reference Files for Target Import Object Attributes

To access reference files for this object’s target import objects, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

For detailed information on importing geographies using file-based import, refer to Document No. 1481758.1, Importing Master Reference Geography Data, on the Oracle Support site.
The following table lists the reference files that are available by target import object.

<table>
<thead>
<tr>
<th>Target Import Object</th>
<th>Description</th>
<th>Reference Guide File Names</th>
</tr>
</thead>
<tbody>
<tr>
<td>ImpGeoStructureLevel</td>
<td>Information that specifies a country’s geography structure.</td>
<td>HZ_IMP_GEO_STRUCTURE_LEVELS_Reference</td>
</tr>
</tbody>
</table>

**Related Topics**
- File-Based Data Import: How It Works
- Getting Started with File-Based Import: Documentation Overview
- Importing Country Structures Using File-Based Import: Explained
- Importing Country Structures Using File-Based Import: Quick Start
- Extending Oracle Sales Cloud: How It Works

### Importing Country Structures Using File-Based Import: Quick Start

This topic describes how to get you started with importing country structures. It includes the following information:
- An example of how to identify and associate records
- The minimum data required and the prerequisite tasks
- How to access and use reference files to evaluate attributes
- Additional tips

With the country structure of a country you can define which geography types are part of the country, and how the geography types are hierarchically related within the country. For example, you can create geography types called State, City, and Postal Code. Then you can rank the State geography type as the highest level within the country, the City as the second level, and the Postal Code as the lowest level within the country structure.

The following table lists the file-based import objects and target objects that you can use to import country structures.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Target Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Country Structure</td>
<td>ImpGeoStructureLevel</td>
</tr>
</tbody>
</table>

### Identifying and Associating Records with Each Other

To add country structures to a country, or to update existing country structure values using file-based import, your source file must contain information about the country to which the country structure is being assigned. Also, you need to specify the level number at which the geography types you are importing need to be placed. All countries are, by default, placed at Level 1. Create the country structure by increasing the level number as you go down the country structure. Thus, geography types such as states, provinces, and so on, can be placed at Level 2; districts or counties at Level 3, and so on.
To add an additional identifier to an existing party or to update an existing additional identifier record, your source file must include values that enable the import process to identify the existing records. These values will be either source system and source system reference value combination, Oracle Sales Cloud internal ID, or public unique identifiers, such as business keys or external IDs.

If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file. The file-based data import process stores a cross-reference between the source system information and Oracle Sales Cloud internal ID. The import process uses the combination of source system and source system reference value to identify the existing record.

Minimum Data Required and the Prerequisite Setup Tasks

The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create or update a country structure record. You can optionally include attributes that are available for import in your import file but that aren’t listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Country Structure Record for an Existing Country</th>
</tr>
</thead>
<tbody>
<tr>
<td>CountryCode</td>
<td>The code representing the country to which the country structure belongs.</td>
<td>Identify country codes using the Setup and Maintenance, Manage Geographies task.</td>
<td>Required</td>
</tr>
<tr>
<td>GeographyType</td>
<td>A code used for internal reference by Oracle Sales Cloud at an administrative level. For example, this administrative code may be COUNTRY, STATE, COUNTY, CITY, etc. You must include either the LevelNumber or GeographyType field to define the hierarchical structure of your data.</td>
<td>No prerequisite tasks</td>
<td>Required</td>
</tr>
<tr>
<td>LevelNumber</td>
<td>The level number for the geography type in the hierarchy. For example, a country has the level of 1 in the hierarchy because it is the top level.</td>
<td>No prerequisite tasks</td>
<td>Required</td>
</tr>
</tbody>
</table>
The following image depicts a sample attribute mapping to import a country structure.

![Map Fields](image)

Using Reference Files to Evaluate Attributes

For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

Additional Tips

Tip: You can explore related topics by entering the following keywords in Oracle Sales Cloud Help Application search.

- Importing country structures

Tip: For sample country structures and detailed steps for importing country structures, see article 1341174.1 on My Oracle Support.

Related Topics

- Importing Country Structures Using File-Based Import: Explained
- Country Structure Import Objects: How They Work Together
- Geography Structure, Hierarchy, and Validation: How They Fit Together
8 Importing Customer Hierarchies

Importing Customer Hierarchy Using File-Based Import: Explained

This topic describes how to import a customer hierarchy and customer hierarchy members. A customer hierarchy captures the hierarchical relationships that an account has with other accounts. A customer hierarchy consists of hierarchy members.

A customer hierarchy can be used to process payments from one account and apply them to another account in the same hierarchy. It can also be used to create the revenue roll-up report that rolls up revenue numbers from opportunities for all accounts in a hierarchy.

Consider the following questions when importing customer hierarchy information:

- Are all members of the customer hierarchy in Oracle Sales Cloud?
- Do you have to import additional hierarchy members to complete the hierarchy?
- How does your legacy system or source system represent the customer hierarchy compared to how Oracle Sales Cloud represents the same data?
- Do you have to configure values in Oracle Sales Cloud to map your existing data values to the Customer Hierarchy import object?
- Do you have to customize Oracle Sales Cloud to capture additional attributes that are critical to the way you do business?
- What import features are available for importing your business object?
- How do you verify my imported data?

To import a customer hierarchy, you must:

1. Import accounts that must be a part of the hierarchy.
2. Import a hierarchy definition that includes the hierarchy structure and the root node.
3. Import members from the existing accounts into the hierarchy.

Comparing Business Object Structures

Before you can import your legacy or source system customer hierarchy data, you must first analyze the data and see how it corresponds to the account object structure of Oracle Sales Cloud. You must understand how Oracle Sales Cloud represents the structure of the data for a customer hierarchy.

The customer hierarchy import object structure is hierarchical. At the top level is the customer hierarchy, which must exist before you can import customer hierarchy members. The customer hierarchy and customer hierarchy member contain information about the customer hierarchy and customer hierarchy nodes, such as the hierarchy type, hierarchy code, hierarchy name, and hierarchy version.

Comparing Business Object Data

After you understand the structure of the data, then compare the detailed attribute values of your data with the Oracle Sales Cloud data. Each import object is a collection of attributes that helps to map your data to the Oracle Sales Cloud data and to support one-to-many relationships between the structural components that make up the customer hierarchy.
You must understand the attribute details of the import objects so that you can prepare your import data. Reference files with detailed attribute information are provided in the File Based Data Import for Oracle Sales Cloud guide that is available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm).

Reference files contain attribute descriptions, values that populate attributes by default when you don't provide values, and validation information for each attribute. The validation information includes the navigation path to the task where you can define values in Oracle Sales Cloud. For example, if you have values in your data that correspond to a choice list, then the validation information for that attribute will provide the task name in the Setup and Maintenance work area where you can define your values.

Extensible Attributes
If you want to extend the import object to import your legacy or source data, then you must use Application Composer to design your object model extensions and to generate the required artifacts to register your extensions and make them available for importing.

The corresponding import object is updated with the extensible attributes, which you can then map to your source file data. You can use the same source file to import both extensible custom attributes and the standard import object attributes.

Importing Customer Hierarchy Using File-Based Data Import
For the customer hierarchy business object, you must use the File-Based Data Import feature. You prepare XML or text source data files, such as CSV, in a form that is suitable for a file-based import. The file-based import process reads the data in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables. The Define File-Based Data Import Setup and Maintenance task list includes the tasks that are required to configure the import objects, to create source file mappings, and to schedule the import activities. You submit file-based import activities for each import object.

An import activity defines the instructions for processing import data, including the source file, import mapping from the source file to the object and attribute, and the import schedule.

When importing customer hierarchy information, you first import the customer hierarchy information and then the customer hierarchy members for the customer hierarchy. When importing customer hierarchy members, you must provide the parent reference information that refers to the customer hierarchy of the member.

Verifying Your Imported Data
Oracle Sales Cloud provides File-Based Import activity reports, which you can use to verify imported data. You can view the list of import activities from the Manage Import Activities page. You can verify the status of the import activity by clicking the Status column for your import activity. Alternatively, you can also navigate to the Manage Hierarchies task from Setup and Maintenance to view the customer hierarchy information that you have imported.

Related Topics
- Party Hierarchy: Explained
- Customer Hierarchy Import Objects: How They Work Together
- File-Based Data Import: How It Works
- Getting Started with File-Based Import: Documentation Overview
- Extending Oracle Sales Cloud: How It Works
Customer Hierarchy Import Objects: How They Work Together

The customer hierarchy import object imports customer hierarchy and customer hierarchy members. This topic describes the Customer Hierarchy object and introduces the following:

- Target objects for the customer hierarchy import object
- Target import object attributes
- Reference guide files for target import object attributes

Customer Hierarchy Target Import Objects

The target import objects in the customer hierarchy import object are grouped into information about the customer hierarchy and information about the customer hierarchy members. The CustomerHierarchy is the target import object containing attributes to import information about the customer hierarchy, and the CustomerHierarchyMember is the target import object containing attributes to import information about the customer hierarchy members. You can have multiple hierarchy members associated with a customer hierarchy.

To import a customer hierarchy, you must do the following:

1. Import customers who must be part of the hierarchy.
2. Import the customer hierarchy definition that includes the hierarchy structure and the root node.
3. Import existing customers as customer hierarchy members into the hierarchy.

Target Import Object Attributes

You must compare the attributes that you want to import with the target object attributes that are available and with their valid values. To evaluate your source data and Oracle Sales Cloud attributes for mapping and validation, you use a reference file. See the File Based Data Import for Oracle Sales Cloud guide. In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. A reference guide file includes attribute descriptions, default values, and validations performed by the import process. Review the validation for each attribute to determine whether there are functional prerequisites or prerequisite setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can predefine and manage import mappings using the Manage File Import Mappings task, or you can define the mapping when you define the import activity using the Manage File Import Activities task. Both tasks are available in the Setup and Maintenance work area.

Note: If any of the attributes that you want to import do not have an equivalent target object attribute, then review the Application Composer extensibility features for the customer hierarchy.

Reference Files for Target Import Object Attributes

To access reference files for this object’s target import objects, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. The following table lists the reference files that are available by target import object.
Importing Customer Hierarchies Using File-Based Import: Quick Start

This topic describes a few key concepts and provides guidance to get you started on importing customer hierarchies. It includes the following information:

- Customer hierarchy overview
- Identifying and associating records
- The minimum data required and the prerequisite steps
- How to access and use reference files to evaluate attributes
- Additional tips

The following table lists the file-based import objects and target objects that you can use to import customer hierarchies.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Target Objects</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer Hierarchy</td>
<td>CustomerHierarchy</td>
</tr>
</tbody>
</table>

Customer Hierarchy Overview

A customer hierarchy captures the hierarchical relationships that a customer has with other customers, and consists of hierarchy members. A customer hierarchy can be used to process payments from one customer and apply them to another customer in the same hierarchy. It can also be used to create the revenue roll-up report that rolls up revenue numbers from opportunities for all customers in a hierarchy.

You can create multiple versions of a customer hierarchy; however, only one hierarchy version can be active on a specific date. You can create a new hierarchy to represent the structure of a new customer. If the customer has made minor changes to its organization, you can create a new version rather than edit the existing one.
Identifying and Associating Records with Each Other

To update an existing customer hierarchy, your source file must include values that enable the import process to identify the existing records. These values will be either a source system and source system reference value combination or an Oracle Fusion record ID.

If the source of your data is a third party or external system, and you intend to import updates to previously imported records from the same system, you can provide the source system code and the unique reference value for the party record in your source system. Oracle Fusion stores a cross-reference between the source system information and Oracle's record ID. To add a sales account profile to an existing party, you provide the source system code and reference values and the import process uses the value combination to identify the existing party record in Oracle Fusion.

You can set up source systems to identify the source of the data you are importing. Source systems are external sources of data that are used to import data into Oracle Sales Cloud. Source system references are used by the application to create references between source IDs and the Oracle Sales Cloud database IDs. You can configure source systems either by using the Manage Trading Community Source System task or by importing the source system information into Oracle Sales Cloud using the Source System Reference import object.

When the source of your data is not an external system and you want to add a sales account profile to an existing party, you need the Oracle Fusion party record ID in your source file. The import process uses the record ID to identify the existing record. When the source of your data is not an external system and you do not plan to regularly update the data, you do not need the source system information. To import updates to your existing data, you can export the Oracle Fusion record ID and add it to your source file. The import process uses the record ID to identify the existing record.

Minimum Data Required and the Prerequisite Steps

The minimum data that is required to import customer hierarchy information depends on the following:

- The purpose of the import. The data requirements are different when you're creating a customer hierarchy or when you're updating a customer hierarchy record.
- The data requirements are different when using source system information or record IDs to identify and associate records.

The values that you provide in your source file may require a setup task or manual steps to ensure that the information passes the import validation. Before you prepare your source file, complete the prerequisite steps to determine the data values, and then include those values in your source file. The following table lists the minimum attributes required to import customer hierarchies based on two possible scenarios for your import, any prerequisite setup or manual steps needed to prepare your data, and the import attributes that you use to map your data to Oracle Fusion. You can optionally include attributes that are available for import in your import file but that aren't listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Customer Hierarchy Records in the Same Batch</th>
<th>Creating Customer Hierarchy Record for an Existing Party</th>
</tr>
</thead>
<tbody>
<tr>
<td>Language</td>
<td>Indicates the code of the language into which the contents of the translatable columns are translated.</td>
<td>View valid language codes from the Setup and Maintenance, Manage ISO Languages task.</td>
<td>Conditionally required.</td>
<td>Required when creating a new hierarchy (tree version).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Default is US if not specified.</td>
<td></td>
<td>Required when creating a new hierarchy (tree version).</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Party and Customer Hierarchy Records in the Same Batch</td>
<td>Creating Customer Hierarchy Record for an Existing Party</td>
</tr>
<tr>
<td>----------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>--------------------------</td>
<td>---------------------------------------------------------------</td>
<td>----------------------------------------------------------</td>
</tr>
<tr>
<td>SourceLangFlag</td>
<td>This flag indicates whether the language specified by the Language attribute is the source language in which the contents of this record are being created or were originally created.</td>
<td>No task required.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Required when creating a new hierarchy (tree version).</td>
<td>Required when creating a new hierarchy (tree version).</td>
</tr>
<tr>
<td>HierarchyType</td>
<td>Unique Identifier for a tree structure.</td>
<td>Tree structure can be created and managed using the Setup and Maintenance, Manage Tree Structures task.</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>HierarchyCode</td>
<td>User-defined identifier for a tree.</td>
<td></td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HierarchyName</td>
<td>The name of the tree.</td>
<td>No setup task required.</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>HierarchyVersionName</td>
<td>The name of the tree version.</td>
<td>No setup task required.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Required when creating a hierarchy (tree version).</td>
<td>Required when creating a hierarchy (tree version).</td>
</tr>
<tr>
<td>ActionCode</td>
<td>This attribute specifies to insert or to delete the hierarchy (tree version). If provided, valid values are INSERT, UPDATE, and DELETE.</td>
<td>Use the UPDATE value when updating a record. If deleting an existing record, use the value, DELETE. If inserting a record, use the value INSERT.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>The value DELETE is required if you are deleting an existing hierarchy (tree version).</td>
<td>The value DELETE is required if you are deleting an existing hierarchy (tree version).</td>
</tr>
<tr>
<td>AsOfDate</td>
<td>The date that is used to determine the tree version to delete when ActionCode is provided as DELETE. Only one version is active on any given date.</td>
<td>No setup task required.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>A value is required if deleting an existing hierarchy (tree version).</td>
<td>A value is required if deleting an existing hierarchy (tree version).</td>
</tr>
</tbody>
</table>
Using Reference Files to Evaluate Attributes

For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

Additional Tips

Tip: You can explore related topics by entering the following keywords in the Oracle Fusion Help Application search. The Help Application is available from any Oracle Fusion Application page by clicking Help in the top-right corner, and selecting Applications Help.

- Importing customer hierarchies.
- Importing customer hierarchy members.

Tip: If your data is not from a source system, then you can emulate a system by defining a source system and manually generating reference values for your data. For example, if you periodically gather information at sales events or through sales calls, then you might define a source system, using a code of SALES_ACTIVITY. The format of your reference numbers might concatenate the date, salesperson name, and sequence number, such as 01_Jan_2014_Jackson_01.

Related Topics

- Importing Customer Hierarchy Member Using File-Based Import: Quick Start

Importing Customer Hierarchy Member Using File-Based Import: Quick Start

This topic describes a few key concepts and provides guidance to get you started on importing customer hierarchy members. It includes the following information:

- Identifying and associating records
- The minimum data required and the prerequisite steps
- How to access and use reference files to evaluate attributes
- Additional tips

The following table lists the file-based import objects and target objects that you can use to import customer hierarchies.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Target Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer Hierarchy Member</td>
<td>CustomerHierarchyMember</td>
</tr>
</tbody>
</table>

A customer hierarchy captures the hierarchical relationships that a customer has with other customers, and consists of hierarchy members.
Identifying and Associating Records with Each Other

To update an existing customer hierarchy member, your source file must include values that enable the import process to identify the existing records. These values will be either a source system and source system reference value combination or an Oracle Fusion record ID. If the source of your data is a third party or external system, and you intend to import updates to previously imported records from the same system, then you can provide the source system code and the unique reference value for the party record in your source system. Oracle Fusion stores a cross-reference between the source system information and Oracle’s record ID. To add a sales account profile to an existing party, you provide the source system code and reference values and the import process uses the value combination to identify the existing party record in Oracle Fusion.

You can set up source systems to identify the source of the data you are importing. Source systems are external sources of data that are used to import data into Oracle Sales Cloud. Source system references are used by the application to create references between source IDs and the Oracle Sales Cloud database IDs. You can configure source systems either by using the Manage Trading Community Source System task or by importing the source system information into Oracle Sales Cloud using the Source System Reference import object.

When the source of your data is not an external system and you want to add a sales account profile to an existing party, you need the Oracle Fusion party record ID in your source file. The import process uses the record ID to identify the existing record. When the source of your data is not an external system and you do not plan to regularly update the data, you do not need the source system information. To import updates to your existing data, you can export the Oracle Fusion record ID and add it to your source file. The import process uses the record ID to identify the existing record.

Minimum Data Required and the Prerequisite Setup Tasks

The minimum data that is required to import customer hierarchy member information depends on the following:

- The purpose of the import. The data requirements are different when you’re creating both the party and the customer hierarchy member in the same import batch, adding customer hierarchy member to an existing party, or updating a customer hierarchy member record.
- Identifying and associating records. The data requirements are different when you’re using source system information or record IDs to identify and associate records.

