Oracle® Cloud

Sync Oracle NetSuite Customers with Oracle CX Sales and B2B Service Accounts

G13890-01
September 2024

ORACLE®

Contents

Preface

This document describes how to install, configure, and run this recipe in Oracle Integration 3.

Topics:

- Documentation Accessibility
- Diversity and Inclusion
- Related Resources
- Conventions

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc`.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info` or visit `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs` if you are hearing impaired.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Related Resources

For more information, see these Oracle resources:

- Oracle Integration documentation on the Oracle Help Center.
- Oracle Cloud at `http://cloud.oracle.com`.

Conventions

The following text conventions are used in this document.

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.

Convention	Meaning
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1
About This Recipe

Use this recipe to synchronize Oracle NetSuite customers with Oracle CX Sales and B2B Service accounts.

> **Note:**
>
> This recipe is available as **Oracle NetSuite — Oracle CX Sales and B2B Service | Sync Customers and Accounts** in Oracle Integration. Oracle provides this recipe as a sample only. The recipe is meant only for guidance, and is not warranted to be error-free. No support is provided for this recipe.

Overview

This recipe creates or updates a corresponding Oracle CX Sales and B2B Service account when a customer is created or updated in Oracle NetSuite. Additionally, whenever an account is created or updated in Oracle CX Sales and B2B Service, the recipe creates or updates a corresponding customer in Oracle NetSuite.

To use the recipe, you must install the recipe and configure the connection and other resources within it. The recipe contains three integration flows - **Oracle NetSuite CX Sales Customer Upsert** , **Oracle CX Sales NetSuite Customer Create** and **Oracle CX Sales NetSuite Customer Update**.

- When a new customer is created or a customer record is updated in Oracle NetSuite, the user event script associated with the customer gets triggered in NetSuite. The user event script provides the JSON data of the customer and subsequently triggers the **Oracle NetSuite CX Sales Customer Upsert** integration flow in Oracle Integration. The integration then creates or updates a corresponding account record in Oracle CX Sales and B2B Service.

- When a new account is created in Oracle CX Sales and B2B Service, the **Oracle CX Sales NetSuite Customer Create** integration flow gets triggered in Oracle Integration. The integration creates a corresponding customer record in Oracle NetSuite.

- When an account is updated in Oracle CX Sales and B2B Service, the **Oracle CX Sales NetSuite Customer Update** integration flow gets triggered in Oracle Integration. The integration then updates the corresponding customer record in Oracle NetSuite.

Basic data such as customer/account name, phone, email, company name and address are synchronized between Oracle NetSuite and Oracle CX Sales and B2B Service platforms.

System and Access Requirements

- Oracle Integration, Version 24.10 or higher
- Oracle NetSuite
- An account on Oracle NetSuite with the Administrator role

- Oracle CX Sales and B2B Service
- An account on Oracle CX Sales and B2B Service with the Administrator role

2

Before You Install the Recipe

You must perform the following configuration tasks on your Oracle NetSuite and Oracle CX Sales and B2B Service instances in order to successfully connect to these external systems using Oracle Integration, and synchronize customers and accounts.

Configure Oracle NetSuite

To successfully connect to Oracle NetSuite using Oracle Integration, you must perform certain configurations on your Oracle NetSuite instance and some configurations specific to the security policy you'll use to access. In this recipe, you'll use the token-based authentication (TBA) security policy.

Log in to your Oracle NetSuite instance as an **Administrator** and execute the following tasks.

1. Perform the general configurations necessary to connect to Oracle NetSuite. See Enable Features on Oracle NetSuite and Assemble the Oracle NetSuite WSDL URL.

2. Perform the TBA-related configuration tasks. See Prerequisites for the Token-Based Authentication Security Policy.

   Grant the following permissions to the role you'll create in this step.

Tab Name	Permission	Level
List	Currency	Full
List	Customers	Full
List	Subsidiaries	Full
List	Contacts	Full
Setup	User Access Tokens	Full
Setup	Log in using Access Tokens	Full
Setup	SOAP Web Services	Full

3. Create a custom field for customer record. See Creating Custom Transaction Body Fields. While creating a custom field, enter the following values.

   a. Specify the value for **Label** as `CxSalesPartyNumber` .

   b. Specify the value for **ID** as `id_cxsalespartynumber`.

   c. In the **Applies To** tab, select **Customer**.

   d. Click **Save**.