Attributes that are available for import in your import file but that aren’t listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Customer Hierarchy Member Records in the Same Batch</th>
<th>Creating Customer Hierarchy Member Record for an Existing Party</th>
</tr>
</thead>
<tbody>
<tr>
<td>ActionCode</td>
<td>This attribute specifies to insert or to delete the hierarchy node (tree node). This attribute is typically used only to delete a record.</td>
<td>Use the UPDATE value when updating a record. If deleting an existing record, use the value, DELETE. If inserting a record, use the value INSERT.</td>
<td>Conditionally required.</td>
<td>A value is required if deleting an existing hierarchy node.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AsOfDate</td>
<td>When ActionCode is provided as DELETE, AsOfDate determines the tree version from which to delete the hierarchy node. Only one version is active on any given date.</td>
<td>No setup task required.</td>
<td>Conditionally required.</td>
<td>A value is required if deleting an existing hierarchy node.</td>
</tr>
</tbody>
</table>
## Attribute

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Party and Customer Hierarchy Member Records in the Same Batch</th>
<th>Creating Customer Hierarchy Member Record for an Existing Party</th>
</tr>
</thead>
<tbody>
<tr>
<td>HierarchyType</td>
<td>Unique Identifier for a tree structure.</td>
<td>Tree structure can be created and managed using the Setup and Maintenance, Manage Tree Structures task.</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>HierarchyCode</td>
<td>User-defined identifier for a tree.</td>
<td>HierarchyCode must exist in the FND_TREE table, column TREE_CODE or, if the hierarchy is being created in the same batch as this node, then HierarchyCode must exist in HZ_IMP_HIERARCHIES_T table (the interface table for the new hierarchy), column TREE_CODE (the HierarchyCode attribute for a hierarchy import).</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>PartyOrigSystem</td>
<td>The source system code that identifies the original source system of the party that you are inserting as a new node or that exists as the node to be deleted. You provide the source system reference (PartyOrigSystemReference) and the source system code (PartyOrigSystem) when the source of your data is your legacy or external system. The import process uses the source system code and source system reference values to: * find the Oracle Sales Cloud internal ID (PartyId) that uniquely identifies an existing party record that was originally imported with the source system reference and source system code values in a prior batch. The party is either being inserted as a new node or is a node to be deleted in this batch.</td>
<td>Predefine your source system code as enabled for parties, using the Setup and Maintenance, Manage Trading Community Source Systems task.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td>PartyOrigSystemReference</td>
<td>The ID that identifies the party that is the parent.</td>
<td>Identify the reference value from your source</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Party and Customer Hierarchy Member Records in the Same Batch</td>
<td>Creating Customer Hierarchy Member Record for an Existing Party</td>
</tr>
<tr>
<td>----------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>PartyId</td>
<td>The unique internal ID of the existing party record that is either being inserted as a new node or is an existing node to be deleted in this batch.</td>
<td>You can obtain the Party ID for an existing party by exporting the Party object using the Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>Conditionally required.</td>
<td>If you're creating a new node, then identify the parent node by providing either ParentPartyId or both PartyOrigSystem and PartyOrigSystemReference.</td>
</tr>
<tr>
<td>ParentPartyId</td>
<td>The unique internal ID for the existing party record that is the parent node of this node.</td>
<td>You can obtain the Parent Party ID for an existing party by exporting the object using the Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>Conditionally required.</td>
<td>If you're creating a new node, then identify the parent node by providing either ParentPartyId or both PartyOrigSystem and PartyOrigSystemReference.</td>
</tr>
<tr>
<td>ParentPartyOrigSystem</td>
<td>The source system code that identifies the node of this node in your legacy or external system. You provide the source system reference (ParentPartyOrigSystemReference) and the source system code (ParentPartyOrigSystem) when the source of your data is your legacy or external system. The import process uses the source system code and source system reference values to: * find the Oracle Sales Cloud internal ID (ParentPartyId) that uniquely identifies the existing parent node (party record) that was originally imported with the source system reference and source system code values in a prior batch. * find the Oracle Sales Cloud internal ID (ParentPartyId) for the new parent node with the source system reference and source system code values that is imported in the same batch as this node.</td>
<td>Predefine your source system code as enabled.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Party and Customer Hierarchy Member Records in the Same Batch</td>
<td>Creating Customer Hierarchy Member Record for an Existing Party</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
<td>-------------------------</td>
<td>-------------------------------------------------</td>
<td>-------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td>original source system of the party that is the parent node of this node. You provide the source system reference (ParentPartyOrigSystemReference) and the source system code (ParentPartyOrigSystem) when the source of your data is your legacy or external system. The import process uses the source system code and source system reference values to: * find the Oracle Sales Cloud internal ID (ParentPartyId) that uniquely identifies the existing parent node (party record) that was originally imported with the source system reference and source system code values in a prior batch. * find the Oracle Sales Cloud internal ID (ParentPartyId) for the new parent node with the source system reference and source system code values that is imported in the same batch as this node.</td>
<td>for parent parties, using the Setup and Maintenance, Manage Trading Community Source Systems task.</td>
<td>If you’re creating a new node, then identify the parent node by providing either ParentPartyId or both ParPartyOrigSystem and ParPartyOrigSystemReference.</td>
<td>If you’re creating a new node, then identify the parent node by providing either ParentPartyId or both ParPartyOrigSystem and ParPartyOrigSystemReference.</td>
</tr>
<tr>
<td>ParentPartyOrigSystemReference</td>
<td>The ID that identifies the party that is the parent node of this node in your legacy or external system. You provide the source system reference (ParentPartyOrigSystemReference) and the source system code (ParentPartyOrigSystem) when the source of your data is your legacy or external system. The import process uses the source system code and source system reference values to: * find the Oracle Sales Cloud internal ID (ParentPartyId) that uniquely identifies the existing parent node (party record) that was originally imported with the source system reference and source system code values in a prior batch. * find the Oracle Sales Cloud internal ID (ParentPartyId) for the new parent node with the source system reference and source system code values that is imported in the same batch as this node.</td>
<td>Identify the reference value from your source system that was used when you imported the parent party in a prior batch, or identify the source system reference you will use when importing the parent party in the same batch.</td>
<td>Conditionally required.</td>
<td>Conditionally required.</td>
</tr>
</tbody>
</table>

Conditionally required.
Using Reference Files to Evaluate Attributes

For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

Additional Tips

Tip: You can explore related topics by entering the following keywords in the Oracle Fusion Help Application search. The Help Application is available from any Oracle Fusion Application page by clicking Help in the top-right corner, and selecting Applications Help.

- Importing customer hierarchies.
- Importing customer hierarchy members.

Tip: If your data is not from a source system, then you can emulate a system by defining a source system and manually generating reference values for your data. For example, if you periodically gather information at sales events or through sales calls, then you might define a source system, using a code of SALES_ACTIVITY. The format of your reference numbers might concatenate the date, salesperson name, and sequence number, such as 01_Jan_2014_Jackson_01.

Related Topics

- Importing Customer Hierarchies Using File-Based Import: Quick Start

Deleting Sales Account Hierarchies Using File-Based Data Import: Procedure

This procedure describes how to delete the Sales Account Hierarchies using file-based data import. Deleting a sales account hierarchy involves two steps:

- Deleting a hierarchy version
- Deleting a hierarchy member
Deleting a Hierarchy Version

You can use the following steps to delete a hierarchy version:

1. On the Setup and Maintenance work area, search for the Manage File Import Activities task.
2. On the Manage Import Activities page, click Create.
3. Structure the CSV file as per the information in below table:
   - Following table displays the required columns, their values and any comments, if any:

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CustomerHierarchy_ HierarchyType</td>
<td>HZ_CUSTOMER_HIERARCHY</td>
</tr>
<tr>
<td></td>
<td>This field is required for deletion.</td>
</tr>
<tr>
<td>CustomerHierarchy_ HierarchyCode</td>
<td>FIOrgTree_BJUN9_2014</td>
</tr>
<tr>
<td>CustomerHierarchy_ HierarchyName</td>
<td>FIOrgTree_BJUN9_2014</td>
</tr>
<tr>
<td>CustomerHierarchy_ AsOfDate</td>
<td>7/25/2014</td>
</tr>
<tr>
<td></td>
<td>Enter current date.</td>
</tr>
<tr>
<td>CustomerHierarchy_ ActionCode</td>
<td>DELETE</td>
</tr>
<tr>
<td></td>
<td>The ActionCode column is used to indicate that a record is deleted during import. Set the ActionCode column to a value of DELETE for any hierarchy version that you want to delete.</td>
</tr>
<tr>
<td>CustsommerHierarchy_ HierarchyVersionName</td>
<td>FIOrgTree_BJUN9_2014</td>
</tr>
</tbody>
</table>

4. Select CustomerHierarchy object from the drop-down list. Provide a name for the Import Activity in the Name text box. Select the File Type Text File. Browse to select the CSV file that you created.
5. Select the check box Header row included and click Next to go to the Map Fields page.
6. Map the fields to the corresponding attributes.
7. Click Next and activate the import process on the Review and Activate page.

Deleting a Hierarchy Member

You can use the following steps to delete a hierarchy member:

1. Perform the steps 1 and 2 as outlined above.
2. Structure the CSV file as per information in below table:

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CustomerHierarchyMember_ HierarchyType</td>
<td>HZ_CUSTOMER_HIERARCHY</td>
</tr>
</tbody>
</table>

<p>| CustomerHierarchyMember_ HierarchyCode | FIOrgTree_BJUN9_2014 |</p>
<table>
<thead>
<tr>
<th>Column Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CustomerHierarchyMember_AsOfDate</td>
<td>25/7/2014</td>
</tr>
<tr>
<td>CustomerHierarchyMember_ActionCode</td>
<td>DELETE</td>
</tr>
<tr>
<td></td>
<td>The ActionCode column is used to indicate that a record is deleted during import. Set the ActionCode column to a value of DELETE for any hierarchy members you want to delete.</td>
</tr>
<tr>
<td>CustomerHierarchyMember_PartyNumber</td>
<td>CDRM_9153</td>
</tr>
<tr>
<td></td>
<td>Either party number, party orig system or party orig system reference should be provided.</td>
</tr>
<tr>
<td>CustomerHierarchyMember_PartyOrigSystem</td>
<td>DNB</td>
</tr>
<tr>
<td>CustomerHierarchyMember_PartyOrigSystemReference</td>
<td>DNB_6_9_2014_A0</td>
</tr>
</tbody>
</table>

3. Select CustomerHierarchy member object from the drop-down list. Provide a name for the Import Activity in the Name text box. Select the File Type Text File. Browse to select the CSV file that you created.

4. Select the check box Header row included and click Next to go to the Map Fields page.

5. Map the fields to the corresponding attributes, as per below screenshot.
Map Fields

Constraints for Deletion

The deletion process has the following constraints:

- You cannot delete a root node within a hierarchy.
- On deleting a node, its child nodes would be reassigned to its parent node.
Related Topics

- Deleting Records Using File-Based Data Import: Explained
9 Importing Geographies

Importing Geographies Using File-Based Import: Explained

This topic explains how to prepare and import geography data from an external data source using the File-Based Data Import feature. A geography is any region with a boundary around it, regardless of its size. It might be a state, a country, a city, a county, or a ward. You must create or import geographies before you can associate them with custom zones and addresses.

Note: The application ships with third-party (Nokia) master geography data for multiple countries that can be easily imported. You can import Oracle-licensed Nokia data from Navteq, for those countries where the data is available, such as the U.S. You can import Nokia Geography data using the Manage Geographies task. Search for the country, and select Import Nokia Data from the Actions menu. If the licensed Navteq data is not available for a particular country, then the Import Nokia Data action is disabled. For more information, see Replacing Existing Master Geography Data with Revised Nokia Geography Data: Procedure. If Nokia geography data is not available for a country, then use the information in this chapter to import it using File-Based Data Import.

Consider the following questions when importing your data:

- How does your legacy system or source system represent the geography data compared to how Oracle applications represent the same data?
- Do you have to configure values in the application to map to your data values?
- What import features are available for importing your business object?
- How do you verify your imported data?

Comparing Business Object Structures

You must understand how your geography data corresponds with the data in the application so that you can map your legacy data to the data that the application requires. First, you must understand how the application represents the structure of the data for a geography.

You must import a separate country structure import object for each country. Each of these import objects must contain the geography types that are used in the country's structure, organized in a hierarchy using geography level numbers. For example, if you are importing the country structure of Australia, the country structure could be the following: 1: Country, 2: State, 3: County, 4: Town, 5: ZIP.

Import Objects for the Geography

To facilitate importing geographies, the application incorporates the structure of the geography into import objects. The import object for the geography is ImpGeography.

Comparing Business Object Data

Each import object is a collection of attributes that helps to map your data to the application data and to support one-to-many relationships between the structural components that make up the geography.

You must understand the attribute details of the import objects so that you can prepare your import data. You can use reference guide files that contain attribute descriptions, values that populate attributes by default when you do not provide values, and validation information for each import object attribute. The validation information includes the navigation path to...
the task where you can define values in the application. For example, if you have values in your data that correlate to a choice list in the application, then the validation information for that attribute provides the task name in the Setup and Maintenance work area where you can define your values. For additional information, including a list of reference guide file names and locations that you need to complete this task, see the following table.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Related Import Object Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>ImpGeography</td>
<td>Geography Import Objects: How They Work Together</td>
</tr>
</tbody>
</table>

Note:
You can use the keyword importing geographies to search for related topics in Help.

Extensible Attributes
The application doesn’t support extensible attributes for geographies. You can import only data for geography object that already exist by default in the application.

Importing Geographies Using File-Based Data Import
For the geography business object, you must use the File-Based Data Import feature. You prepare XML or text source data files in a form that is suitable for a file-based import. The file-based import process reads the data in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables.

The Define File-Based Data Import Setup and Maintenance task list includes the tasks that are required to configure the import objects, to create source-file mappings, and to schedule the import activities. You submit file-based import activities for each import object. When you’re creating a new geography, you import the Geography object. You must be assigned the Master Data Management Administrator job role to access and submit the import activities for geographies.

When importing geography information, you must provide the parent reference information for all parent levels for the entity.

Verifying Your Imported Data
Oracle applications provide File-Based Import activity reports, which you can use to verify imported data. Users with the Master Data Management Administrator job role can also navigate to the Manage Geographies work area to view the imported geographies.

Related Topics
- File-Based Data Import: How It Works
- Getting Started with File-Based Import: Documentation Overview
- Geography Import Objects: How They Work Together

Geography Import Objects: How They Work Together
This topic describes the Geography import object. You use the Geography import object to import geography information.
This topic introduces the following:

- Target objects for the Geography import object
- Target import object attributes
- Reference guide files for target import object attributes

**Geography Target Import Objects**

You can use the Geography import object to import geography hierarchy information to create or update the geography data of a country. To map the source data in your import file to the target attributes in the application, you must understand how the target objects are related and what attributes are included in each target object.

The target import objects in the Geography import object contain information about the geography hierarchy. When updating an existing geography, you must provide the parent reference information of the existing geography, which connects the geography to the country of which it is a part.

Use the ImpGeography target import object to create and update geography information.

**Note:** Before you import geography data for a country, you must define the country’s geography structure.

**Target Import Object Attributes**

You must compare the attributes that you want to import with the target object attributes that are available and with their valid values. To evaluate your source data and Oracle Sales Cloud attributes for mapping and validation, you use a reference file. See the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. A reference guide file includes attribute descriptions, default values, and validations performed by the import process. Review the validation for each attribute to determine whether there are functional prerequisites or prerequisite setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can predefine and manage import mappings using the Manage File Import Mappings task, or you can define the mapping when you define the import activity using the Manage File Import Activities task. Both tasks are available in the Setup and Maintenance work area.

**Note:** If any of the attributes you want to import do not have an equivalent target object attribute, then review the Application Composer extensibility features for geography.

**Reference Files for Target Import Object Attributes**

To access the reference guide files for the geography’s target import objects, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. For detailed information on importing geographies using file-based import, refer to Document No. 1481758.1, Importing Master Reference Geography Data, on the Oracle Support site.

The following table lists the reference files that are available by target import object.
Importing Geographies Using File-Based Data Import: Worked Example

This example demonstrates how to import data using the File-Based Data Import tool. In this example, you have a source file containing geography data that you want to import into the application so that the geography data can be used for real time address validation and tax purposes.

The following table summarizes the key decisions that you must make in this scenario.

<table>
<thead>
<tr>
<th>Decisions to Consider</th>
<th>In This Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>What type of object are you importing?</td>
<td>Geography</td>
</tr>
<tr>
<td>What file type are you using for your source data?</td>
<td>Text file</td>
</tr>
<tr>
<td>Where are you uploading your source data file from?</td>
<td>Your desktop</td>
</tr>
<tr>
<td>What data type is your source data file?</td>
<td>Comma separated</td>
</tr>
<tr>
<td>Which fields are you importing into the application?</td>
<td>All, except for the RecordTypeCode field</td>
</tr>
<tr>
<td>When do you want to process the import?</td>
<td>Immediately</td>
</tr>
</tbody>
</table>

Summary of the Tasks

You perform the following steps to create an import activity and activate the import:

1. Determining what information is in the source file.
2. Creating and scheduling the import activity.
3. Monitoring the import results.
Prerequisites for Importing Additional Geography Data After Your Initial Import

1. Ensure that the combination of the Source ID and Parent Source ID values is unique for each row of data within a single import. However, your source data files don’t need to have the same Source ID and Parent Source ID values as your previously imported geography data. If the geography structure levels and the parents for each geography value are the same, then the changed IDs will not affect the import.

2. Ensure that all the parents of a child geography are included in your data file so that the child geography can be added. For example, if you originally imported US, CA, and San Francisco, and now you want to import the city of San Jose in CA, then your data file must include US, CA, and San Jose.

3. Check that your source data file has the correct values for the geography data that you have already loaded. For example, if your initial import included the value US for country and CA as state, and in a subsequent import you have California as a state, then your geography import creates two state records (CA and California) in the application data, with the US as the country parent.

Determining What Information is in the Source File

1. The source geography data files must include a unique Source ID value for each row of data and Parent Source ID value for the parent of that row of data. The Source or Parent Source IDs should not be longer than 18 characters.

2. You can structure your geography source data as follows:

<table>
<thead>
<tr>
<th>Geography Level</th>
<th>Name</th>
<th>Source ID</th>
<th>Parent Source ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 (Country)</td>
<td>US</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2 (State)</td>
<td>CA</td>
<td>11</td>
<td>1</td>
</tr>
<tr>
<td>3 (County)</td>
<td>Alameda</td>
<td>111</td>
<td>11</td>
</tr>
<tr>
<td>4 (City)</td>
<td>Pleasanton</td>
<td>1111</td>
<td>111</td>
</tr>
<tr>
<td>4 (City)</td>
<td>Dublin</td>
<td>1112</td>
<td>111</td>
</tr>
</tbody>
</table>

Creating and Scheduling the Import Activity

You can create an import activity, enter the import details, and schedule the import. An import activity includes selecting the source file or file location, mapping the source file to the database, and scheduling the import.

1. In the Setup and Maintenance work area, search for the Manage File Import Activities task. Click Go to Task.
2. In the Manage Import Activities page, click Create.
3. In the Create Import Activity: Map Fields page, map each field from your source file to the target object and attribute, as shown in the following table.
## Importing Geographies

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Master Reference Geographies</td>
</tr>
<tr>
<td>Object</td>
<td>Geography</td>
</tr>
<tr>
<td>File Type</td>
<td>Text File</td>
</tr>
<tr>
<td>File Selection</td>
<td>Specific file</td>
</tr>
<tr>
<td>Upload From</td>
<td>Desktop</td>
</tr>
<tr>
<td>File Name</td>
<td>Choose relevant file from desktop</td>
</tr>
<tr>
<td>Data Type</td>
<td>Comma separated</td>
</tr>
</tbody>
</table>

> **Note:** Ensure that the file type that you select in the Create Import Activity: Set Up page matches the file type of the source data file.

4. Click **Next**.

5. In the Create Import Activity: Map Fields page, map each field from your source file to the Oracle Sales Cloud database object and attribute, as shown in the following table.

<table>
<thead>
<tr>
<th>Column Header</th>
<th>Example Value</th>
<th>Ignore</th>
<th>Object</th>
<th>Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>Primary Geography Name</td>
<td>Primary Geography Name</td>
<td>United States</td>
<td>Imp Geography</td>
<td>Primary Geography Name</td>
</tr>
<tr>
<td>Country Code</td>
<td>US</td>
<td>No</td>
<td>Imp Geography</td>
<td>Country Code</td>
</tr>
<tr>
<td>Record Type Code</td>
<td>0</td>
<td>Yes</td>
<td>Imp Geography</td>
<td>Record Type Code</td>
</tr>
<tr>
<td>Source ID</td>
<td>10265</td>
<td>No</td>
<td>Imp Geography</td>
<td>Source ID</td>
</tr>
<tr>
<td>Parent Source ID</td>
<td>1053</td>
<td>No</td>
<td>Imp Geography</td>
<td>Parent Source ID</td>
</tr>
</tbody>
</table>

If you don’t want to import a column in the text file, then you can select **Ignore**.

> **Note:** If you can’t map the fields from your source file to the relevant target object, then see the import object spreadsheets.

6. Click **Next**.

7. In the Create Import Activity: Create Schedule page, select **Immediate** in the Schedule field so that the import will start as soon as you activate it.
Instead of immediately importing the data, you can choose a date and time to start the import. You can also specify whether the import will be repeated and the frequency of the repeated import.

8. Click Next.

Monitoring the Import Results

You can monitor the processing of the import activity and view the completion reports for both successful records and errors.

1. In the Create Import Activity: Review and Activate page, verify your import details in the Import Details, File Details, Import Options, and Schedule sections. Update the import details if required by navigating to the previous screens using the Back link.
2. Confirm your import details, and click Activate to submit the import.
   After the import activity has finished, the Status field value changes to Completed.

Related Topics

- File-Based Import Processing: How it Works
- Importing Geographies: Explained

Importing Country Structures Using File-Based Import: Quick Start

This topic describes how to get you started with importing country structures. It includes the following information:

- An example of how to identify and associate records
- The minimum data required and the prerequisite tasks
- How to access and use reference files to evaluate attributes
- Additional tips

With the country structure of a country you can define which geography types are part of the country, and how the geography types are hierarchically related within the country. For example, you can create geography types called State, City, and Postal Code. Then you can rank the State geography type as the highest level within the country, the City as the second level, and the Postal Code as the lowest level within the country structure.

The following table lists the file-based import objects and target objects that you can use to import country structures.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Target Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Country Structure</td>
<td>ImpGeoStructureLevel</td>
</tr>
</tbody>
</table>

Identifying and Associating Records with Each Other

To add country structures to a country, or to update existing country structure values using file-based import, your source file must contain information about the country to which the country structure is being assigned. Also, you need to specify the
level number at which the geography types you are importing need to be placed. All countries are, by default, placed at Level 1. Create the country structure by increasing the level number as you go down the country structure. Thus, geography types such as states, provinces, and so on, can be placed at Level 2; districts or counties at Level 3, and so on.

To add an additional identifier to an existing party or to update an existing additional identifier record, your source file must include values that enable the import process to identify the existing records. These values will be either source system and source system reference value combination, Oracle Sales Cloud internal ID, or public unique identifiers, such as business keys or external IDs.

If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system's record in your source file. The file-based data import process stores a cross-reference between the source system information and Oracle Sales Cloud internal ID. The import process uses the combination of source system and source system reference value to identify the existing record.

Minimum Data Required and the Prerequisite Setup Tasks

The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create or update a country structure record. You can optionally include attributes that are available for import in your import file but that aren't listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Country Structure Record for an Existing Country</th>
</tr>
</thead>
<tbody>
<tr>
<td>CountryCode</td>
<td>The code representing the country to which the country structure belongs.</td>
<td>Identify country codes using the Setup and Maintenance, Manage Geographies task.</td>
<td>Required</td>
</tr>
<tr>
<td></td>
<td>Oracle Sales Cloud uses two-letter ISO country codes to refer to countries.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GeographyType</td>
<td>A code used for internal reference by Oracle Sales Cloud at an administrative level. For example, this administrative code may be COUNTRY, STATE, COUNTY, CITY, etc. You must include either the LevelNumber or GeographyType field to define the hierarchical structure of your data.</td>
<td>No prerequisite tasks</td>
<td>Required</td>
</tr>
<tr>
<td>LevelNumber</td>
<td>The level number for the geography type in the hierarchy. For example, a country has the level of 1 in the hierarchy because it is the top level.</td>
<td>No prerequisite tasks</td>
<td>Required</td>
</tr>
</tbody>
</table>
The following image depicts a sample attribute mapping to import a country structure.

Using Reference Files to Evaluate Attributes
For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

Additional Tips
Tip: You can explore related topics by entering the following keywords in Oracle Sales Cloud Help Application search.

- Importing country structures

Tip: For sample country structures and detailed steps for importing country structures, see article 1341174.1 on My Oracle Support.

Related Topics
- Importing Country Structures Using File-Based Import: Explained
- Country Structure Import Objects: How They Work Together
- Geography Structure, Hierarchy, and Validation: How They Fit Together

Importing Geography Hierarchies Using File-Based Import: Quick Start
This topic describes how to get you started with importing geography hierarchies. It includes the following information:

- An overview of geography hierarchies
• An example of how to identify and associate records
• The minimum data required and the prerequisite tasks
• How to access and use reference files to evaluate attributes
• Additional tips

The following table lists the file-based import objects and target objects that you can use to import geography hierarchies.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Target Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Geography Hierarchies</td>
<td>ImpGeography</td>
</tr>
</tbody>
</table>

**Geography Hierarchies Overview**

Geography hierarchy is a data model that establishes conceptual parent-child relationships between geographies. A geography, such as Tokyo or Peru, describes a boundary on the surface of the earth. The application can extrapolate information based on this network of hierarchical geographical relationships. For example, in the Master geography hierarchy, the state of California is defined as the parent of San Mateo county, which is the parent of Redwood City, which is the parent of the postal code 94065. If you enter just 94065, the application can determine that the postal code is in California, or that the corresponding city is Redwood City.

**Identifying and Associating Records with Each Other**

Use the source ID and parent source ID of a geography to identify the geography's parent-child relationship with records within the source file data. The ID can also be used to identify relationships with subsequent interface table data.

For example, when importing a country and its province, you may provide the following values:

• Data for a country: SOURCE_ID = 1; PARENT_ID = (NULL). The PARENT_ID is null because the country is the top level of the geography hierarchy and has no parent.

• Data for a province: SOURCE_ID = 2; PARENT_ID = 1 (the source ID of the country, above.)

**Minimum Data Required and the Prerequisite Setup Tasks**

The minimum data that is required to import address information depends on the following:

• The purpose of the import. The data requirements are different when you're creating both the party and the address in the same import batch, adding addresses to an existing party, or updating an address record.

• Identifying and associating records. In some cases, you can choose which attributes you want to provide to the import process when identifying and associating records.

The values that you provide in your source file may require a setup task or manual step to ensure the information passes the import validation. Before you prepare your source file, complete the prerequisite steps to determine the data values, and then include those values in your source file.

The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create or update a geography hierarchy data. You can optionally include attributes that are available for import in your import file but that aren't listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Geography Hierarchies</th>
<th>Updating Geography Hierarchy Records</th>
</tr>
</thead>
<tbody>
<tr>
<td>CountryCode</td>
<td>The unique code of the country to which the geography belongs.</td>
<td>Identify country codes using the Setup and Maintenance, Manage Geographies task.</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>LevelNumber</td>
<td>The level number of the geography in the hierarchy of geography types for a country where level 1 is the country.</td>
<td>No prerequisite tasks.</td>
<td>Required if you're not specifying the geography type.</td>
<td>Required if you're not specifying the geography type.</td>
</tr>
<tr>
<td>GeographyType</td>
<td>The geography type, such as CITY, COUNTY, and PROVINCE.</td>
<td>Identify geography types by exporting the Geography Type object using the Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task or using the Setup and Maintenance, Manage Geographies task.</td>
<td>Required if you're not specifying the level number.</td>
<td>Required if you're not specifying the level number.</td>
</tr>
<tr>
<td>ParentSourceId</td>
<td>The parent source ID of the geography, along with the source ID, is used to identify the geography’s parent/child relationship with records within the source file data. The ID can also be used to identify relationships with subsequent interface table data.</td>
<td>No prerequisite tasks.</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>SourceId</td>
<td>The source ID for the geography, along with the parent source ID, is used to identify the geography’s parent/child relationship within the source file data and subsequent interface table data.</td>
<td>No prerequisite tasks.</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>PrimaryGeographyName</td>
<td>The name of the primary geography.</td>
<td>No prerequisite tasks.</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>RecordTypeCode</td>
<td>The record type code that represents the intent of the import data.</td>
<td>No prerequisite tasks.</td>
<td>Required</td>
<td>Required</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Geography Hierarchies</td>
<td>Updating Geography Hierarchy Records</td>
</tr>
<tr>
<td>------------------------</td>
<td>-------------</td>
<td>--------------------------</td>
<td>---------------------------------</td>
<td>--------------------------------------</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Geography Hierarchies</td>
<td>Updating Geography Hierarchy Records</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Setup Task</td>
<td>Creating Geography Hierarchies</td>
<td>Updating Geography Hierarchy Records</td>
</tr>
</tbody>
</table>

The values are:

- 0: Indicates a geography already exists in the base table.
- 1: Indicates the intent to create a new geography.
- 2: Indicates the intent to add an alternate identifying name or code to an existing geography. If you choose this record type code, then you must specify the additional fields required.

**Using Reference Files to Evaluate Attributes**

For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.
Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

**Additional Tips**

Tip: You can explore related topics by entering the following keywords in Oracle Sales Cloud Help Application search. You can access help from any Oracle Sales Cloud page by clicking Help in the top-right corner, and selecting Applications Help.

- Importing geographies
- Importing country structures

Tip: If your data isn’t from a source system, then you can emulate a system by defining a source system and manually generating reference values for your data. For example, if you periodically gather information at sales events or through sales calls, then you might define a source system, using a code of SALES_ACTIVITY. The format of your reference numbers might concatenate the date, salesperson name, and sequence number, such as 01_Jan_2014_Jackson_01.

**Related Topics**

- Importing Geographies Using File-Based Import: Explained
- Geography Import Objects: How They Work Together

**Importing Territory Geographies Using File-Based Data Import: Explained**

This topic describes the tasks that you must perform to import territory geography information. A territory is defined as the jurisdiction of responsibility of a salesperson over a set of sales accounts. A territory can be based on various factors called dimensions, such as the geography, product, organization type, industry, business size, account, account type, and sales channel dimension. The territory geography defines the boundaries of a territory according to the selected dimensions, such as Europe or Asia. Dimensions are attributes that define jurisdictional boundaries of territories. The geography dimension, which you can use to define territories by country or postal code, is based on Territory Geographies. Therefore, you must create or import territory geographies before they can be used in territory definitions.

You must create or import geographies, also referred to as Master Geographies, before you can associate them with custom zones and addresses. You must also set up master geographies and create territory geography zone types used in the import, before you can create and import territory geographies.

Importing of the Territory geographies is an incremental import, which either creates a new territory geography hierarchy in the case of the fresh environment or adds branches to the existing territory geographies.

**Comparing Business Object Structures**

You must understand how your territory geography data corresponds with the data in Oracle Sales Cloud to map your legacy data to the data needed by Oracle Sales Cloud. First, you must understand how Oracle Sales Cloud represents the structure of the data for territory geography.

The territory geographies are a combination of user-defined zones and master geography data (also known as Master). If you use only master geographies to define your territory geography, then your territory definition is based only on the elements...
seen in the master geography. For example, in this case, the territory geography is based only on country, state, and ZIP
code.

<table>
<thead>
<tr>
<th>Territory Geographies</th>
<th>Territory</th>
<th>Geography</th>
</tr>
</thead>
<tbody>
<tr>
<td>US</td>
<td>Master</td>
<td>North America</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US, Canada</td>
</tr>
<tr>
<td>PA</td>
<td>Master</td>
<td>US</td>
</tr>
<tr>
<td>01243</td>
<td>Master</td>
<td>East US</td>
</tr>
<tr>
<td>01244</td>
<td>Master</td>
<td>Mid Atlantic</td>
</tr>
<tr>
<td>...</td>
<td>Master</td>
<td>PA</td>
</tr>
<tr>
<td>NY</td>
<td>Master</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td>Master</td>
<td></td>
</tr>
<tr>
<td>Canada</td>
<td>Master</td>
<td></td>
</tr>
</tbody>
</table>

If you use both zones and master geographies to define your territory geography dimension, then you have more flexibility. You can create Masters under zones (for example, US is under North America), and zones can be placed under Masters (for example, East Zone is under US).

<table>
<thead>
<tr>
<th>Territory Geographies</th>
<th>Territory</th>
<th>Geography</th>
</tr>
</thead>
<tbody>
<tr>
<td>North America</td>
<td>Zone</td>
<td>North America</td>
</tr>
<tr>
<td>US</td>
<td>Zone</td>
<td>US</td>
</tr>
<tr>
<td>East Zone</td>
<td>Zone</td>
<td>East</td>
</tr>
<tr>
<td>Mid Atlantic</td>
<td>Zone</td>
<td>Mid Atlantic</td>
</tr>
<tr>
<td>PA</td>
<td>Master</td>
<td>PA</td>
</tr>
<tr>
<td>01243</td>
<td>Master</td>
<td></td>
</tr>
<tr>
<td>01244</td>
<td>Master</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td>Master</td>
<td></td>
</tr>
<tr>
<td>Western NY</td>
<td>Master</td>
<td></td>
</tr>
<tr>
<td>Canada</td>
<td>Master</td>
<td></td>
</tr>
</tbody>
</table>

Import Objects for the Territory Geography

To facilitate the import of territory geographies, Oracle Sales Cloud incorporates the structure of the territory geography into import objects. The import object for the territory geography is ImpGeoTerritory.
Comparing Business Object Data

Each import object is a collection of attributes that helps to map your data to the Oracle Sales Cloud data and to support one-to-many relationships between the structural components that make up the territory geography.

You must understand the attribute details of the import objects so that you can prepare your import data. You can use reference guide files that contain attribute descriptions, and validation information for each attribute. The validation information includes the navigation path to the task where you can define values in Oracle Sales Cloud. For example, if you have values in your data that correlate to a choice list in Oracle Sales Cloud, then the validation information for that attribute provides the task name in the Setup and Maintenance work area where you can define your values. For additional information, including a list of reference guide file names and locations that you require to complete this task, see the topic listed in the following table.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Related Import Object Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>ImpGeoTerritory</td>
<td>Territory Geography Import Objects: How They Work Together</td>
</tr>
</tbody>
</table>

Note: You can use the keyword importing territory geographies to search for related topics in Help.

Extensible Attributes

Oracle Sales Cloud does not support extensible attributes for territory geographies. You can import data only for attributes provided by Oracle Sales Cloud.

Importing Territory Geographies Using File-Based Data Import

For the territory geography business object, you must use the File-Based Data Import feature. You prepare source data file in a form of CSV (comma-separated values). The file-based import process reads the data in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables. For more information about the list of territory geography attributes and its descriptions, see the "Importing Territory Geography Hierarchies Using File-Based Data Import: Quick Start" topic.

The Define File-Based Data Import Setup and Maintenance task list includes the tasks required to configure the import objects, to create source-file mappings, and to schedule the import activities. You submit file-based import activities for each import object. When you're creating a new territory geography, you import the Territory Geography object. You must be assigned the Master Data Management Administrator job role to access and submit the import activities for geographies.

When importing territory geography information, you must provide the complete territory geography hierarchy.

Verifying Your Imported Data

Oracle Sales Cloud provides File-Based Import activity reports, which you can use to verify imported data. Users with the Master Data Management Administrator job role can also navigate to the Manage Territory Geographies work area to view the imported territory geographies.
Territory Geography Import Objects: How They Work Together

This topic describes the territory geography import object. You use the territory geography import object to import territory geography information.

This topic introduces the following:

- Target objects for the territory geography import object
- Target import object attributes
- Reference guide files for target import object attributes

Note: Importing of the Territory geographies is an incremental import, which either creates a new territory geography hierarchy in the case of the fresh environment or adds branches to the existing territory geographies.

Territory Geography Target Import Objects

The territory geography import object is used to import territory geography hierarchy information to create branches of the territory geography hierarchy. To map the source data in your import file to the target attributes in Oracle Sales Cloud, you must understand how the target objects are related and what attributes are included in each target object.

The target import objects in the territory geography import object contain information about the territory geography hierarchy. When updating existing territory geography, you must provide the parent reference information of the existing territory geography, which connects the territory geography with the top node that it is a part of.

Use the ImpGeoTerritory target import object to create and update the territory geography information.

Note: Before you import the territory geography data, you must set up master geographies and create territory geography zones.

Target Import Object Attributes

You must compare the attributes that you want to import with the target object attributes that are available and with their valid values. To evaluate your source data and Oracle Sales Cloud attributes for mapping and validation, you use a reference file. See the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. A reference guide file includes attribute descriptions, default values, and validations performed by the import process. Review the validation for each attribute to determine whether there are functional prerequisites or prerequisite setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can predefine and manage import mappings using the Manage File Import Mappings task, or you can define the mapping when you define the import activity using the Manage File Import Activities task. Both tasks are available in the Setup and Maintenance work area.
Importing Territory Geography Hierarchies Using File-Based Data Import: Quick Start

This topic describes a few key concepts and provides guidance to get you started with importing territory geography hierarchies. It includes the following information:

- An overview of territory geography hierarchies
- An example of how to identify and associate records
- The minimum data required and the prerequisite tasks
- How to access and use reference files to evaluate attributes
- Additional tips

**Note:** Importing of the Territory geographies is an incremental import, which either creates a new territory geography hierarchy in the case of the fresh environment or adds branches to the existing territory geographies.

The following table lists the file-based import objects and target objects that you can use to import territory geography hierarchies.

<table>
<thead>
<tr>
<th>Import Object</th>
<th>Target Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Territory Geography Hierarchies</td>
<td>ImpGeoTerritory</td>
</tr>
</tbody>
</table>

** Territory Geography Hierarchies Overview**

A territory geography hierarchy is a data model that establishes conceptual parent-child relationships between territory geographies. The territory geographies form the basis for the geography dimension in territory management. It defines the boundaries of the territories when used in the territory definitions. A territory geography is a combination of master geography data and user-defined zones and master geography data.

**Identifying and Associating Records with Each Other**

Use the source ID and parent source ID of a territory geography to identify the territory geography’s parent-child relationship with records within the source file data. The ID can also be used to identify relationships with subsequent interface table data.

For example, when importing a country, which is the top node, and its province, you may provide the following values:

- Data for a country: SOURCE_ID = 1; PARENT_ID = (NULL). The PARENT_ID is null because the country is the top level of the territory geography hierarchy and has no parent.
- Data for a province: SOURCE_ID = 2; PARENT_ID = 1 (the source ID of the country, above.)
Minimum Data Required and the Prerequisite Set up Tasks

The minimum data that is required to import territory geography information depends on the following:

- The purpose of the import. The data requirements are different when you’re creating both the master geography and zone in the same import batch.
- Identifying and associating records. In some cases, you can select which attributes you want to provide to the import process when identifying and associating records.

The values that you provide in your source file may require a setup task or manual step to ensure the information passes the import validation. Before you prepare your source file, complete the prerequisite steps to determine the data values, and then include those values in your source file.

The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create or update a territory geography hierarchy data. You can optionally include attributes that are available for import in your import file but that aren’t listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Set up Task</th>
<th>Creating Territory Geography Hierarchy Branches</th>
</tr>
</thead>
<tbody>
<tr>
<td>TerritoryGeographyName</td>
<td>The name of a territory geography.</td>
<td>No prerequisite tasks.</td>
<td>Required</td>
</tr>
<tr>
<td>SourceId</td>
<td>The source ID for the territory geography, along with the parent source ID, is used to identify the territory geography’s parent/child relationship within the source file data and subsequent interface table data.</td>
<td>No prerequisite tasks.</td>
<td>Required</td>
</tr>
<tr>
<td>ParentSourceId</td>
<td>The parent source ID of the territory geography, along with the source ID, is used to identify the territory geography’s parent/child relationship with records within the source file data. The ID can also be used to identify relationships with subsequent interface table data.</td>
<td>No prerequisite tasks.</td>
<td>Required except when the territory geography is a top node.</td>
</tr>
<tr>
<td>LanguageCode</td>
<td>The unique language code of the country to which the territory geography belongs. The default value is US.</td>
<td>No prerequisite tasks.</td>
<td>Required</td>
</tr>
<tr>
<td>TerritoryGeographyType</td>
<td>The territory geography type that could either be a combination of master geography data and user-defined zones. The master geography data includes values such as city, county, and province.</td>
<td>For master geographies, identify the geography type, such as country, state, province, using the Setup and Maintenance, Manage Geographies task.</td>
<td>Required</td>
</tr>
</tbody>
</table>

For zones, identify the existing zone types in the application using the Add Zone option.
## ZoneCode

A unique user-defined code of the zone to which the territory geography belongs. It is used only for zones.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Set up Task</th>
<th>Creating Territory Geography Hierarchy Branches</th>
</tr>
</thead>
<tbody>
<tr>
<td>ZoneCode</td>
<td>A unique user-defined code of the zone to which the territory geography belongs. It is used only for zones.</td>
<td>No prerequisite tasks.</td>
<td>Required</td>
</tr>
<tr>
<td>GeoCountryName</td>
<td>The name of the country for that geography. It is used only for master geographies.</td>
<td>No prerequisite tasks.</td>
<td>Required</td>
</tr>
<tr>
<td>GeoParentName1</td>
<td>Used to specify the complete path of any master geography, starting with the country name. For example, if your TerritoryGeographyName is San Francisco, then the complete path of the master geography is:</td>
<td>Identify the full path of the master geography from the Setup and Maintenance, Manage Geographies task.</td>
<td>Required only for master geography.</td>
</tr>
<tr>
<td>GeoParentName2</td>
<td>Used to specify the complete path of any master geography, starting with the country name. For example, if your TerritoryGeographyName is San Francisco, then the complete path of the master geography is:</td>
<td>Identify the full path of the master geography from the Setup and Maintenance, Manage Geographies task.</td>
<td>Required for master geography, only if the master path is at least two elements long.</td>
</tr>
</tbody>
</table>

**Note:**
- You can create a zone of a specific type in the Manage Territory Geographies task using the Add Zone option. When you delete the zone, the zone type remains in the application, and appears in the Zone Type drop-down list the next time you create a new zone.
Using Customer Data Management

Chapter 9

Importing Geographies

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Set up Task</th>
<th>Creating Territory Geography Hierarchy Branches</th>
</tr>
</thead>
<tbody>
<tr>
<td>GeoParentName2</td>
<td>• GeoParentName2 = &quot;CA&quot;</td>
<td>Identify the full path of the master geography from the Setup and Maintenance, Manage Geographies task.</td>
<td>Required for master geography, only if the master path is at least three elements long.</td>
</tr>
<tr>
<td></td>
<td>• GeoParentName3 = &quot;San Francisco&quot; (county)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• GeoParentName4 = &quot;San Francisco&quot; (city)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• GeoParentName5 = blank</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GeoParentName3</td>
<td>Used to specify the complete path of any master geography, starting with the country name. For example, if your TerritoryGeographyName is San Francisco, then the complete path of the master geography is:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• GeoParentName1 = &quot;United States&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• GeoParentName2 = &quot;CA&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• GeoParentName3 = &quot;San Francisco&quot; (county)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• GeoParentName4 = &quot;San Francisco&quot; (city)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• GeoParentName5 = blank</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GeoParentName4</td>
<td>Used to specify the complete path of any master geography, starting with the country name. For example, if your TerritoryGeographyName is San Francisco, then the complete path of the master geography is:</td>
<td></td>
<td>Required for master geography, only if the master path is at least four elements long.</td>
</tr>
<tr>
<td></td>
<td>• GeoParentName1 = &quot;United States&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• GeoParentName2 = &quot;CA&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• GeoParentName3 = &quot;San Francisco&quot; (county)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• GeoParentName4 = &quot;San Francisco&quot; (city)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• GeoParentName5 = blank</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GeoParentName5</td>
<td>Used to specify the complete path of any master geography, starting with the country name. For example, if your TerritoryGeographyName is San Francisco, then the complete path of the master geography is:</td>
<td></td>
<td>Required for master geography, only if the master path is at least five elements long.</td>
</tr>
<tr>
<td></td>
<td>• GeoParentName1 = &quot;United States&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• GeoParentName2 = &quot;CA&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• GeoParentName3 = &quot;San Francisco&quot; (county)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• GeoParentName4 = &quot;San Francisco&quot; (city)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• GeoParentName5 = blank</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Prerequisite Set up Task</td>
<td>Creating Territory Geography Hierarchy Branches</td>
</tr>
<tr>
<td>--------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------------------------</td>
<td>-----------------------------------------------</td>
</tr>
<tr>
<td>IncludeChildGeoType1</td>
<td>Used to include the specific type of children of a specific master geography.</td>
<td>For master geographies, identify the available children type, such as country, state, province, for a given country from the Setup and Maintenance, Manage Geographies task.</td>
<td>Required only if IncludeChildGeoType2 is specified.</td>
</tr>
<tr>
<td>IncludeChildGeoType2</td>
<td>Used to include the specific type of children of a specific master geography.</td>
<td>For master geographies, identify the available children type, such as country, state, province, for a given country from the Setup and Maintenance, Manage Geographies task.</td>
<td>Required only if IncludeChildGeoType3 is specified.</td>
</tr>
<tr>
<td>IncludeChildGeoType3</td>
<td>Used to include the specific type of children of a specific master geography.</td>
<td>For master geographies, identify the available children type, such as country, state, province, for a given country from the Setup and Maintenance, Manage Geographies task.</td>
<td>Required only if IncludeChildGeoType4 is specified.</td>
</tr>
<tr>
<td>IncludeChildGeoType4</td>
<td>Used to include the specific type of children of a specific master geography.</td>
<td>For master geographies, identify the available children type, such as country, state, province, for a given country from the Setup and Maintenance, Manage Geographies task.</td>
<td>Required only if IncludeChildGeoType5 is specified.</td>
</tr>
<tr>
<td>IncludeChildGeoType5</td>
<td>Used to include the specific type of children of a specific master geography.</td>
<td>For master geographies, identify the available children type, such as country, state, province, for a given country from the Setup and Maintenance, Manage Geographies task.</td>
<td>Required only if you want to specify a child node for the parent.</td>
</tr>
<tr>
<td>StartDate</td>
<td>The territory geography start date.</td>
<td>No prerequisite tasks.</td>
<td>The attribute is neither required nor conditionally required.</td>
</tr>
<tr>
<td>EndDate</td>
<td>The territory geography end date.</td>
<td>No prerequisite tasks.</td>
<td>The attribute is neither required nor conditionally required.</td>
</tr>
</tbody>
</table>

For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

**Additional Tips**

Tip: You can explore related topics by entering the following keywords in the Oracle Sales Cloud Help Application search.