4. Configure and deploy the User Event script.

   a. Download the sample user event script: NS-CXSales.js.

   b. Open the script in a text editor such as Notepad++ and make the following updates:

      i. On **line 30**, there's a variable with the value set to `***your admin user email***`. Replace this value with the email address of the integration user that you created. The end result should look like this: `var integrationUserEmail = "john@netsuite.com"`.

    ii. On **line 85**, there's a variable with the value set to `*The base URL of your Oracle Integration instance*`. Replace this value with the domain name of your Oracle Integration instance. Make sure to omit everything after the `.com` top-level domain.

    iii. On **line 83**, there's a variable with the value set to your username. Replace this value with the username of your Oracle Integration instance.

    iv. On **line 84**, there's a variable with the value set to your password. Replace this value with the password of your Oracle Integration instance.

    v. Save the changes made to the script.

c. In NetSuite, navigate to **Documents**, then **Files** and then **SuiteScripts**.

d. Within the SuiteScripts folder, create a new folder. For example: `Stripe Integration`.

e. Upload the script that you modified and saved in step b to the newly created folder.

f. Go to **Customization**, then **Scripting**, and then **Scripts**.

g. In the **Script File** field, start typing `Oracle_NS_Stripe_Customer_Create.js`, and select the auto-completed option.

h. Click **Create Script Record**.

i. Set the **Name** field to `Oracle NS Stripe Customer Script` and **ID** to `_oracle_stripe_cust`.

j. Click **Save** and then click **Deploy Script**.

k. Set the **Applies To** field to `Customer` and the **ID** to `_oracle_stripe_cust_depl`.

l. Set the **Status** to **Deployed**.

m. In the **Audience** tab, select all check boxes where applicable or select all items in the respective lists.

n. Click **Save**.

Configure Oracle CX Sales and B2B Service

To successfully connect to Oracle CX Sales and B2B Service using Oracle Integration, you must perform certain configurations on your Oracle CX Sales and B2B Service instance.

1. Create an integration user account.

To invoke an Oracle CX Sales and B2B Service service catalog or event catalog web service from , you create a separate user.

a. Log in to the Oracle CX Sales and B2B Service with a user with system administrator privileges.

b. Go to **Navigator** > **My Team** > **Manage Users**.

c. In the Manage Users page, click **Manage Users**.

d. Click the **Create New User** icon beside **Show Photo**.

e. Enter the following information, and click **Save**.

Field	Description
Last Name	Enter `FUSION_APPS_ICS_APPID`
Email	Enter a valid email address.

Field	Description
Hire Date	Enter the date.
User Name	Enter `FUSION_APPS_ICS_APPID`.
Person Type	Enter `Employee`.
Legal Employer	Select a valid legal organization.
Business Unit	Select a valid business unit.
Send user name and password	Select this checkbox.

A notification email is sent to the email address after the user is created.

f. Log out of the Oracle CX Sales and B2B Service.

g. Log in to the Oracle CX Sales and B2B Service instance with FUSION_APPS_ICS_APPID and the temporary password provided in the notification email.

h. Change the password when prompted at the first log in.

The Oracle CX Sales and B2B Service welcome page appears.

i. Log out of the Oracle CX Sales and B2B Service.

2. Assign integration roles.

Use the Oracle Security Console to assign the integration user with certain roles and privileges.

> **Note:**
>
> Access to the Security Console is provided by the predefined **Security Manager** role.

a. Select **Navigator** > **Tools** > **Security Console**.

b. On the top right corner of the window, click **Create Role**.

c. In the Create Role: Basic Information page, create a new record with the following information and click **Next**.

Parameter	Value
Role Name	OIC Integration Role
Role Code	INT_ OIC_ Integration_ Role
Role Category	CRM - Job Roles
Description	Custom Role for Accessing OSC Services Catalog

d. In the Create Role: Functional Security Policies page, click **Add Functional Security Policy**.

e. In the Add Function Security Policy page, enter `FND_MANAGE_CATALOG_SERVICE_PRIV` in the Search box and click **Add Privilege to Role**. Click **Next**

f. In the Create Role: Data Security Policies page, click **Next**.

g. Add the **Sales Administrator** and the **SOA operator** roles, and then click **Next**.

To add the **Sales Administrator** role, do the following on the Create Role: Role Hierarchy page:

      i.   Click **Add role**.

      ii.   In the Search field, enter `Sales Admin`.

      iii. Select the **Sales Administrator** role, and then click **Add Role Membership**.

      iv. Close the Add Role Membership window.

To add the **SOA Operator** role, do the following on the Create Role: Role Hierarchy page:

      i.   Click **Add role**.

      ii.   In the Search field, enter `SOA Operator`.

      iii. Select the **SOA Operator** role, and then click **Add Role Membership**.

      iv. Close the Add Role Membership window.

h. Assign the integration user to the roles.

On the Create Role: Users page, click **Add user** and then do the following:

      i.   In the Search field, enter `FUSION_APPS_ICS_APPID`.

      ii.   Choose the **FUSION_APPS_ICS_APPID** user and then click **Add user to Role**.

      iii. Close the Add user window.

i. Click **Next**, review the details on the Summary and Impact page, and click **Save and Close**.