- Importing geographies
10 Importing Households

Importing Common Supporting Objects: Documentation Overview

Common supporting objects often support the primary import objects. For example, additional party identifiers support accounts, contacts, and households. Notes support leads and opportunities. For a mapping of primary import objects to the common object import topics that support them, see:

Related Topics
- Importing Common Supporting Objects: Overview

Importing Households Using File-Based Import: Explained

This topic explains how to prepare and import household data from an external data source into Oracle Sales Cloud using the File-Based Data Import feature.

A household is a party consisting of a collection of persons and organizations. A household, similar to a person or organization, can enter into a business relationship with other parties and can be assigned to opportunities and leads. A household can have members of party type person or organization. You can maintain household information in Oracle Sales Cloud to create leads and opportunities for a household, or capture information about a household to know it better. However, maintaining this information may not let you assign all sales and marketing activities to a household.

Consider the following questions when importing your data:

- How does your legacy or source system represent the household information compared to how Oracle Sales Cloud represents the same data?
- Do you have to configure values in Oracle Sales Cloud to map your existing data values to the Household import object?
- Do you have to customize Oracle Sales Cloud to capture attributes that are critical to the way you do business?
- What import features are available for importing your business object?
- How do I verify my imported data?

Comparing Business Object Structures

You must understand how your household data corresponds with the data in Oracle Sales Cloud so that you can map your legacy data to the data that Oracle Sales Cloud requires. First, you must understand how Oracle Sales Cloud represents the structure of the data for an household.

The Household import object structure is hierarchical. The root of the hierarchy is the group profile, which must exist before you can import lower-level components, such as classifications, contacts, and relationships. These child entities can have other entities as their child entities. This hierarchical structure supports one-to-many relationships between the components that make up the household.
The following figure shows the household import object and its child entities.

The household profile contains basic information about the household, such as the group name, group type, and the group-party usage. For each household, you can assign classifications, members, relationships, additional identifier, and additional names.

Note: All entities referring to contact information, such as primary phone or e-mail, include a child entity that captures the contact preference. For example, the Phone Contact Preference entity captures the contact preference of the household for the phone contact method.

Comparing Business Object Data
Each import object is a collection of attributes that helps to map your data to the Oracle Sales Cloud data and to support one-to-many relationships between the structural components that make up the household.

You must understand the attribute details of the import objects so that you can prepare your import data. You can use reference guide files that contain attribute descriptions, values that populate attributes by default when you do not provide values, and validation information for each attribute. The validation information includes the navigation path to the task where you can define values in Oracle Sales Cloud. For example, if you have values in your data that correlate to a choice list in Oracle Sales Cloud, then the validation information for that attribute provides the task name in the Setup and Maintenance work area where you can define your values.

Note:
You can use the keyword importing households to search for related topics in Oracle Sales Cloud Help.

Extensible Attributes
If you want to extend the Oracle Sales Cloud object to import your legacy or source data, then you must use Application Composer to design your object model extensions and to generate the required artifacts to register your extensions and make them available for importing. The corresponding import object is updated with the extensible attributes, which you
can then map to your source file data. You can use the same source file to import both extensible custom attributes and the standard import object attributes.

Importing Households Using File-Based Data Import

For the household business object, you must use the File-Based Data Import feature. You prepare XML or text source data files in a form that is suitable for a file-based import. The file-based import process reads the data in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables.

The Define File-Based Data Import Setup and Maintenance task list includes the tasks that are required to configure the import objects, to create source-file mappings, and to schedule the import activities. You submit file-based import activities for each import object. When you're creating a new household, you import the household object.

You must be assigned the Customer Relationship Management Application Administrator job role to access and submit the import activities for household.

When importing household information, you first import the basic household information, followed by the child entities for the household. When importing child entities, you must provide the parent reference information for all parent levels for the entity. For example, you first import basic profile details, such as name and household type. You then import contacts and contact information, such as phone, address, contact points, and fax for the household. You must provide the PartyOrigSystem and PartyOrigSystemReference of the household when importing contacts for the household. PartyOrigSystem is the source system code that identifies the source of the information being imported.

Verifying Your Imported Data

Oracle Sales Cloud Applications provide File-Based Import activity reports, which you can use to verify imported data. Alternatively, you can also navigate to the Customer Center work area to view the household information that you have imported.

Related Topics

- Getting Started with File-Based Import: Documentation Overview
- File-Based Data Import: How It Works
- Household Import Objects: How They Work Together
- Extending Oracle Sales Cloud: How It Works

Household Import Objects: How They Work Together

You use the Household import object to import households, their basic information, and members of the household. This topic describes the Household import object and introduces the following:

- Target objects for the Household import object
- Target import object attributes
- Reference guide files for target import object attributes

Household Target Import Objects

The Household import object is split into separate target import objects for organizing the individual attributes for the different aspects of the household. The target import objects in the Household import object are grouped into information about the
household and information about the member. The group profile is the target import object containing attributes to import information about the household. You can have multiple members associated with a household. You can assign only an organization or a person as a member of a household. You cannot assign a household as a member of another household.

When updating an existing household with additional information, you must provide the parent reference information for the existing household. When importing contacts or contact information for a household, you must provide relationship reference information in addition to the parent reference. When importing information about a member, you must refer to the specific relationship that you want to import information for. For example, you might want to import information for John Smith the employee or John Smith the board member. If you do not include the reference information for a relationship, then the import process will create a relationship.

To update the information for an existing household or to create a household record, you can import household profile information, addresses, and contact points, such as phone and fax. The following target import objects are for creating and updating the household information: GroupProfile, Fax, Mobile, Phone, e-mail, InstantMessenger, Relationship, Member, Address, Classification, AdditionalName, and AdditionalIdentifier.

All contact-related entities, such phone or e-mail, include a child entity that captures the contact preference. For example, the Phone Contact Preference entity captures the contact preference of the household for the contact method primary phone. Additionally, the Address import object for a household includes another child entity, AddressPurpose, that captures the purpose of the current household address.

Target Import Objects Attributes

You must compare the attributes that you want to import with the target object attributes that are available and with their valid values. To evaluate your source data and Oracle Sales Cloud attributes for mapping and validation, you use a reference file. See the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects. A reference guide file includes attribute descriptions, default values, and validations performed by the import process. Review the validation for each attribute to determine whether there are functional prerequisites or prerequisite setup tasks that are required.

To import your source file data, you define a mapping between your source file data and the combination of the target object and target object attribute. You can predefine and manage import mappings using the Manage File Import Mappings task, or you can define the mapping when you define the import activity using the Manage File Import Activities task. Both tasks are available in the Setup and Maintenance work area.

*Note:* If any of the attributes you want to import do not have an equivalent target object attribute, then review the Application Composer extensibility features for the marketing response.

Target Import Objects Attributes Resources

To access reference files for this object’s target import objects, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

The following table lists the reference files that are available by target import object.

<table>
<thead>
<tr>
<th>Target Import Object</th>
<th>Description</th>
<th>Reference Guide File Names</th>
</tr>
</thead>
<tbody>
<tr>
<td>GroupProfile</td>
<td>Household information. The default party usage for a household is SALES_PROSPECT.</td>
<td>HZ_IMP_PARTIES_T_Reference</td>
</tr>
<tr>
<td>Target Import Object</td>
<td>Description</td>
<td>Reference Guide File Names</td>
</tr>
<tr>
<td>---------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>---------------------------------------</td>
</tr>
<tr>
<td>Relationship</td>
<td>Relationship between the household and the household member. You must enter a relationship code in the RelationshipCode column, when creating a relationship.</td>
<td>HZ_IMP_RELSHIPS_T</td>
</tr>
<tr>
<td>Member</td>
<td>Household member's information.</td>
<td>HZ_IMP_RELSHIPS_T</td>
</tr>
<tr>
<td>Classification</td>
<td>Household classification. Classification allows you to categorize entities such as parties, projects, tasks, and orders as hierarchies.</td>
<td>HZ_IMP_CLASSIFICS_T</td>
</tr>
<tr>
<td>AdditionalIdentifier</td>
<td>Household's additional identifier information.</td>
<td>HZ_IMP_ADDTNLPARTYID_T</td>
</tr>
<tr>
<td>AdditionalName</td>
<td>Household's alternative name.</td>
<td>HZ_IMP_ADDTNLPARTYNAMES_T</td>
</tr>
<tr>
<td>Url</td>
<td>Household's URL.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>InstantMessenger</td>
<td>Household's instant messenger or social networking information.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>Fax</td>
<td>Household's fax.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>FaxContactPreference</td>
<td>Indicates the household preference about being contacted through fax.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>Mobile</td>
<td>Household's mobile number.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>MobileContactPreference</td>
<td>Indicates the household's preference about being contacted through mobile phone.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>Phone</td>
<td>Primary phone number of the household. If the household has multiple phone numbers, one of the phone numbers is designated as the primary phone number.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>PrimaryPhoneContactPreference</td>
<td>Indicates the household's preference about being contacted through phone.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>Email</td>
<td>Household's e-mail.</td>
<td>HZ_IMP_CONTACTPTS_T_Reference</td>
</tr>
<tr>
<td>EmailContactPreference</td>
<td>Indicates the household's preference about being contacted through e-mail.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>Address</td>
<td>Household's address. If the household has multiple addresses, one of the addresses is designated as the primary address.</td>
<td>HZ_IMP_LOCATIONS_T HZ_IMP_PARTYSITES_T</td>
</tr>
</tbody>
</table>
## Target Import Object

<table>
<thead>
<tr>
<th>Description</th>
<th>Reference Guide File Names</th>
</tr>
</thead>
<tbody>
<tr>
<td>Indicates the household’s preference about being contacted at the primary address.</td>
<td>HZ_IMP_CONTACTPREFS_Reference</td>
</tr>
<tr>
<td>Sample attributes: ContactType, PreferenceCode, PreferenceEndDate, PreferenceStartDate, and ReasonCode.</td>
<td></td>
</tr>
<tr>
<td>Reference attributes: CpOrigSystem, CpOrigSystemReference.</td>
<td></td>
</tr>
<tr>
<td>Indicates the purpose of a household’s address.</td>
<td>HZ_IMP_PARTYSITEUSES_T</td>
</tr>
<tr>
<td>Sample attributes: SiteUseType, EndDate, and StartDate.</td>
<td></td>
</tr>
<tr>
<td>Reference attributes: SiteOrigSystem and SiteOrigSystemReference.</td>
<td></td>
</tr>
</tbody>
</table>

### Related Topics
- Importing Households Using File-Based Import: Explained
- Households: Explained
- Getting Started with File-Based Import: Documentation Overview
- Extending Oracle Sales Cloud: How It Works

## Importing Group Profiles Using File-Based Import: Quick Start

This topic describes how to get you started with importing groups. It includes the following information:

- An example of how to identify and associate records
- The minimum data required and the prerequisite tasks
- How to access and use reference files to evaluate attributes
- Additional tips

Groups allow you to create a collection of contacts and accounts. A group can enter into a business relationship with other types of customers. You can assign member types to group members to define their relationship with the group. You use the Group import object to import groups.

### Identifying and Associating Records with Each Other

To add groups or to update existing group profiles using file-based import, your source file must contain information about the groups.
To update an existing group profile, your source file must include the values that enable the import process to identify the existing records. These values will be source system and source system reference value combination, Oracle Sales Cloud internal ID, or public unique identifiers, such as business keys or external IDs.

If the source of your data is an external system, and if you intend to import updates to previously imported records from the external system, then you can provide the source system code and the unique reference value for the source system’s record in your source file. The file-based data import process stores a cross-reference between the source system information and Oracle Sales Cloud internal ID. The import process uses the combination of source system and source system reference value to identify the existing record.

You can use external IDs or business keys to identify and associate records with each other when you’re importing a smaller set of records. Business keys are a set of attributes that uniquely identify a party record. For example, a person’s last name and first name uniquely identify a person. An external ID is a unique record identifier from a system outside of Oracle Sales Cloud. You can use the following business keys or external IDs in Oracle Sales Cloud:

- Organization name for organizations
- First name and last name for persons
- Contact information, which is a combination of email Id, mobile number, IM, and URL.
- Address information, which is a combination of address1, address2, city, and postal code.

You can configure source systems to identify the source of the data that you’re importing. Source systems are external sources of data that are used to import data into Oracle Sales Cloud. Oracle Sales Cloud uses source system references to create references between source IDs and the Oracle Sales Cloud database IDs. You can configure source systems either by using the Manage Trading Community Source System task, or by importing the source system information into Oracle Sales Cloud using the Source System Reference import object.

When the source of your data isn’t an external system and you don’t intend to regularly update the data, you don’t need the source system information. To import updates to your existing data, you can export the Oracle Sales Cloud internal ID and add it to your source file. The import process uses the internal ID to identify the existing record.

Minimum Data Required and the Prerequisite Steps

The minimum data that is required to import group profile information depends on the following:

- The purpose of the import. The data requirements are different when you’re creating the group profile or updating an existing group profile record.
- Identifying and associating records. In some cases, you can choose which attributes you want to provide to the import process when identifying and associating records.

The values that you provide in your source file may require a setup task or manual step to ensure that the information passes the import validation. Before you prepare your source file, complete the prerequisite steps to determine the data values and then include those values in your source file.

The following table lists the attributes (minimum data) that are required or conditionally required in your import file to create or update a group profile record. You can optionally include attributes that are available for import in your import file but that aren’t listed in the table. For each attribute, the prerequisite setup task before importing that attribute is listed.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Group Profile</th>
<th>Updating Group Profile</th>
</tr>
</thead>
<tbody>
<tr>
<td>PartyOrigSystem</td>
<td>The code representing the external source system.</td>
<td>Predefine your source system code as enabled for Trading Community members, using the Setup and</td>
<td>Conditionally required</td>
<td>Conditionally required</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Provide reference information to identify</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# Importing Households

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Group Profile</th>
<th>Updating Group Profile</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maintenance, Manage</td>
<td>Source Systems task.</td>
<td></td>
<td>the existing party. The reference information can be:</td>
<td>the existing party. The reference information can be:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• PartyId (Oracle Sales Cloud record ID).</td>
<td>• PartyId (Oracle Sales Cloud record ID).</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
<td>• PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Party Number (public unique identifier for an organization profile).</td>
<td>• Party Number (public unique identifier for an organization profile).</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• A business key such as organization name.</td>
<td>• A business key such as organization name.</td>
</tr>
<tr>
<td>PartyOrigSystemReference</td>
<td>The reference number or text representing the source system unique ID for the group.</td>
<td>Conditionally required</td>
<td>Identify the reference value from your source system.</td>
<td>Conditionally required</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• PartyId (Oracle Sales Cloud record ID).</td>
<td>• PartyId (Oracle Sales Cloud record ID).</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
<td>• PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Party Number (public unique identifier for an organization profile).</td>
<td>• Party Number (public unique identifier for an organization profile).</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• A business key such as organization name.</td>
<td>• A business key such as organization name.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>GroupName</th>
<th>The name of the group.</th>
<th>No prerequisite tasks.</th>
<th>Required</th>
<th>The attribute is neither required or conditionally required.</th>
</tr>
</thead>
</table>

---

**Oracle Sales Cloud**

**Using Customer Data Management**

Chapter 10

Importing Households

---

268
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Prerequisite Setup Task</th>
<th>Creating Group Profile</th>
<th>Updating Group Profile</th>
</tr>
</thead>
<tbody>
<tr>
<td>ObjectKey or PartyId</td>
<td>The Oracle Sales Cloud record ID for the group.</td>
<td>Identify the PartyId for an existing group profile by exporting the Group object using the Setup and Maintenance, Manage Bulk Data Export, Schedule Export Processes task.</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>Conditionally required</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Provide reference information to identify the existing party. The reference information can be:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyId (Oracle Sales Cloud record ID).</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• PartyOrigSystem and PartyOrigSystemReference (source system code and source system reference values) provided when importing the organization profile.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• Party Number (public unique identifier for an organization profile).</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• A business key such as organization name.</td>
</tr>
<tr>
<td>GroupType</td>
<td>The type of the group.</td>
<td>Identify valid lookup codes for the HZ_PARTY_GROUP_TYPES lookup using the Setup and Maintenance, Manage Standard Lookups task.</td>
<td>Required</td>
<td>The attribute is neither required nor conditionally required.</td>
</tr>
<tr>
<td>Type</td>
<td>The party type. The type is defaulted to Group for new group profiles.</td>
<td>No prerequisite tasks</td>
<td>The attribute is neither required nor conditionally required.</td>
<td>The attribute is neither required nor conditionally required.</td>
</tr>
</tbody>
</table>
The following figure shows a sample attribute mapping to import a new group profile. In this scenario, the origin system and origin system reference information identify the group profile.

The following figure shows a sample attribute mapping to update an existing group profile. In this scenario, the origin system reference information identifies the group profile and updates its group type.

Using Reference Files to Evaluate Attributes

For information about available import attributes, see the File Based Data Import for Oracle Sales Cloud guide available on the Oracle Sales Cloud Help Center (https://docs.oracle.com/cloud/latest/salescs_gs/docs.htm). In the File Based Data Imports chapter, see the topic for your import object of interest, which includes links to reference files for target import objects.

Review a reference file to see attributes that are available to import and information for each attribute, including requirement for inclusion in an import, validation requirements, default value, data type, data length, and description.

Additional Tips

Tip: You can explore related topics by entering the following keywords in the Oracle Sales Cloud Help Application search.

- Importing groups
- Importing contacts

Tip: If your data isn’t from a source system, then you can emulate a system by defining a source system and manually generating reference values for your data. For example, if you periodically gather information at sales events or through sales calls, then you might define a source system, using a code of SALES_ACTIVITY. The format of your reference values might concatenate the date, salesperson name, and sequence number, such as 01_Jan_2014_Jackson_01.

Related Topics

- Importing Households Using File-Based Import: Explained
• Households: Explained

• Household Import Objects: How They Work Together
11 Importing Source System References

Importing Source System References Using File-Based Import: Explained

This topic explains how to prepare and import source system references from an external data source into Oracle Sales Cloud, using the File-Based Data Import feature. Source systems are external sources of data that are used to import data into Oracle Sales Cloud. Source system references identify the source of the data and specify the references to existing source systems and base tables of Oracle Sales Cloud. Oracle Sales Cloud uses source system references to create references between source system IDs and the Oracle Sales Cloud IDs. For example, if you’re importing account records from a legacy system, then you can import the source system reference to the legacy system. During an import, you can specify the details of the legacy system and also specify the base table in Oracle Sales Cloud.

You can import the first set of Origin System and Origin System Reference for an object, while importing the object. Additionally, you can import other sets of Origin System and Origin System Reference using the Source System Reference import object. You must consider the following questions before importing source system references:

- How does your legacy or source system represent source system information compared with how Oracle Sales Cloud represents the same data?
- Do you have to configure values in Oracle Sales Cloud to map them to your existing data values?
- What import features are available for importing your business object?
- How do I verify my imported data?

Comparing Business Object Structures

You must understand how your source system data corresponds to the data in Oracle Sales Cloud so that you can map your legacy data to the data that Oracle Sales Cloud requires.

In Oracle Sales Cloud, you must import separate source system references for each source system. Each of these source system references must contain the owner table name, origin system, and origin system reference information. Additionally, while importing records into Oracle Sales Cloud, you must specify the existing owner table, origin system, and origin system reference.

Importing Source System References Using File-Based Data Import

For the Source System Reference business object, you must use the File-Based Data Import feature to import data into Oracle Sales Cloud. You prepare XML or text source data files in a form that is suitable for a file-based import. The file-based import process reads the data in your source file, populates the interface tables according to your mapping, and imports the data into the application destination tables.

The Define File-Based Data Import Setup and Maintenance task list includes the tasks that are required to configure the import objects, to create source-file mappings, and to schedule the import activities. You submit file-based import activities for each import object. When you’re creating a classification, you use the Source System Reference import object. An import activity defines the instructions for processing the import data, including the source file, the import mapping from the source file to the object and attribute, and the import schedule.
Verifying Your Imported Data

You can view the list of import activities from the Manage Import Activities page. You can verify your imported data by clicking the Status column for your import activity. Alternatively, you can also navigate to the Manage Trading Community Source Systems work area to view the source system information that you have imported.

Related Topics

- Source Systems: Explained
- Managing Source System References: Explained
About the Oracle Social Data and Insight Cloud service: Overview

You use the Oracle Social Data and Insight Cloud service to enrich your account and contact data in Oracle Sales Cloud. Data enrichment improves the quality of your existing account or account contact data, address information, and also enriches that data with additional information.

You can enrich Sales Cloud data either manually or automatically. Additionally, you can also download enriched account and contact data from the Oracle Social Data and Insight Cloud service. The following figure depicts the manual and automated enrichment process flows:

For more information, see Using Oracle Social Data and Insight Cloud Service guide.
Enriching Oracle Sales Cloud Data with Social Attributes: Process Overview

Enriching Oracle Sales Cloud Data with Social Attributes consist of two set of tasks. They are as follows:

- Set up the social attributes supported by Oracle Social Data and Insight Cloud Service also known as Data as a Service (DaaS) in Oracle Sales Cloud (OSC).
- Enrich the Oracle Sales Cloud account and contact data with the social data downloaded from the Oracle Social Data and Insight Cloud service.

Perform the following seven tasks to set up DaaS social attributes and enrich the Oracle Sales Cloud account and contact data:

- Identifying the social attributes supported by Data as a Service
- Creating custom fields for social attributes in Sales Cloud
- Adding custom social attributes to UI pages
- Mapping custom social attributes with the DaaS supported social attributes
- Adding the custom social attributes to select fields to enrich page
- Enriching account and contact data with custom social attributes
- Publishing the sandbox used for creating custom fields

Social Attributes Supported by Data as a Service (DaaS): Explained

Data as a Service (DaaS) supports the following social attributes or handles for accounts (companies) and contacts.

For a complete list of account (company) and contact attributes, see the Understanding Attributes chapter in the Using Oracle Social Data and Insight Cloud Service guide. The following is the list of all the supported Data Cloud account (company) social attributes or handles:

<table>
<thead>
<tr>
<th>Company Data Attribute</th>
<th>Geographic Availability</th>
<th>Length of Data in Characters</th>
<th>Subscription Level</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blog URL</td>
<td>Global</td>
<td>350</td>
<td>D&amp;B Company Enterprise Records</td>
<td>Blog URL for the company.</td>
</tr>
<tr>
<td>Twitter URL</td>
<td>Global</td>
<td>350</td>
<td>D&amp;B Company Enterprise Records</td>
<td>Twitter URL for the company.</td>
</tr>
<tr>
<td>Twitter Followers</td>
<td>Global</td>
<td>9</td>
<td>D&amp;B Company Enterprise Records</td>
<td>Number of Twitter followers for the company Twitter URL.</td>
</tr>
</tbody>
</table>
Note: All the attributes listed here are of the data type VARCHAR2. This data type can be either a Text or a Long Text depending on the length of data. While creating custom fields using the application composer, you must select the type of the field according to the listed length of data in characters.

For a list of all the supported Data Cloud contact social attributes or handles, see the Related Links section.

Related Topics
- Supported Data Cloud contact social attributes or handles:

Creating Custom Fields for Social Attributes in Sales Cloud: Procedure

Enriching the Oracle Sales Cloud account and contact data with social attributes requires the creation of custom fields in Oracle Sales Cloud for the social attributes.

You must create a sandbox before you create custom fields using the Application Composer.

Creating a Sandbox for Creating Custom Fields

Complete the following steps to create a sandbox to create custom fields:

1. Sign in to Oracle Sales Cloud as Sales Administrator.
2. Click the Settings and Actions menu.
3. Click Manage Sandboxes. The Manage Sandboxes page appears.
4. Click the Actions menu on the Manage Sandboxes page.
5. Select New.
6. Enter the sandbox name and description in the Create Sandbox dialog box.
7. Click Save and Close. A confirmation message dialog box appears.
8. Click OK in response to the confirmation message.
9. Navigate to the Manage Sandbox page.
10. Search for the newly created sandbox in the Manage Sandbox page.
11. Select the newly created sandbox.
12. Click Set as Active.
13. Click Yes in response to the warning message.

Now you are in the new sandbox and are ready to create social attributes.

Creating Custom Fields Using the Application Composer

Complete the following steps to create a custom field using the Application Composer:

1. Sign in to Oracle Sales Cloud as Sales Administrator.
2. Navigate to the Application Composer from the Tools section of the Navigator work area.
3. Navigate to Fields through Standard Objects and Account or Contact in the Application Composer. See the following screenshot.

**Note:** This section describes how to create a custom field for the account object. The procedures for creating a custom field for the account and contact objects involve the same steps. You may navigate to Account or Contact under the Standard Objects in the Application Composer UI accordingly.

4. Click Actions in the Fields page.
5. Select the Create option to add a new custom field. The Select Field Type dialog box appears.

![Image of Application Composer fields](image.png)

6. Select the field type as Text or Long Text or Number.

   **Note:** You must select the type of the field according to the length of the data in characters listed in the "Identifying the Social Attributes Supported by Data as a Service (DaaS)" section. All the attributes listed in the section are of the data type VARCHAR2. VARCHAR2 can be either a Text or a Long Text depending on the length of the data in characters.

7. Click OK. The Create Text Fields page appears.
8. Enter the Display Label and Help Text of the social attribute. Name is auto populated based on the display label.
9. Change the attribute data length if required.

   **Note:** The maximum data length for all the supported social URLs is 2080 characters.

10. Click Save and Close.

    Once the attribute is created, you can see it on the Fields page, under the Custom tab.

Repeat this procedure to create additional custom fields or attributes.

### Adding Custom Sales Cloud Social Attributes to UI Pages: Procedure

To enrich newly created custom fields for social attributes in Sales Cloud, you must add the attributes to the simplified pages in the Application Composer. When you add the attributes in the simplified page, it is displayed in the details page of Account and Contact.
Adding Account Custom Fields to Simplified Pages

Complete the following steps to add the account custom fields to simplified UI pages:

1. Navigate to Pages through Account in the Application Composer.
2. Select the Simplified Pages tab on the Accounts: Pages UI.
3. Click Actions under the Details Page layouts. To add new custom fields, you must create a duplicate page layout based on the default page layout.
5. Specify the New Layout Name and select Default Layout as the Source Layout.
6. Click Save and Edit. The Creation layout: Social Attributes page appears.
7. Click the Edit icon next to FUSE Customer ObjectCreation View. The Edit Creation page appears.
8. Select the newly created fields from the Available Fields (left) pane of the box and use the arrow buttons to move the required into the Selected Fields (right) pane.
9. Click Save and Close after editing and adding these custom fields.

Accounts Custom fields (social attributes) are now added to the Simplified Pages.

Adding Contact Custom Fields to Simplified Pages

Complete the following steps to add contact custom fields to the simplified UI pages:

1. Navigate to Pages through Contact in the Application Composer.
2. Select the Simplified Pages tab in the Contact: Pages UI.
3. Click Actions under the Details Page Layouts. To add new custom fields, you must create a duplicate page layout based on the default page layout.
5. Specify the New Layout Name and select the Default Layout as the Source Layout.
6. Click Save and Edit. The Details Layout: Social Attributes page appears.
7. Click the Edit icon next to Contact Overview Form. The Edit Details Page Summary Form page appears.
8. Select the newly created fields from the Available Fields pane of the box and use the arrow buttons to move the required into the Selected Fields pane.
9. Click Save and Close after editing and adding these custom fields.

The Contact Custom fields (social attributes) are now added to the Simplified Pages.

Mapping Custom Social Attributes with the DaaS

Supported Social Attributes: Procedure

A newly created custom field must be mapped to the Data Cloud social attributes to show the fields on the data matching screen. This enables you to enrich Oracle Sales Cloud account and contact data with the enriched social data downloaded from the Oracle Social Data and Insight Cloud service.

Mapping the Sales Cloud Account Attributes with the Data Cloud Social Attributes

The following are the steps to map Sales Cloud account attributes with the Data Cloud Social attributes:

1. Navigate to the Setup and Maintenance page.
2. Enter Manage Social Data and Insight Cloud Attribute Mapping and Enrichment Preferences in the tasks search box.
3. Click Search.
4. Navigate to Manage Social Data and Insight Cloud Attribute Mapping and Enrichment Preferences UI.
5. Click the Attribute Mappings tab on the Manage Social Data and Insight Cloud Attribute Mapping and Enrichment Preferences page.
6. Search for the social attributes in the Social Data Cloud and Insight Attribute column that you must map with the custom fields.
7. Select Account as the Sales Cloud Object.
8. Click the corresponding Sales Cloud Attribute drop-down list. The Search and Select FusionAttributeName dialog box appears.
9. Enter the name of the custom attribute that you created.
10. Click Search.
11. Select the Custom attribute from the search results table.
12. Click OK.
13. Click Save and Close on the Manage Social Data and Insight Cloud Attribute Mapping and Enrichment Preferences Page.

The mapping of the Sales Cloud account attributes with the Data Cloud Social attributes is now complete.

Mapping the Sales Cloud Contact Attributes with the Data Cloud Social Attributes

The following are the steps to map Sales Cloud contact attributes with the Data Cloud Social attributes:

1. Navigate to the Contact Attributes section on the Manage Social Data and Insight Cloud Attribute Mapping and Enrichment Preferences page.
2. Search for the Data Cloud contact social attributes in the Social Data Cloud and Insight Attribute column.
3. Select Contact from the Sales Cloud Object drop-down list.
4. Click the corresponding Sales Cloud Attribute drop-down list. The Search and Select FusionAttributeName dialog box appears.
5. Enter the name of custom attribute that you created.
6. Click Search.
7. Select the custom attribute from the search results.
8. Click OK.
9. Click Save and Close on the Manage Social Data and Insight Cloud Attribute Mapping and Preferences page.

The mapping of the Sales Cloud contact attributes with the Data Cloud Social attributes is now complete.

Adding the Custom Social Attributes to Select Fields to Enrich Page: Procedure

To enrich a Sales Cloud custom social attribute it must be displayed on the Select Fields to Enrich page. The following are the steps to add the custom social attributes to display on the Select Fields to Enrich page:

1. Click the Preferences tab on the Manage Social Data and Insight Cloud Attribute Mapping and Enrichment Preferences Page.
2. Search for the newly mapped attributes in the Social Data Cloud and Insight Attribute column.
3. Select the Display on Select Fields to Enrich page options for the newly mapped attributes.
4. Click Save and Close.
You can see the newly created attributes on the Select Fields to Enrich page. In the matching screen, you can select the Data Cloud data to overwrite Sales Cloud data.

Similarly, you can enable the contact social attributes to display on the Select Fields to Enrich page by navigating to the Contact Enrichment Preference section on the Manage Social Data and Insight Cloud Attribute Mapping and Enrichment Preferences page.

Enriching Oracle Sales Cloud Account and Contact Data with Social Attributes: Procedure

You can enrich Oracle Sales Cloud account and contact data with social attributes using the Sales Cloud simplified user interface.

Enriching Oracle Sales Cloud Account Data with Social Attributes

Complete the following steps to enrich Oracle Sales Cloud Account and Contact Data with Social Attributes:

1. Navigate to the Accounts simplified UI page.
2. Select the account that you must enrich with new custom fields.
3. Click on Actions on the Edit Account: {Account Name} page.
4. Select the Enrich Account option.
   - The Select Fields to Enrich page appears with the newly added social attributes.
5. Select the Data Cloud values as the source for the fields that you must enrich.
6. Click Enrich.
   - You can view the enriched fields in the Edit Account: {Account Name} page if data cloud has values.
7. Click Save and Close.

Enriching Oracle Sales Cloud Contact Data with Social Attributes

Complete the following steps to enrich the Oracle Sales Cloud contact data with social attributes:

1. Navigate to the Contacts simplified UI page.
2. Select the contact that you must enrich with new custom fields.
3. Click Actions.
4. Select the Enrich Contact option.
   - The Select Fields to Enrich page appears with the newly added social attributes.
5. Select the Data Cloud values as the source for the fields that you must enrich.
6. Click Enrich.
   - You can view the fields in the Edit Contact {Contact Name}: Profile page if data cloud has values.
7. Click Save and Close.

Publishing the Sandbox Used for Creating Custom Fields

The final step is to publish the sandbox used to create custom fields to get these changes across. Complete the following steps to publish the sandbox.

1. Navigate to the Manage Sandboxes page.
Enriching Sales Cloud Data Manually

Enriching Account and Account Contact Data Manually: Explained

You could manually enrich account and account contact data using the Oracle Social Data and Insight Cloud service.

To manually enrich account and account contact data:

1. Export the data you want to enrich from Oracle Sales Cloud to a CSV file.
2. Enrich the exported data using the Oracle Social Data and Insight Cloud service.
3. Download the enriched data file from Oracle Social Data and Insight Cloud service.
4. Import the enriched data file into Oracle Sales Cloud.

Exporting Data from Oracle Sales Cloud

You export account or account contact data from Oracle Sales Cloud using the Schedule Export Processes task. The export process provides two predefined objects, one each for exporting account and account contact data. While exporting data, select the Organization for Data Cloud Enrichment object to export account data and the Person for Data Cloud Enrichment object to export account contact data.

Enriching Data in the Oracle Social Data and Insight Cloud Service

You can manually enrich account or account contact data using the Enrich Data functionality of the Oracle Social Data and Insight Cloud service.

To enrich data, select the CSV file exported from Oracle Sales Cloud in the Enrich Data Records page of the Oracle Social Data and Insight Cloud service. You must select the CRM template during enrichment, so that the enriched file can be later downloaded in Oracle Sales Cloud. After enrichment is complete, a CSV file with the enriched data is available for download from the Jobs list.

For more information about Oracle Social Data and Insight Cloud service, see Using Oracle Social Data and Insight Cloud Service guide.

Importing Enriched Data into Oracle Sales Cloud

You must import the CSV file generated in the Oracle Social Data and Insight Cloud service into Oracle Sales Cloud using the file-based import functionality to import the enriched account or account contact data.

To import data into Oracle Sales Cloud using file-based import:

1. Create an import activity.
2. Enter the appropriate import details. While specifying the import object, select the Account to import account data and the Contact to import account contact data.
3. Provide the file mapping. Mapping defines the mapping between the columns provided in a source file and object attributes in the import file. Oracle Sales Cloud provides seeded mappings to import account data or account contact data from Oracle Social Data and Insight Cloud service. You must select Seeded Data to view the available seeded mappings.
While specifying the import mapping, select **Data Cloud Account Import** mapping to import account data and **Data Cloud Contact Import** mapping to import account contact data. Oracle Sales Cloud then automatically maps the data in the CSV file to object attributes.

4. Schedule the import process.

**Note:** In some cases, the import activity may display a warning about source and target attributes. You can ignore the warning message and click OK to submit the import process.

---

### Importing New Records into Oracle Sales Cloud from Oracle Social Data and Insight Cloud Service: Explained

You can import accounts or account contacts from Oracle Social Data and Insight Cloud service into your sales cloud instance.

To import data from Oracle Social Data and Insight Cloud service to Oracle Sales Cloud:

1. Export the accounts or account contacts from Oracle Social Data and Insight Cloud service.
2. Import the CSV file into Oracle Sales Cloud.

Before importing account contacts, you must ensure that the accounts related to the contacts you want to import exist in Oracle Sales Cloud.

### Exporting Accounts or Account Contacts from Oracle Social Data and Insight Cloud Service

You can either export a set of account or account contact data or export all available accounts and account contacts from the Oracle Social Data and Insight Cloud service.

You can search for accounts and account contacts you want to import from the Search window. After searching for the records you want to export, click **Export** to export the data. You must select **Use CRM Template** as the export template so that the exported file can be later imported in Oracle Sales Cloud.

For more information about Oracle Social Data and Insight Cloud service, see Using Oracle Social Data and Insight Cloud Service guide.

### Importing Accounts or Account Contacts into Oracle Sales Cloud

You can import the account or account contact data exported from Oracle Social Data and Insight Cloud service into Oracle Sales Cloud using the file-based import functionality. You create an import activity, enter the appropriate import details, provide the file mapping, and schedule the import to import the data from the CSV file into staging tables. Import processing is subject to the settings defined for the import object, mapping, and import activity.

To import data into Oracle Sales Cloud using file-based import:

1. Create an import activity.
2. Enter the appropriate import details. While specifying the import object, select **Account** to import account data and **Contact** to import account contact data.
3. Provide the file mapping. Mapping defines the mapping between the columns provided in a source file and object attributes in the import file. Oracle Sales Cloud provides seeded mappings to import account data or account contact data from Oracle Social Data and Insight Cloud service. You must select **Seeded Data** to view the available seeded mappings.
While specifying the import mapping, select **Data Cloud Account Import** to import account data and **Data Cloud Contact Import** to import account contact data. Oracle Sales Cloud then automatically maps the data in the CSV file to object attributes.

4. Schedule the import process.

### Enriching Sales Cloud Data Automatically

#### Enriching Sales Cloud Automatically: Overview

You can enrich sales cloud data automatically using the Oracle Social Data and Insight Cloud service. Before you can enrich data, you must first purchase a third-party data service subscription and your administrator must create a new user with the **Data Service Client API ApplID** role. This user is used by Oracle Sales Cloud to securely communicate with the Oracle Social Data and Insight REST APIs.

Before you start enriching sales cloud data, you must:

- Set up Sales Cloud and Data Cloud integration
- Set up automatic enrichment preferences and administrator profile options.

After setting up the enrichment and integration preferences, you can enrich an account or contact from the Accounts or Contacts pages.

#### Managing Social Data and Insight Cloud Attribute Mapping and Preferences: Procedure

This topic illustrates configuring the enrichment preferences and attribute mappings for Oracle Social Data and Insight Cloud Service. You must configure the enrichment preferences and attribute mappings before enriching accounts or contacts.

To configure enrichment preferences, you must:

- Enable and configure enrichment preferences for account and contact.
- Configure attribute mappings.
- Set administrator profile options.

#### Enabling and Configuring Enrichment Preferences

You can configure the enrichment preferences to define what data is updated in an enriched account or contact.

Perform the following steps to enable and configure enrichment preferences:

1. Navigate to the Setup and Maintenance work area.
2. Enter **Manage Social Data and Insight Cloud Attribute Mapping and Preferences** in the task search box.
3. Click Search.
4. In the search results, click the Go to Task button for the task. The Manage Social Data and Insight Cloud Attribute Mapping and Preferences page appears.
Note: The Manage Social Data and Insight Cloud Attribute Mapping and Preferences page includes two regions, one each for account and contact. You can use these to enable enrichment for account and contact, and configure the data that should be included in the enriched account.

5. Select Enable Enrichment for Account or Enable Enrichment for Contact to enable enrichment for account or contact respectively.