3

Install and Configure the Recipe

On your Oracle Integration instance, install the recipe to deploy and configure the integration and associated resources.

1. On the Oracle Integration Home page, in the **Get started** section, click **Browse store**.

2. Find the recipe you want to install, then click **Get**.

   A message confirms that the recipe was successfully installed, and the recipe card shows **In use**.

3. Click **Configure** on the recipe to configure its resources.

   The project workspace opens, displaying all the resources of the recipe. Configure the following resources before you activate and run the recipe.

Configure the Oracle NetSuite Connection

1. In the Connections section, click the connection name.

2. In the Properties section, enter the Oracle NetSuite WSDL URL. For example: `https://webservices.netsuite.com/wsdl/v2022_1_0/netsuite.wsdl`. See Configure Oracle NetSuite.

3. In the Security section, enter the following details:

Field	Information to Enter
Security Policy	Select **Token Based Authentication**.
Consumer Key	Enter the consumer key of the integration record in Oracle NetSuite. See Configure Oracle NetSuite.
Consumer Secret	Enter the consumer secret of the integration record in Oracle NetSuite.
Token	Enter the token ID provided by Oracle NetSuite.
Token Secret	Enter the token secret provided by Oracle NetSuite.
Account ID	Enter your Oracle NetSuite account identifier.

4. Click **Save**. If prompted, click **Save** again.

5. Click **Test** to ensure that your connection is successfully configured. In the resulting dialog, click **Test** again.

   A message confirms if your test is successful.

6. To return to the project workspace, click **Go back** .

Configure the REST Trigger Connection

1. In the Connections section, click the connection name.

2. Click **Test** to ensure that your connection is successfully configured. In the resulting dialog, click **Test** again.

   A message confirms if your test is successful.

3. Click **Save**. If prompted, click **Save** again.

4. To return to the project workspace, click **Go back** .

Configure the Oracle CX Sales and B2B Service Connection

1. In the Connections section, click the connection name.

2. In the Properties section, enter the Oracle CX Sales and B2B Service host name. For example: `https://your_domain_name.fa.DC.oraclecloud.com`.

3. In the Security section, enter the following details:

Field	Information to Enter
Security Policy	Select **Username Password Token**.
Username	Enter the username of the account created for Oracle Integration on the Oracle CX Sales and B2B Service instance. See Configure Oracle CX Sales and B2B Service.
Password	Enter the password of the account created for Oracle Integration on the Oracle CX Sales and B2B Service instance.

4. Click **Save**. If prompted, click **Save** again.

5. Click **Test** to ensure that your connection is successfully configured. In the resulting dialog, click **Test** again.

   A message confirms if your test is successful.

6. To return to the project workspace, click **Go back** .

Configure the Lookup Table

Edit the **ORACLE-BRT-EC_NS_SETTINGS** lookup table.

1. In the Lookups section, click the lookup name.

2. Edit lookup keys in the **ECCode** and **NSCode** columns and map NetSuite values with Oracle CX Sales and B2B values.

3. Click **Save**. If prompted, click **Save** again.

4. To return to the project workspace, click **Go back** .

4
Activate and Run the Recipe

After you've configured the connections and other resources, you can activate and run the recipe.

1. In the project workspace, click **Activate**. In the Activate project panel, with the default project deployment selected, choose an appropriate tracing option, then click **Activate**.

   A message confirms that the integrations have been activated. Refresh the page to view the updated status of the integrations.

2. Run the recipe.

   a. Log in to your Oracle NetSuite instance and create a new customer or update a customer.

      This triggers the **Oracle NetSuite CX Sales Customer Upsert** integration flow which subsequently creates or updates a corresponding account in Oracle CX Sales and B2B Service.

   b. Log in to your Oracle CX Sales and B2B Service instance and create a new account.

      This triggers the **Oracle CX Sales NetSuite Customer Create** integration flow which creates a corresponding customer in Oracle NetSuite.

   c. Now, update an account in your Oracle CX Sales and B2B Service instance.

      This triggers the **Oracle CX Sales NetSuite Customer Update** integration flow which updates a corresponding customer in Oracle NetSuite.

3. Monitor the running of the integration flows in Oracle Integration.

   a. In the project workspace, click **Observe**. You'll see the integration flows being triggered and running successfully.

   b. To manage errors in your project, see Manage Errors in a Project.

4. Verify if a corresponding account has been created or updated in your Oracle CX Sales and B2B Service instance for the new/updated NetSuite customer.

5. Similarly, verify if a corresponding customer has been created or updated in your Oracle NetSuite instance for the new/updated Oracle CX Sales and B2B Service account.

Related Documentation

- *Using the Oracle NetSuite Adapter with Oracle Integration 3*
- *Using the Oracle CX Sales and B2B Service Adapter with Oracle Integration 3*