6. To define the data in an enriched account or contact, you can do one of the following:
   - Select Auto fill Blank Fields in Record to update all blank Sales Cloud fields with the data from the Oracle Social Data and Insight Cloud service. This updates all fields in an account or contact record with the data from the Oracle Social Data and Insight Cloud service.
   - Select Customize Settings by Field to define what data should be updated for each field in the enriched account or contact.

7. To customize data at the field level, select:
   - Auto fill Blank: To update the field with data from Oracle Social Data and Insight Cloud service if it is blank.
   - Update with Data Cloud Value: To update all blank fields with data from Oracle Social Data and Insight Cloud service.
   - Display on Select Fields to Enrich page: To display the value in the Select Fields on the Enrich page and prompt the user to select the field value.

8. Enter a match threshold for account and contact in the Match Threshold field.

9. Enter the maximum number of records that should be enriched in the Maximum Number of Accounts for Real-Time Enrichment and Maximum Number of Contacts for Real-Time Enrichment fields.

10. Enter the maximum number of new contacts you want to create during enrichment in the Maximum Number of New Contacts field.

11. Click Save.

Note: Make sure that you have set the administrator profile options using the procedures documented in the Manage Administrator Profile Values section of this document.

If you have a preproduction or staging environment for Oracle Sales Cloud, then test the service association between your Sales Cloud preproduction environment and your Data Cloud trial subscription. You must limit testing to 500 records. When you go to production, do the service association between your Sales Cloud production environment and your Data Cloud production environment. The service association steps are the same, but the trial or production service URLs are different. The service URLs are listed in your welcome e-mail and in the My Services application.

Setting up Administrator Profile Options

Complete the following steps to set administrator profile options:

1. Navigate to Setup and Maintenance.
2. Enter Manage Administrator Profile Values in the task search box.
3. Click Search.
4. Navigate to Manage Administrator Profile Values task. The Manage Administrator Profile Values page appears.
5. Enter DAAS_PRODUCTION_MODE as the Profile Option Code on the Manage Administrator Profile Values page.
6. Click Search. The Manage Administrator Profile Values page with the specified Profile Option appears.
7. Make sure that the Profile Level Site value is set to No.

   Note: Do not set the Profile Level Site value to Yes. The only supported value today is No.

8. Click Save and Close.
Configuring sales Cloud to Data Cloud Integration: Procedure

You can configure the Sales Cloud to the Data Cloud Integration after you set up the DaaS user. To configure the Sales Cloud to the Data Cloud Integration you navigate to Sales Cloud for adding DaaS URL and credentials. The following are the steps to configure sales Cloud to Data Cloud Integration:

1. Navigate to Setup and Maintenance.
2. Enter Manage Sales Cloud to Data Cloud Integration in the task search box.
3. Click Search.
4. Navigate to Manage Sales Cloud to Data Cloud Integration page.
5. Enter the following information:
   a. URL of your DaaS instance.
   b. This service instance URL is available in the post-activation e-mail and on the My Services application.
   c. Select oracle/wss_username_token_over_ssl_client_policy as Security Policy.
   d. Enter the DaaS user name and password you created in DaaS setup.
6. Click Save and Close.

Enriching Account and Account Contact Data Automatically: Procedure

This topic illustrates how you can enrich account and account contact data automatically from Sales Cloud, using Oracle Social Data and Insight Cloud Service. Oracle Social Data and Insight Cloud Service includes authenticated D&B records of companies and contacts. You can use the service to authenticate and enrich account and contact data from Sales Cloud.

Enrich Account Data Automatically

You enrich account data from the Accounts page of the simplified UI.

To enrich account data:

1. In the Account page of the simplified UI, select the accounts you want to enrich.
2. Select Enrich Selected Account from the Actions menu.

   If you select multiple accounts for enrichment, Sales Cloud automatically enriches the accounts and displays the results. The result includes information about the accounts enriched and the status of each account.

   If you select one account for enrichment, the Select Fields to Enrich page is displayed. In the page, you can select what data values you would like use in the enriched account record for each of the fields.

3. For each of the field, select if you want to use existing value or the Data Cloud value.
4. Optionally, click on Hide Identical Fields to hide the fields where the Sales Cloud data is same as the Data Cloud data.
5. Click Enrich.

Enrich Account Contact Data Automatically

You enrich account contact data from the Contacts page of the simplified UI.

To enrich contact data:

1. In the Contacts page of the simplified UI, select the contacts you want to enrich.
2. Select **Enrich Selected Contacts** from the Actions menu.

   If you select multiple contacts for enrichment, Sales Cloud automatically enriches the contacts and displays the results. The result includes information about the contacts enriched and the status of each contact.

   If you select one contact for enrichment, the Select Fields to Enrich page is displayed. In the page, you can select what data values you would like use in the enriched contact record for each of the fields.

3. For each of the field, select if want to use the existing value or the Data Cloud value.

4. Optionally, click on **Hide Identical Fields** to hide the fields where the Sales Cloud data is same as the Data Cloud data.

5. Click **Enrich**.
13 Managing Customer Data Quality

Overview of Data Quality Management

Data Quality Management Components: How They Fit Together

You manage the Sales Cloud data quality using these components:

- Data quality configurations
- Data quality management during import process
- Duplicate identification, resolution, and cleansing

Additionally, you can manage party information using the Party Center work area.

Data Quality Configurations

Data quality configurations define how the data quality services, such as data quality matching and cleansing, are run during real-time and during batch execution. Data quality services use the configurations to invoke appropriate services of the embedded data quality engine.

Data Quality Management During Import Process

You can configure the import process to:

- Define deduplication within the data being loaded.
- Define deduplication of the data being loaded against the records that exist in the registry.

You also select a matching configuration to identify duplicates and specify the action to perform on the duplicate records during the import process.

Duplicate Identification, Resolution, and Cleansing

Duplicate identification lets you identify potential duplicates during data entry, data integration, or among records already in the application.

Duplicate resolution lets you resolve duplicates either by directly merging them or by creating duplicate resolution request such as merge or link. The resolution request can be verified, approved or rejected, and processed at a later date.

Data cleansing lets you cleanse data existing in the registry and ensures data accuracy over time. Real-time data cleansing ensures that the incoming data from source systems follows the same convention as the target system for consistent information.

Party Information Management

You can manage party information, such as profile, usage assignments, linked parties, contacts, accounts, tasks, interactions, and notes, in the Party Center work area.
Cleanse Customer Information

Scheduling a Data Cleansing Process: Points to Consider

Data cleansing processes can be time consuming, depending on the batch size.

Consider the following before scheduling a data cleansing process:

- Time required for the process
- System resources
- Process dependencies

Time Required for the Process

The time required to complete the data cleansing process depends on the number of records, cleansing complexity, and hardware characteristics. You must calculate the possible time taking into account these three factors and the available benchmark.

System Resources

You must estimate system performance based on recent data, by reviewing the batch and environment statistics. You can also review various object lists to know the time when most system resources are available for the data cleansing process.

Process Dependencies

You must understand process dependencies before scheduling the data cleansing process. For example, you must consider whether other processes are waiting for the results of the data cleansing process.

Creating Address Cleansing Batches: Points to Consider

You create an address cleansing batch to cleanse and validate addresses. Consider the following before creating an address cleansing batch that best suits your address cleansing needs:

- Use a copy of an existing address cleansing batch
- Create an address cleansing batch for a single run or to be run periodically

Use a Copy of an Existing Address Cleansing Batch

You can quickly create an address cleansing batch by copying an existing cleansing batch. You can edit the batch details such as batch selection criteria, address cleansing mode, and scheduling before you submit the batch.

You create a new address cleansing batch, instead of copy an existing cleansing batch, if your address cleansing batch details are unique.

Create an Address Cleansing Batch for a Single Run or to be Run Periodically

You can create an address cleansing batch for a single run or to be run periodically. You can run address cleansing batches periodically for tasks that are mostly ongoing and repetitive, such as the execution of registry address cleansing.
You can schedule the batch to run at a specific time or at a specific interval. For example, you can schedule the batch to run daily, every night at 9 PM, weekly, or every Sunday at 10 PM.

Specifying Selection Criteria for a Data Cleansing Batch: Worked Example

Batch cleansing lets you validates and cleanses address data that already exist in the registry. This resolves existing errors as well as updates the addresses with the changes in the geography reference data such as postal code.

You can specify selection criteria in the Create Data Cleansing Batch page. You define the criteria similar to a database query for better results. For example, multiple selection criteria are joined with an AND condition.

To specify the selection criteria:

1. Navigate to the Create Data Cleansing Batch page.
2. Enter the following information in the **Selection Criteria** table.

<table>
<thead>
<tr>
<th>Object</th>
<th>Attribute</th>
<th>Operator</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Location</td>
<td>Address1</td>
<td>Contains</td>
<td>500 Oracle</td>
</tr>
<tr>
<td>Location</td>
<td>Country</td>
<td>Equal to</td>
<td>US</td>
</tr>
</tbody>
</table>

3. Click **Submit**.

Accepting Data Cleansing Results: Critical Choices

You can accept or reject the results of a batch data cleansing process run in simulated mode as follows:

- Accept or reject the data cleansing process results for the entire batch
- Accept or reject the data cleansing process results by country
- Accept or reject data cleansing process results by address record

> **Note:** If you submit a batch for cleansing in the automated cleansing mode, then you can't accept or reject the process results. The cleansed data is automatically saved to the database.

Accept Or Reject the Data Cleansing Process Results for the Entire Batch

Accept the results for the entire batch only if you are sure that the data cleansing results are accurate. Reject the results if you are not sure about the reliability or accuracy of the batch results, or if the process results may reduce the data quality.

Accept or Reject the Data Cleansing Process Results by Country

The data cleansing process results depend on the data quality, coverage, and the geography information of the country or region.
Consider the following before accepting or rejecting data cleansing process results by country:

- Accept data only for countries with acceptable cleansing results if the percentage of errors is high and the results for all countries are not acceptable.
- Reject data for countries with very high percentage of errors.

**Accept or Reject Data Cleansing Process Results by Address Record**

Accept or reject cleansing results for each individual address based on the cleansing status and business rules and policies.

**FAQs for Cleanse Customer Information**

**What's a Data Cleansing Batch?**

Data cleansing batch is a set of records that is submitted for the data cleansing. Batch cleansing validates and cleanses address data in the registry. This resolves existing errors as well as updates the addresses with the changes in the geography reference data such as postal code.

**Can I resubmit a data cleansing batch?**

No. However, you can edit and resubmit the data as a new batch.

**What's the difference between automated cleansing and simulated cleansing?**

Automated Cleansing mode doesn't require any user action. You use this mode to automatically save cleansed data batch to the database.

Simulated Cleansing mode lets you review the output of the batch cleansing process before saving the cleansed data. Additionally, you can accept or reject cleansed data in its entirety, by country, or by individual address.

**Why did the batch cleansing process skip some records?**

Batch cleansing skips a record if the record is already cleansed and validated. For example, an address that already follows the formats, norms, and standards of the related postal directory.

**Identify Duplicate Customer Information**
Managing Duplicate Sets: Explained

You manage duplicate sets by reviewing the details of the completed duplicate identification batch, and submitting duplicate resolution requests.

To manage duplicate sets you must:

- Decide whether to resolve duplicates within each identified duplicate set through a merge or link request. You create a merge request to combine duplicate records, a link request to join duplicate records, or a generic request to select the resolution later.
- Determine which records to remove from the duplicate resolution request. You can remove records that you do not want include in the duplicate resolution request. You can view the removed records in the removed records table, and can also restore these records back to the set.
- Mark records as nonduplicate. You can mark a pair of records as nonduplicate to prevent them from being identified as potential matches. To remove the pair from nonduplicate list, end date the nonduplicate record.
- Change the master record.
- Submit the duplicate sets as merge or link requests for resolution.

Creating Duplicate Identification Batches: Points to Consider

You have the following options while creating a duplicate identification batch:

- Create the duplicate identification batch for a single run or to be run periodically
- Create the duplicate identification batch from a copy of an existing duplicate identification batch

Duplicate Identification Batch for a Single Run or to be Run Periodically

You can create a duplicate identification batch for a single run or to be run periodically. You can run duplicate identification batches periodically for tasks that are mostly ongoing and repetitive, such as the execution of registry duplicate identification.

You can schedule the batch to run at a specific time or at a specific interval. For example, you can schedule the batch to run daily, every night at 9 PM, weekly, or every Sunday at 10 PM.

Using the Copy of an Existing Duplicate Identification Batch

You can quickly create a duplicate identification batch by copying an existing duplicate identification batch. You can edit the batch details such as batch selection criteria, duplicate identification rules, and scheduling before you submit the batch.

You create a new duplicate identification batch, instead of copy an existing batch, if your duplicate identification batch details are unique.

Defining Subset Rules: Worked Example

This example demonstrates how to define subset rules, also known as batch criteria rules, to create duplicate identification batches.

You can identify potential duplicate records either manually or by creating a duplicate identification batch. While creating a duplicate identification batch, you can use subset rules to specify the criteria for retrieving a subset of records in the batch. The data quality engine identifies duplicate records from this subset of records. The rules for identifying duplicates are defined
by the Relation Type Matching option. You can either select the Match all keywords option to perform an AND operation or the Match any keyword option to perform an OR operation.

In this example, we will create a rule to retrieve a subset of records where the person name contains John and the address contains Redwood.

To create a subset rule:

1. Select the Create Duplicate Identification Batch task from the Customer Data Management Dashboard work area.
2. In the Subset Rules for Identifying Duplicates region, select Add from the Actions menu.

Adding New Subset Rules

1. Navigate to the Create Duplicate Identification Batch page as follows: Customer Data Management Dashboard - Duplicate Identification Overview - Create Duplicate Identification Batch
2. Use the Create menu option or button to add a new subset rule.
3. Enter the following information in the Subset Rules for Identifying Duplicates table.

<table>
<thead>
<tr>
<th>Object</th>
<th>Attribute</th>
<th>Operator</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Person</td>
<td>Name</td>
<td>Contains</td>
<td>John</td>
</tr>
<tr>
<td>Address</td>
<td>Address Line 1</td>
<td>Contains</td>
<td>Redwood</td>
</tr>
</tbody>
</table>

4. Enter other details about the batch and click on Save and Close.

View Nonduplicate Mapping

Nonduplicate Records: Explained

You mark a pair of records as nonduplicate to prevent the pair from being identified as potential matches.

You can modify the period for which a record should be considered as a nonduplicate by changing the end date. You can also delete the nonduplicate status of the records by end dating it.

FAQs for Assign Duplicate Identification Batch

Why am I unable to assign or reject duplicate identification batches?

Only a Data Steward Manager can assign new duplicate identification batches to data stewards for review and processing. You can initiate the assignment of duplicate identification batches to an assignee from the Duplicate Identification work area, or the customer data management dashboard, and complete it on the Assign Duplicate Identification Batch page. Data Steward Managers can reject any duplicate identification batch, but they can only reject the batches that are assigned to them. Ensure that the users to whom you want to assign a duplicate identification batch have the Customer Data Steward resource role assigned to them. Unless the users have this role assigned, you may not see their names in the ‘Assign To’ drop down on the duplicate resolution UI.
How can I cancel a batch process?

You can cancel a batch process in the Scheduled Processes Overview page. You must search for the batch process by its process ID, and then cancel the batch process request.

Resolve Duplicate Customer Information

Creating Duplicate Resolution Request: Explained

You create a duplicate resolution request to resolve potential duplicates in the data registry. You can either manually identify duplicates for a duplicate resolution request or create a duplicate identification request.

While creating a duplicate resolution request, set Automatic Processing Option to Create Duplicate Resolution Request to submit a generic resolution request.

Identify Duplicates Manually

You can search for parties, by specifying search criteria, and identify the parties that may be duplicates. You can then decide to either to create a merge or link request for the records.

Create a Duplicate Identification Request

You can create a duplicate identification batch to identify the duplicates that you want to resolve. The batch process uses the data quality matching engine to identify duplicates.

To resolve duplicates using a duplicate identification batch:

1. Create a duplicate identification batch to identify potential duplicates.
2. Review the duplicate sets and create either merge or link requests to resolve the duplicates.

Selecting Duplicate Resolution Request Type: Critical Choices

You can create merge request or a link request to resolve potential duplicates. You decide the type of request to create based on your business needs. The duplicate resolution request type determines how the records are maintained after the resolution process is complete.

Merge Request

You create a merge duplicate resolution request to combine duplicate records into one new master record. After the merge request is completed, all duplicate records are deleted from the data registry and you can update only the master record.

Link Request

You create a link request to associate the duplicate records. The linked records are treated as unique records in the data registry, and have their own unique identifiers.

While creating the request, you define one of the records as the master record. The resolution process marks links the other records to the master records, and marks them as duplicates.
Merging Duplicate Records: Explained

This topic discusses the process to merge records for resolving duplicates. To resolve duplicates, you can either merge duplicates or link duplicates.

The following figure illustrates the merge process flow:
Creating a Merge Request

A merge request is created as a result of one of the following:

- The batch duplicate identification process creates a duplicate resolution request automatically for each identified duplicate set.
- A data steward or a user create a merge request submit a merge request through create duplicate resolution request or through the data quality dashboard.
- A bulk or file-based import process creates a merge request.

Note: You can also use the duplicate resolution request web service to create merge requests.

Status of Merge Requests

The following table lists the possible statuses of a merge request:

The possible statuses of a merge request are described below:

<table>
<thead>
<tr>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pending</td>
<td>All merge requests, and generic duplicate resolution requests, are created with the status as Pending. The requests with pending status require review by a data steward.</td>
</tr>
<tr>
<td>New</td>
<td>The request dispatcher is run periodically for pending requests, and sets status of pending requests to New. When the request dispatcher is run, Oracle Sales Cloud selects the master record and attributes based on the survivorship rules, and run the agreement rule check for the request.</td>
</tr>
<tr>
<td>Submitted</td>
<td>This status indicates that the merge request has been submitted for processing. You can also create merge requests with the status as Submitted to process the request without data steward intervention.</td>
</tr>
<tr>
<td>Rejected</td>
<td>This status indicates that the merge request is reject by the data steward or by the process for agreement rule violations. You can restore a rejected merge request.</td>
</tr>
<tr>
<td>Error</td>
<td>This status indicates the request has failed because one or more records cannot be processed.</td>
</tr>
<tr>
<td>Completed</td>
<td>This status indicates that the request has been processed successfully.</td>
</tr>
</tbody>
</table>

Selecting Master and Records to Merge

You can review new merge requests to:

- Select the master record
- Mark records as nonduplicates
- Remove records from the duplicate set to avoid merging them

The merge process selects the master record based on the set master rules, if the survivorship rules are active. However, you can select to review and choose to override system settings.
Deciding to Override System Mapping
You can override system mappings, during the merge request review, if you are not satisfied with the system mapping of the request. You can select the accounts, account addresses, and the relationships that should be merged, transferred, or removed.

Reviewing Merge Request Related Notifications
The merge process sends any of the following related notifications:

- Rejection: This notification is sent to the merge request's initiator, such as a business users, sales person, or administrator. You can restore and resubmit a rejected merge request, by selecting a different master record or by remove removing some records from the batch.
- Error: This notification is sent to the merge request's initiator and submitter. The notification indicates that the merge request has errors and cannot be process.
- Completion: This notification is sent to the initiator and the assignee of the merge process. The process merges duplicate parties and updates the parties' hierarchies. The notifications include information such as the master and duplicate records.

After receiving the completion notification, the request initiator and assignee can:

- Verify if the updated hierarchies are correct in the Manage Hierarchies work area.
- Optionally, make the necessary edits to the updated hierarchies.

Related Topics
- Overriding System Mapping: Procedure

Automerger: Explained
The automerge functionality merges duplicate records without any approval or intervention from the data steward. Automatic processing of merge requests is critical when processing large volumes of customer data as automerge can expedite the resolution of duplicate records without manual review. Note that during automerge, the child entities of the duplicate records, such as contact points, relationships, classifications, and cross references, become the child entities of the master record.

Understanding How Records are Selected for Automerge
Records are selected for automerge based on the following criteria:

- Match threshold: The match threshold is defined in the Match Configuration and determines if a record is included in a duplicate set.
- Merge threshold: The merge threshold is defined by the ZCH_AUTO_MERGE_THRESHOLD profile option and determines if the merge request for a duplicate set is processed automatically or if it must be reviewed manually.

Three possible outcomes for each record with regard to duplicate identification and merging are as follows:

- Low score below match threshold: The record is not included in duplicate set and in the merge request for that duplicate set.
- Medium score above match threshold and below merge threshold: The record is included in duplicate set but merge request for that duplicate set must be reviewed manually.
- High score above match threshold and above merge threshold: The record is included in duplicate set and merge request is processed automatically.
The score for all the records in a duplicate set must be above the merge threshold for automated processing. If one record in the duplicate set is below merge threshold, and the other records are above the merge threshold, the merge request must be reviewed manually.

Configuring Automerge

Enabling Automerge involves several implementation steps that must be completed by an implementor using the following Setup and Maintenance tasks:

- Manage Customer Hub Profile Options: Navigate to this Setup and Maintenance task to perform the following implementation steps:
  - Set Auto Merge Threshold profile option (ZCH_AUTO_MERGE_THRESHOLD) to the wanted value. This profile option specifies the threshold for auto merge. Merge requests with lower scores need data steward review. An exact match is 100.
  - Review the Record Size Limit of Duplicate Set (ZCH_DI_MERGEREQ_REC_SIZE). This profile option determines the maximum number of records in the duplicate set that can be merged automatically. By default, the maximum number is set to 10 records.
  - Set the Survivorship Enabled profile option (ZCH_ENABLE_SURVIVORSHIP) to Yes. This profile option enables the survivorship rules to select the master record and retain the attributes during a merge or update operation.

- Define Survivorship: Navigate to this Setup and Maintenance task to create Set Master survivorship rules to choose the master record for merge requests created from the duplicate identification batch and set the rule to active.

  If there are no active Set Master rules or if the Set Master rules did not trigger, the merge request must be reviewed manually, even if the ZCH_AUTO_MERGE_THRESHOLD profile option is set, the score for all records is above the threshold value, and the number of records is below the record size limit.

  **Note:** You can use the Set Attribute rules with Set Master rules to determine the Golden Master record. For automerge, Set Master rule is mandatory.

- Define Data Quality: Navigate to this Setup and Maintenance task to perform the following implementation steps:
  - Create an active Match Configuration in Manage Enterprise Data Quality Matching Configurations task or use a predefined Match Configuration. Rebuild the keys if necessary.
  - Enable EDQ Real Time and Batch Basic Match Server in Manage Server Configurations task.

Running Automerge

This task involves the following two steps:

1. Create a duplicate identification batch and select Create Merge Request as the Automatic Processing Option.
2. Run the task Run Request Dispatch Job to disposition the duplicate resolution sets.

The Dispatch Job processes any resolution request in Pending or Submitted status. You can run this job in two modes:

- On demand: Navigate to Run Request Dispatch Job task and click Submit
- Per a specific schedule: Do the following steps to set up a recurring job:
  a. Click Advanced on the Run Request Dispatch Job task.
  b. Click Schedule tab and select the Using a Schedule radio button.
  c. Select the frequency you want and click Submit.
To see the list of dispatch jobs, and their statuses, navigate to Scheduled Processes under Tools. The name of the process is Schedule Duplicate Resolution Requests.

Troubleshooting Automerge Issues

After you create your Duplicate Identification Batch, drill down into the completed batch to see the results. If duplicate sets have been found, and automerge is enabled, resolution requests are automatically submitted for merging.

If the resolution request was not submitted automatically, you can drill down to the duplicate set and compare the score for each record with the threshold in the ZCH_AUTO_MERGE_THRESHOLD profile option and the number of records with the limit in the ZCH_DI_MERGEREQ_REC_SIZE profile option. If all scores are above the threshold and the number of records is below the limit, verify that Set Master rules are active and triggered to choose a master for the records in the duplicate set, and that ZCH_ENABLE_SURVIVORSHIP is set to yes.

Performing Automerge: Worked Example

This example demonstrates how to perform automerge or automatically merge confirmed potential duplicates based on duplicate resolution requests created by invoking the DuplicateResolutionRequestService Web service using Oracle Enterprise Manager (OEM). This example assumes that the survivor is sales account and victims include sales account. Consequently the merge behavior is to retain the survivor’s sell-to address and merge the victim’s sell-to addresses to the survivor’s sell-to address.

Performing automerge involves the following tasks:

- Enabling the automerge
- Accessing the DuplicateResolutionRequestService Web Service using OEM
- Creating an automatic merge request
- Finding out the merge result

Enable Automerge

1. Set the User Merge Requests profile option to Allow Processing Without Approval. When the User Merge Requests profile option is set to this value, user merge requests are processed immediately without requiring any approval from Data Steward.
2. Set the Merge Request Enabled (ZCA_MERGE_REQUEST) profile option to Yes.
3. Enable the User Merge Requests (ZCH_USER_MERGE_REQUESTS) profile option. This profile option controls the behavior after the merge request is entered, see section Customer Hub Profile Options.

Access the Web Service Using OEM

1. Sign in to Oracle Enterprise Manager.
2. From the Navigation menu, select Fusion Applications - Oracle Fusion Customer Relationship Management.
3. Select CrmCommonApp and then select CRMCommonServer_1.
4. Under the Web Services section, select DuplicateResolutionRequestService and click Test to launch the Web Service Tester.
5. On the Request tab of the Test Web Service Page, select Custom Policy security option and enter the Policy URL syntax `oracle/wss11_username_token_with_message_protection_client_policy` and the username and password for a user who has the ZCH_MASTER_DATA_MANAGEMENT_APPLICATION_ADMINISTRATOR_JOB role. For example, in development environments, this could be MDM_ADMIN_V1.
Create Merge Request

1. On the Test Web Service Page, select the operation `createMergeRequest` from the Operation drop-down list and enter the following payload:

```xml
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
      <input:record>
        <Master>HZ_PARTIES</Master>
        <SourceSystemReference>100100001834094</SourceSystemReference>
        <SourceSystem>SIEBEL</SourceSystem>
      </input:record>
      <input:record>
        <NonMaster>HZ_PARTIES</NonMaster>
        <SourceSystemReference>100100001834098</SourceSystemReference>
        <SourceSystem>SIEBEL</SourceSystem>
      </input:record>
    </input:createMergeRequest>
  </soap:Body>
</soap:Envelope>
```

2. Click Test Web Service.

3. Review the response to the operation `createMergeRequest` and make a note of the Duplicate Resolution Request ID, which you can use in the operation `getMergeResult` to query the status of the Duplicate Resolution Request.

Get Merge Result

1. On the Test Web Service Page, select the operation `getMergeResult` from the Operation drop-down list and enter the following payload:

```xml
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
      <input:record>
        <Request>ZCH_DEDUP_HEADERS</Request>
        <SourceSystemReference>300100007253327</SourceSystemReference>
        <SourceSystem>FUSION</SourceSystem>
      </input:record>
    </input:getMergeResult>
  </soap:Body>
</soap:Envelope>
```

2. Click Test Web Service.

Managing Rejected Merge Requests: Explained

Merge requests are rejected either manually by a data steward or if the agreement rules are violated. You must be logged in as a data steward to review or restore rejected merge requests.

As a data steward, you can:

- Review rejected requests: You can review the rejected requests to view the rejection reasons.
- Restore rejected requests: After reviewing rejected requests, you can choose to correct the rejection reason and then restore rejected requests. You can either select a different master record or remove certain records from the merge request and then submit the merge request for reprocessing.
Linking Duplicate Records: Explained

You associate two or more duplicate records using linking to resolve duplicate records. This association only creates a link between the records and does not affect the records. You use this association to refer to the member records of a link.

To resolve duplicate records, you either link the records or merge the records. You can also create links for normal records, even if they are not duplicate records. In a link, you designate of the records as the master and the others as duplicates. This helps users identify the master records for use in transactions.

During merge, duplicate records are combined into a single master record. However, some parties cannot be merged because of functional or legal reasons. For example, for certain bank operations, you cannot merge duplicate records from different business units or geographical reasons. In these scenarios, we create a link between duplicate records instead of merging them.

A link does not change the party and the associated child entities such as addresses, contact points, and relationships. A link does not affect the transactional processing of the record. The link is a separate record in the data model, and has its own registry ID.

Related Topics

- Managing Links: Explained
14 Managing Party Information

Manage Party Information

Party Center: Explained
You can manage comprehensive party information using the Party Center work area. In the Party Center, you can search for a party, fetch party data from various systems, review the data, and edit the data. In Party Center, you can manage organization, households, and persons.

Party Center Tree
Party Center fetches party data from various systems and displays it as a tree, with one node for each of the data entity. The party center nodes display information, such as profile, profile history, party usage assignments, relationships, classifications, and source system references of the party.

Customizing Party Center Using Application Composer
You can customize the Party Center work area using Application Composer. Application Composer lets you create custom fields and objects, which are then available in the Party Center.

For example, in Party Center, unlike the Customer Center work area, the Organization Name field is displayed instead of the Unique Name Alias field. In this scenario, the Party Center may not display the updated organization name which is fetched from the Unique Name Alias field. You can customize Party Center in the Application Composer and add the Unique Name Alias field and display it in the Party Center.

Households: Explained
Households enable you to create a party consisting of a collection of persons and organizations, and have a business relationship with other parties. A household can be assigned to opportunities and leads.

A household includes basic information such as name, type, and certification level, and child entities such as group members, relationships, classifications, and contact methods. The group type such as Couple, Household, or Joint Ownership classifies a group based on its purpose. A group party tree manages group information and has nodes such as profile, usage assignments, relationships, source system references, and linked parties.

Group Members
A group member can be a person or an organization, and their member type attribute defines the relationship of the member with the group. For example, a member of type Group Contact is the contact for the group. You can manage member of a group from the group’s Profile party tree node.
Party Center Trees: Explained

Party center trees comprise nodes that let you fetch data about a party from various systems, and let you review and edit the data.

You can customize a party center tree by:

- Showing or hiding its nodes as required.
- Reviewing and editing the node names as appropriate.
- Determining the need for entering additional parameter values to gain access to a node.

Party Center Tree Nodes

Party center trees are different for each of the party type organization, group, and person. Each tree displays slightly different nodes, and the information that you can view and edit on each node depends upon your security privileges. However, all nodes on the person party tree are visible to all users.

A party center tree may include the following nodes:

- Profile: Review and edit party profile information such as name, party usage, additional names, additional identifiers, addresses, and contact points.
- Profile History: View the history of changes to profile records of the party, person or organization, in context in the party center.

  **Note:** The profile history information for a group party type is displayed in the Profile node.

- Usage assignments: View the party usages currently assigned to the party. Party usages that allow manual create, manual update, and unconditional assignment are maintainable through this user interface.
- Relationships: Manage all relationships for the party in context in the Party Center. The node also displays the contact details of a relationship.
- Classifications: Manage classifications for the party in context in the Party Center. The displays the primary classification first in the list of classifications.
- Source system references: Manage all source system references contributing to the master organization party record in context in the Party Center.
- Linked parties: Manage the parties linked to the party in context.

Customizing Party Center Trees: Procedure

A party center tree comprises nodes that include party data from various systems. You can customize the nodes that are displayed in the party center tree in the Manage Party Center Tree page.

To customize the party center tree:

1. In the Setup and Maintenance work area, search for the Define Customer Hub Configuration task list.
2. In the Define Customer Hub Configuration task list, select:
   - **Manage Organization Party Tree** task to customize party tree for an organization.
   - **Manage Person Party Tree** task to customize party tree for a person.
Manage Group Party Tree task to customize party tree for a household. This will open the respective manage tree page. The page displays party's available tree nodes and the node details.

3. You can customize the following for each of the node:
   - Name: Specify the name of the node that you want to display in the party center tree.
   - Visible: Specify the display status, hidden or visible, of a node in the party center tree.
   - Parameter: Specify the additional parameter values required to gain access to a node.

You can also select a node and click on the Set as Default icon, to set the node as the default node.

4. Click Save and Close to save the changes you have made to the nodes.

Name
Set this attribute to specify the name of the node that you want to display in the party center tree.

Visible
Set this attribute to specify the display status, hidden or visible, of a node in the party center tree.

Parameter
Specify the additional parameter values required to gain access to a node.

Party Profile: Explained

The party profile includes party information such as party details, additional names, additional identifiers, addresses, and contact points; and depends on the party type and party usage.

You can add most of the party profile information during party creation, except for some profile attributes, such as an organization’s financial and bank details. You can edit these profile attributes from the Edit Profile page. The Edit Profile page for a party lets you set a primary contact point or address for the party.

Oracle Sales Cloud maintains a history of the changes to the party profile that you can view in the Profile History page.

Some of the profile attributes, such as certification and primary contact details are common for all party types and party usages. There are other party profile attributes that are specific to the party type:

- Organization: Includes name, organization information, certification information, legal information, and contact details.
- Person: Includes person details such as name, address, and contact information.
- Group: Includes name, group information, and group members.

Viewing Profile History: Explained

Profile history refers to the various versions of the party profile information. The profile history functionality lets you view the history of changes to profile records of the party, person or organization, in context in the party center. You can also view a party's profile records that will become effective in the future.

Viewing Profile History Information

Use the From Date and To Date filters to view the profile history for a date range. Note that the current, historical, and the future dated records are sorted by date descending. When multiple changes are done within a day use the sequence of
changes on a given date along with effective dates to determine the most recent record. The profile history information also includes changed attributes such as name, legal status, and size.

**Change Details**

The change details include multiple changes done within a day and also indicate the record that is available to be used in transactions. The old and new values of different attributes, such as name, legal status, and size, are also displayed.

You can export both the profile history information and the change details to spreadsheets.

**Party Usage Assignments: Explained**

Party Usage Assignments define if the party usage is assigned manually, or is automatically assigned by a business event. Parties can have multiple usage assignments, such as Partner or Customer. You can manage party usage assignments for a party from the Usage Assignment party tree node of the Party Center work area.

There are two types of party usage assignments:

- Conditional party usage assignments
- Unconditional party usage assignments

**Conditional Party Usage Assignments**

A conditional party usage assignment requires a business event to occur. For example, adding a person as a contact for a group creates contact party usage for the person.

You can assign rules to a conditional party usage assignment to define the business event. There are four types of rules:

- Assignment rules: Define how the party usage can and cannot be assigned to parties.
- Exclusivity rules: Restrict party usage assignment, so that between a specified date range the party usage can be the only usage allowed to be assigned to a party.
- Incompatibility rules: Specify which party usages cannot be assigned concurrently to a party during a date range.
- Transition rules: Define which party usages can transition to the party usage you are creating or editing.

**Unconditional Party Usage Assignments**

You can update unconditional party usage assignments without the need for a business event. For example, you can update the party usage for an organization from a party of interest to a customer.

**Managing Party Relationships: Explained**

You can manage complex real-life relationships among various entities such as employees, employers, contacts, and organizations in the Party Center work area. In Oracle Sales Cloud, a party relationship is a party’s role in the context of another party.

>Note: You can only view the relationships for the party in context, unless you have the manage privilege.

You must define the following for each party relationship:

- Relation types, such as employment or subsidiaries.
- Relationship role pairs, such as employee and employer.
Managing Source System References: Explained

A source system is any data source from which party and other related data can be imported. Source system references are the cross references between the source system unique identifier of a record and the unique identifier of that record in Oracle Sales Cloud.

Source System References in Oracle Sales Cloud

Organizations use multiple software applications to run their businesses. These applications are supplied by different vendors and run on different databases, which may cause the following issues:

- Duplication of records and data across systems.
- Logical representation or structure of a record may vary.
- Keeping the data in different systems synchronized, generating a consolidated view, and sharing data across systems becomes highly complex because of the difference in the logical representation.

You can resolve these issues in Oracle Sales Cloud by:

- Defining multiple external systems as source systems. While defining source systems you can specify the type of the source system. The source system can be a Spoke system, such as a legacy system, or a Purchased system, such as data from a third party provider. You can also specify which objects, such as parties only or parties and contacts, are importable from a particular source system.
- Creating records in Oracle Sales Cloud database from one or multiple source systems by:
  - Creating master records. Multiple source system references can create one master record by merging or linking multiple duplicate records imported from multiple source systems.
  - Creating records for every source system record. If you do not allow multiple source system references, then a record will be created for every source system record. This means that you could potentially create duplicate records in the database.
- Using source system references, also known as cross references, to present a consolidated view. You can maintain a central register of global identities, links to master data in source systems, lists of transformation rules, and a minor subset of information that is needed to aid in matching. At runtime, you can access the source systems’ master data and assemble a point-in-time consolidated view.

Managing Source Systems

In Oracle Sales Cloud, you can:

- View source systems contributing to the party in the context from the Referenced Source Systems table.
• View source system identifiers information for the child entities of a party for the source selected from the Referenced Source Systems table.

• Delete source system identifiers to change the status of the source system reference to inactive.

• Add source system identifiers to include a new source system reference.

**Note:** The source system, source system reference identifier, and the start date cannot be updated. To change a source system reference identifier, delete the current record and add a new source system reference a new source identifier value.
15 Managing Hierarchy Information

Manage Hierarchy Information

Party Hierarchy: Explained

A party hierarchy captures the hierarchical relationships a party has with other parties. This capability is frequently used to capture your customer’s corporate hierarchy and to show how headquarters, branches, subsidiaries, and so on are related. It can be used to capture the hierarchy of any party, and not just of customers.

Organizing customer data as party hierarchies offers the following advantages:

- Enables you to understand and get a better view of your customer’s organization.
- Enables you to better analyze customer-related data.
- Application modules using the party hierarchy can use hierarchies to roll up transactions, and apply business rules.

You can leverage the party hierarchy of your customers in many business processes. For example, the territory management functionality uses customer hierarchy information to define account dimensions. Financial applications use customer hierarchy information to process payments from one customer and apply them to another customer in the same hierarchy. The revenue roll-up report uses customer hierarchy information to roll up revenue numbers from opportunities across all customers in a hierarchy.

Oracle Sales Cloud has a common framework to manage various hierarchies called the Tree framework. The party hierarchy uses this common framework. You can create or modify party hierarchies, using the Manage Hierarchies task.

Hierarchies and Hierarchy Versions: How They Fit Together

While creating hierarchies, you have to decide whether you want to create a new hierarchy or use a new version of an existing hierarchy.

When to create a new hierarchy

You can create a new hierarchy when any of the following is true:

- You have a new customer, and a new hierarchy is required to represent the corporate structure of the new customer.
- Your existing customer has changed the structure of its organization radically. This is quicker and more efficient to create a new version rather than edit the existing one.

When to create a new hierarchy version

You can create a new version of a hierarchy when any of the following is true:

- You have minor changes to make to an existing version of a hierarchy, such as adding a new customer, removing or repositioning an existing customer in the hierarchy.
- You must make extensive changes to an active hierarchy, but want to render the changes active only when they are all incorporated into the hierarchy. Then, create a new version of the hierarchy and set it to become active after a
Hierarchical Edits and Hierarchy Versions: How They Work Together

While editing a hierarchy, you must decide whether you want to edit the existing version of the hierarchy or create a new one. The changes you make to the existing version become active immediately, but the new versions of a hierarchy become active only on the date on which the hierarchy was active.

Editing Party Hierarchies

You must edit the hierarchy versions to change details of an existing hierarchy version. When you edit a hierarchy, the changes are saved instantly and are available to users even as you edit them. However, you can't cancel any changes while editing hierarchy versions.

Two ways to deal with the changes while editing active hierarchies are:

- You can render hierarchies inactive before you edit them. Once you have made your changes, you can change their status to active again.

  Note: The hierarchy is not available to users while it is inactive.

- You can create a new hierarchy version and set it to be active the next day. Thus, while the existing version is active, you can make your changes to the new version and activate it the next day.

Editing Party Hierarchies in Response to Merge

The party merge process merges the duplicate parties and updates the hierarchies to which these parties belong. If the updates to the hierarchies done by the party merge process are not as expected, then edit the updated hierarchies in the Manage Hierarchy Types task.

Related Topics

- Merging Duplicate Records: Explained

FAQs for Manage Hierarchy Information

How can I create a new hierarchy type?

You can create a new hierarchy type using the Manager Tree Structures task. Select Trading Community Model as the application, an existing data source or new custom data source, and any labeling scheme you want. Next, enter the other required details such as the appropriate usage of the new hierarchy type. Add the newly created hierarchy type under the Party Hierarchy Type lookup to include the new hierarchy type in the lookups for hierarchy types.
Can I create multiple versions of the same hierarchy?
Yes. You must ensure that the date on which the new version is active doesn’t overlap with the date of any existing active version of the same hierarchy.

How can I respond to a merge completion notification if I don't have the privileges to edit hierarchies?
If you don’t have the privileges to edit hierarchies, reassign the merge completion notification to a user with the appropriate privileges. For example, a sales person can reassign a merge completion notification to the Sales Administrator.

Why is the hierarchy type I created not visible in the Party Hierarchy UI?
When you create a hierarchy type, you must also update the list of hierarchy lookups. You cannot use the new hierarchy type, unless you have updated the hierarchy lookup list.
You can update the list of hierarchy types using the Manage Hierarchy Lookups task. Select Party Hierarchy Type (PARTY_HIERARCHY_TYPE) lookup type to edit. Add the new hierarchy type to the Party Hierarchy Type lookup type. After updating the list of hierarchy type lookups and saving your changes, you can view the new hierarchy type under the list of hierarchy types.

Why is an active hierarchy version that has been saved recently sometimes set automatically to inactive?
By default, the newly edited hierarchy is saved as an auto-commit process, even before you save and close it. The application triggers the tree flattening service and the audit job, when you save and close your changes. It doesn't wait for the job to complete, and notifies you to revisit the version after the audit job is complete, to check the version status of the hierarchy.
If the audit job identifies any errors at the time of hierarchy validation, then the status of the hierarchy is set to Inactive. However, the application does not notify you of this. You must revisit the hierarchy management interface and rectify issues in the previously saved version, if the status is inactive.
16 Managing Enterprise Resource Information

Manage Resource Information

Understanding Sales Resources

Sales resources are users who participate in Oracle Sales Cloud business processes. You must create or import resources before you can associate them with resource organizations, resource teams, or work objects.

You can enter your resource information using the Setup and Maintenance work area, Manage Resources task. Alternatively, you can import data to create or update employee resources.

Related Concepts

You must understand the following concepts to understand resources in Sales Cloud.

- **Resource team**: A resource team is a temporary group of resources formed to complete a business task. A resource team cannot be hierarchically structured and is not intended to implement an organization.
- **Resource Role**: You use the resource roles to define provisioning rules. For example, a provisioning rule can assign the Sales Manager job role to a user with the Sales Manager resource role.
- **Resource Directory**: The Resource Directory offers detailed information about all the resources within the deploying organization. The Resource Directory also enables you to find and communicate with other resources, and to network and collaborate with them.
- **Resource Organization**: You create sales resources and provision permissions the resources need to do their jobs. In the process, you also build the organization chart of your sales organization.

Setting Up Resources

You create sales resources and provision permissions the resources need to do their jobs. In the process, you also build the organization chart of your sales organization.

To set up resources, you must:

- Identify an existing employee, contingent worker, or partner member as a resource.
- Specify the end date for a resource’s engagement with the deploying company.
- Assign resource roles to the resources. You can define resource roles using the Manage Resource Roles task from the Setup and Maintenance work area.
- Assign resources to organizations. Before assigning resources to organizations, you must ensure that the organization hierarchy for your sales organization is built. You can define organizations using the Manage Internal Resource Organizations task from the Setup and Maintenance work area.
- Assign resources to teams. You can include resources from different resource organizations or an entire organization to work as members of a resource team. You can assign identified resources to teams and assign them roles within the team.
Managing Resources: Explained

This topic provides information about managing resources.

Managing resources involves viewing and modifying a resource’s profile, organization membership, role assignment, skills, additional contact information, and salesperson information.

All the resources within the deploying company can view any resource’s profile. However, only the administrator can modify a resource’s profile, organization and team membership, and role assignment. Managers can modify the skill information of their direct and indirect reports. However, resources can only modify their own profiles, skills, and additional contact information.

Managing Resource Profiles

Resource profile management involves managing a user’s profile, including the resource’s core skills, photo, time zone, additional contact information, and so on. As part of managing a resource’s profile, the administrator specifies the dates between which the resource is available to the organization. The administrator also ensures that role assigned to the resource is within the period mentioned in the profile.

Managing Organization Membership

At any point in time, a resource belongs to an organization, and the administrator can assign this resource to any other organization within the deploying company.

Managing Resource Role Assignment

The administrator or the organization manager can assign or modify resource roles within an organization, with one role assigned to a resource at a time. However, administrators can also assign multiple roles to resources outside an organization.

Note:

1. The dates of the resource role assignment must be within the date range during which the resource is active in the deploying organization.
2. An organization can have only one manager.

Resource Directory: Explained

This topic explains the Resource Directory and how you can use it.

The Resource Directory offers detailed information about all the resources within the deploying organization. The Resource Directory also enables you to find and communicate with other resources, and to network and collaborate with them.

You use the Resource Directory to perform the following tasks:

- View and modify your profile
- View your organization and team membership information
- View information related to other organizations and teams
- View the profiles of other resources
- Communicate with other resources
Manage Contact Preference Information: Explained

Managing contact preference information includes creating and editing preferences about contact permissions and restrictions.

You can manage contact preferences on the customer’s Edit Contacts page in the classic interface by expanding the Contact Points region of the customer’s contact and selecting Manage Contact Preferences from the regional Action menu.

Creating Contact Preference Information

When you are viewing Address or Contact Point information for a customer or contact, you can select a specific address or contact point, and choose Manage Contact Preferences from the Action menu. You capture whether there is a restriction (Do not) or permission (Do) in the Preference attribute, and a Reason Code for such preference. You record a specific start date and can set an end date for the preference. As preconfigured, the start date is the current date, and the end date is null.

Reviewing Contact Preference Information

On seeing the Do Not Contact icon, you must review contact preference information for restrictions before taking any action. You can review the contact restriction information by clicking on the Do Not Contact icon or on the appropriate option from the action menu. Note that do-not-contact entries are made against each phone, e-mail, and address and not at the organization or person level. If restrictions are present for a phone number, the CTI action is disabled.

Privileges Required for Managing Contact Restriction Information

Contact restriction information, such as opting in or out of the Public Do Not Call Registry, is captured as a Reason Code. Regular business users, such as salespeople and managers, can create and edit contact preference information with any Reason Code that is not identified as Legal. However, to be able to create and edit contact restriction information using a Reason Code that is tagged as Legal, you must add the HZ_LEGAL_CONTACT_PREFERENCES_PRIV privilege to the required role.

A Reason Code can be setup as Legal by tagging the Reason Code lookup value in the lookup type REASON_CODE with the value LEGAL using Manage Trading Community Common Lookups task.

Updating Resource Skills: Examples

This example illustrates the various contexts in which you can update your skills in Oracle Resource Management.

You can add a wide range of skills into Oracle Resource Management. For the sake of convenience, let us categorize these into core skills, achievements, and mandatory requirements.

Each of these skills must be used in specific contexts as described below.

Updating Core Skills

Skills that reflect your key abilities are categorized under the Core Skills head, such as competencies, degrees, languages, licenses and certifications, skills, and problem codes.

- Use the Competencies option to record your key abilities as a resource. For instance, if you have the ability to quickly internalize and analyze complex information, list it here.
- Use the Degrees option to record your educational degrees.
- Use the Language option to record your proficiency with the various languages you know.
• Use the Licenses and Certifications option to record any educational licenses and certifications you may have received. For instance, if you have a Six Sigma certification, list it here.

• Use the Skills option to list out any specific software or engineering skills you may have. You can search for these skills by category, product, platform, or component.

• Use the Problem Code option to record your ability to manage and remedy specific service-related problems that may arise within your company. For instance, if you are skilled at troubleshooting hardware issues, list it here.

Updating Achievements
Skills that reflect your key achievements and uncommon abilities are categorized under the achievements head, such as accomplishments, honors, awards.

• Use the Accomplishments option to record any major task you completed that reflects your abilities as an individual. For instance, if you completed a rather difficult task in a short deadline, list it here.

• Use the Honors and Awards option to record any special honors or awards you may have received. For instance, if you were the recipient of a certificate of merit award, list it here.

Updating Mandatory Requirements
Skills that capture the mandatory requirements, which you have met for your employment, are categorized under the mandatory requirements head. Thus, job requirements and work requirements are part of mandatory requirements.

• Use the Job Requirements option to update all job requirements you have met. For instance, many companies require you to take a medical test before you join. If you took such a test before you joined the company, list it here.

• Many roles have specific requirements that employees must fulfill before these roles can be assigned to them. Such requirements can be listed out under the Work Requirements head. For instance, if you are willing to travel overseas on duty, list it here.

FAQs for Manage Resource Information

What's the Resource Directory?
The Resource Directory offers a comprehensive view of resources in the deploying organization. You can use the Resource Directory to find other resources, and to network and collaborate with other resources in the system.

Resources can access the Resource Directory to view and modify their own resource information and the information of their direct reports. Resources can also use the Resource Directory to view their organization and team membership details.

What's a salesperson set?
Sales person sets enable you to assign sales person to a specific group or area. For instance, you could have sales person sets that organize your sales teams based on their region of operation. Thus, you could have separate sales person sets for sales teams working with customers in the Americas, Europe, Asia, etc.

Use sales person sets to control access to sensitive data within the deploying company. For instance, only members of the sales person set that comprises sales persons working out of Asia need access to data like addresses and phone numbers of customers in the Asian region.
Can I assign multiple concurrent resource roles to an organization member?

No. While you can assign multiple roles to an organization or team member, the roles cannot be concurrent. This means that at any point in time, a resource can have only one role in an organization.

How can I search and view information from other sources?

You can search for resources and view their public profiles using the Resource Directory. You can search for resources either by name or organization, or by the names of the teams to which they belong.

The Resource Directory enables you to view the following details of a resource’s profile:

- Key profile information, including the resource’s name, manager’s name, job title, contact information, etc.
- Organization membership information
- Team membership information
- Roles of the resource
- Key skills of the resource
- Additional contact information
- Salesperson details, if applicable.

You can search for organizations or teams, instead of searching for users. Then, you can view the list of resources that belong to an organization or team, and view the profiles of any resource listed.

Can I add resources outside my reporting hierarchy to my social network?

Yes. Use the Resource Directory to view the public profile of all active resources within the deploying company and to add them to your social network.

You can add a resource to your social network using the Contextual Action interface. Search for the resource with whom you want to make a connection, view the resource’s detailed profile, and then add the resource as a connection. Once the resource accepts your connection request, the resource becomes a part of your social network.

Manage Resource Organization Information

Resource Organizations and Organization Usage: Explained

You can assign organization usage information to resource organizations to classify them based on how on how you want to use them. For instance, you can assign resource organizations engaged in sales activities to Sales Organization usage. This enables you to sort organizations based on their usage, simplifying your task of working with them.
Resource Role and Resource Organization Components: How They Work Together

This topic explains how resource roles and resource organizations work together.

A single resource can belong to only one resource organization and be associated with only a single resource role, during a given period. The organizations and organization hierarchies that a resource belongs to determine the reporting relationships for that resource.

Resource Roles

A resource role denotes the function of a resource in an enterprise from the perspective of the deploying company. A resource role not only describes who a resource is in the enterprise, but also specifies the role a resource performs within the context of an organization or team. A resource can have only one role in a resource organization during a given period.

Note: You can assign only one role to a resource at a time.

Resource Organizations

A resource organization represents the internal organization and structure of the deploying company. Resource organizations are hierarchically structured, and the organization hierarchy helps to derive the reporting relationships. A resource can be a part of different organizations within the deploying company, and can have separate roles in each.

FAQs for Manage Resource Organization Information

Why am I unable to view the resource hierarchy for my organization?

For a resource or reporting hierarchy to be visible, it is necessary to have a reporting structure enabled in your organization. If there is no manager defined for an organization, the reporting hierarchy of the organization cannot be constructed. Please check if your organization has been assigned a manager in Resource Management.

What's the difference between a resource organization and a resource team?

A resource organization is an organization whose members are resources. Resource organizations are used to implement sales organizations, partner organizations, and so on.

A resource team is a temporary group of resources formed to work on work objects. A resource team may contain a resource organization or resources or both. A resource team cannot be hierarchically structured and is not intended to implement an organization.
Can I remove a resource organization membership while active assignments exist?

Yes. Once you delete a resource organization membership, the application checks if the concerned resource has any active assignments within the organization. It displays a warning message whether you want to remove the membership of the active assignments. Once you confirm to delete, all the active assignments and the resource organization membership are removed.

Can I update organization membership details?

Yes. To update the membership details of an organization, search and click the organization whose membership you want to update. Click the Members tab to display a list of the members of the organization. Click Create to add more members, click Edit to update the roles of existing members, or click Delete to remove members from the resource organization.

Can I view a resource organization hierarchy version as of a particular date?

Yes. Unless otherwise specified, organization hierarchies are active from the date they are created. Navigate to the View Organization Hierarchies page to view the hierarchy of a resource organization at a specific date. Search and select the organization hierarchy whose details you want to view and this displays the concerned organization’s active hierarchy. A drop-down list at the top of the screen displays the date from which the current organization hierarchy has been active. Click the list and, from the options available, select the date you want. This displays the version of the organization hierarchy for the date you selected.

Manage Resource Team Information

Resource Members, Role Assignment Dates, and Active Dates: How They Work Together

Every resource in the deploying company has start and end dates for a specific activity. When you assign resources to teams, you must ensure that their date of assignment is within the end date.

Resource Active Dates and Resource Assignment

When you identify internal or partner employees as resources, you can specify the time for which they are active within the organization. By default, the date you identify a resource becomes the start date, but there is no end date listed. Thus, an identified resource is active from the day of activation and you can assign the resource to any organization or team for an infinite period. However, once you specify an end date for a resource, you cannot assign the resource to any team or organization during the resource’s active period.
Resource Skills and Resource Assignment

To get the best out of resources, you must ensure that their skills are relevant to the team or organization within which you deployed them. Determine whether a resource’s skill set matches the requirements of an organization or team. Then, deploy the resource as appropriate for a period of time that falls within the active dates specified for the concerned resource.

FAQs for Manage Resource Team Information

What's a resource team?

A resource team is a group of resources formed to work on work objects. A resource team can comprise resource organizations, resources, or both. A resource team is neither hierarchically structured nor intended to implement an organization structure. You can use resource teams as a quick reference to groups of related resources to which you can quickly assign work objects.

Note: You can either individually assign the members of a team to a task or assign entire teams to tasks.

Can I assign multiple resource roles to a team member at the same time?

Yes. Resources within resource teams can have multiple resource roles. You can add roles to a resource in a resource team using the Manage Resource Teams task. Select the resource and click the Edit button to assign additional roles to the resource.

Perform Mass Transfer

Transferring Records Between Users: Explained

Mass Transfer of records lets you move records from one user to other. A record owner or any user above the owner in the role or territory hierarchy can transfer records from one user to the other.

Before transferring records, you must understand:

- Record Types
- Record Filters
- Record Transfer Status

Record Types

Record types are broad categories of objects or information related to a user. For example, Deal Registrations associated with a user. Currently, you can mass transfer Leads, Opportunities, and Deal Registrations that belong to a user.
Record Filters

Record Filters let you refine the list of records associated with the user for a record type. For example, you can transfer only deal registrations that were created during a time period. You can't specify filters for all record type. The Transfer Records: Define Filters page lets you view the record types that allow filtering, and specify the record filters.

Record Transfer Status

Record transfer statuses appear in the Mass Transfer Status page and show the status of transfer jobs. The Mass Transfer processes records to ensure data integrity before transferring the records from a user to the other.

A Record Transfer job can have one of these statuses:

- In Progress: Transfer job is currently in process.
- Completed: Transfer job has been completed without errors.
- Errors: Transfer job has resulted in an error.

You can click on the transfer job name to view the record types that were transferred, the status of each record type, and the log file associated with a record type.

Transferring Records Between Users: Procedure

This procedure describes how you can transfer records from one user to the other using the Mass Transfer tool.

To transfer records from one user to another:

1. Navigate to Mass Transfer from the Tools menu.
2. In the Mass Transfer Status page, click Transfer Records.
3. In the Transfer Records: Select Owners and Records page, search for the current owner and the new owner of the records. For example, if you are transferring from Adam Smith to Samantha Hayes, then you must select Adam Smith as the current owner and Samantha Hayes as the new owner.
4. Select the types of records you want to transfer. For example, if you are transferring opportunities and leads, then you must select Opportunities and Sales Leads.

The Transfer Details column lists the types of records that will be transferred.

5. Click Next.

In the Transfer Records: Define Filters page, you specify filters for record types you have selected. For example, you can specify the start and close dates for opportunities to transfer the opportunities that were closed during a specified period of time.

6. Select a record type to view the filters available, and specify the filters.
7. Click Submit.
8. Click Yes in the confirmation dialog box.

The Mass Transfer Status page lists the recent mass transfer jobs and their statuses.
Securing and protecting customer information against data breaches, data theft, or unauthorized access is an increasing concern in any enterprise. To address this issue, Oracle Sales Cloud provides restricted access to information that is considered private to an individual, also known as Personally Identifiable Information (PII).

PII is any information that uniquely identifies an individual, such as personal phone number, personal e-mail address, personal address, citizenship number, or Social Security Number (SSN). PII data can be used to uniquely identify, contact, or locate an individual, or can be used with other sources to uniquely identify a person. For example, a SSN uniquely and directly identifies an individual, whereas a telephone area code identifies a set of people.

The attributes that are identified as PII are as follows:

- Home Address
- Home Phone Number
- Personal Email Address
- Taxpayer Identification Number (Social Security Number)

In Oracle Sales Cloud, access to the PII data is restricted only to the Sales Administrator job role. If you need to work with PII data for business purposes, then contact the IT security manager for the necessary privileges.
18 Viewing Audit Histories

Audit History: Explained

Using audit history you can view changes to the application data such as the business objects that were created, updated, and deleted. To view the history or to create a report, you must have a role with the assigned privilege View Audit History (FND_VIEW_AUDIT_HISTORY_PRIV). For appropriate assignment of roles and privileges, check with your security administrator.

To open the Audit History work area, from the Navigator menu, select Audit Reports.

The default search displays a summary of the audit history in the search results table. It includes key data such as date, user, event type, business object type, and description. For a detailed report, search again with modified search criteria. You can export the report summary to Microsoft Excel.

The following table lists the search parameters used and the outcome of their selection in the detailed report.

<table>
<thead>
<tr>
<th>Search Parameter</th>
<th>Result of Selection</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business Object Type</td>
<td>• Narrows the search results to that specific business object within the selected product. • Enables the Show Attribute Details check box.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> This parameter is applicable only for the business objects that belong to Oracle Applications Cloud.</td>
</tr>
<tr>
<td>Include Child Objects</td>
<td>Displays all the child objects that were listed under the business object when audit was set up. For example, a sales order object that contains several items as child objects.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> Displays the objects at the immediate parent-child level only. To view the children at subsequent levels, select the child object as the business object type and search again.</td>
</tr>
<tr>
<td>Show Attribute Details</td>
<td>Enables the attribute list so that users can select either all attributes or a specific attribute to view the changes. Based on the selection, the search results indicate whether the attribute is created, updated or deleted, and the corresponding old and replaced values.</td>
</tr>
<tr>
<td>Show Extended Object Identifier Columns</td>
<td>Displays the instances (contexts) in which the business object was used. The context values identify the objects and the transactions in which they were used. Each context is unique and assigns a unique description to the business object.</td>
</tr>
</tbody>
</table>

**Note:** The default report displays a standard set of columns that contain prominent details of the audit history. To view additional details, you can customize the display of columns.
Audit Event Types: Explained

Events trigger the audit process and the event details are stored in the audit table. Along with audit history, you can view which events triggered the changes to the object data within a given period of time. However, for events to trigger the audit process for the business objects, the objects must be enabled for auditing.

For Oracle Applications Cloud, transactional events trigger the audit process based on the create, update, and delete operations. The attributes of the audited object data can be used in lookups and foreign keys to display values in the audit history. Oracle Applications Cloud supports displaying the audited information as per the application’s language settings.

For Oracle Fusion Middleware products, events such as failed sign in attempts are used as triggers. In absence of an audit table, the information is stored in the sandbox and presented only in English.

>Note: This topic focuses on the create, update, and delete event types available in Oracle Applications Cloud. For details on events used in the Oracle Fusion Middleware products, refer to the Oracle Fusion Middleware documentation.

- Create Operation - This event triggers audit activity whenever business object data is created and stored in the database. The audit report displays the name of the created object, the name of the user who created the object, and the time stamp.

- Update Operation - This event triggers audit activity whenever the existing business object data is updated. The audit report displays both the old and updated values of the object, the name of the user who updated it, and the time stamp.

- Delete Operation - This event triggers audit activity whenever the existing business object data is removed from the database. The audit report displays the last value that the object attributes contained and the name of the user who deleted the object.
**data cleansing**
Identifying and correcting incomplete, corrupt, or inaccurate records such as addresses.

**enterprise**
An organization with one or more legal entities under common control.

**external system or external application**
A system or application that is external to and not part of Order Management. An order capture system that resides upstream of Order Management is an example of an external system. A fulfillment application that resides downstream of Order Management is an example of an external application.

**interface table**
A database table that stores data during data transfer between applications or from an external system or data file.

**organization**
A unit of an enterprise that provides a framework for performing legal, managerial, and financial control and reporting. Organizations can be classified to define their purpose, for example, as departments, divisions, legal entities, and can own projects and tasks, or incur project expenses.

**organization hierarchy**
A tree structure that determines the relationship between organizations.

**resource**
People designated as able to be assigned to work objects, for example, service agents, sales managers, or partner contacts. A sales manager and partner contact can be assigned to work on a lead or opportunity. A service agent can be assigned to a service request.

**resource organization**
An organization whose members are resources. Resource organizations are used to implement sales organizations, partner organizations, and so on.

**resource role**
Resource roles indicate the role a resource plays as an individual, or within a resource team.

**resource skills**
A resource skill is a self-proclaimed, self-rated knowledge set that a resource has. Skills are defined in terms of categories, products, components and platforms.
resource team
A resource team is a temporary group of resources formed to work on work objects. A resource team may contain a resource organization or resources or both. A resource team cannot be hierarchically structured and is not intended to implement an organization.

simulated cleansing
A mode for batch cleansing in which the results are available for preview before they are saved in the database.