ORACLE®

Oracle Eloqua-Salesforce App

User Guide

05-Nov-2018

Contents

1 Salesforce Integration app

> **ⓘ Important**: This feature is currently released under our Controlled Availability program. To request access to this feature, please log in to My Oracle Support (https://support.oracle.com) and create a service request.

The Salesforce Integration app creates an integration between Oracle Eloqua and Salesforce to send data between the two platforms. Leverage the Salesforce Integration app in your marketing cloud solution to:

- Transfer the leads generated from marketing campaigns in Oracle Eloqua into Salesforce so they are nurtured and synchronized as sales leads.

- Use the app within a program or campaign canvas to create or update records in supported Salesforce objects in real time. Typical use cases include lead creation, lead updates, and contact updates.

- Schedule imports from Salesforce to Oracle Eloqua for supported Salesforce objects.

You can use the app for various marketing qualification and lead generation use cases. For example, you can configure a listener step to listen for contacts who complete a form. Once the contact is added to the program canvas, you may want to decide to create a lead for the contact. Depending on the next decision, you can configure this app to either create a new lead or update an existing lead within Oracle Eloqua.

1.1 Supported Salesforce objects

The integration supports bi-directional data flow for all objects available via the Salesforce API with the following caveats:

- Objects that are read-only will not be available for use in actions.

- Objects that are not readable will not be available for use in imports.

- If required, you can restrict access to some Salesforce objects. For details, refer to the Salesforce documentation.

2 Getting started with the app

The steps below outline the recommended setup and configuration process for the Salesforce Integration app.

1. Install the Salesforce Integration app. Once the app is installed, configuration settings for connections, actions, imports, and notifications can be found in the apps list. Go to **Settings** ⚙ > **Apps** and click **Configure** ⚙ to access the configuration settings.

2. Create a connected app in Salesforce.

3. Configure one or more connections. Each connection can have multiple connected apps but only one can be active per connection.

4. Create actions. Actions are data transactions between Oracle Eloqua and Salesforce.
   - Use step actions to create and update records to be used in a program or campaign.
   - Use campaign actions to send campaign updates from an Oracle Eloqua campaign object to a custom object in Salesforce. One campaign update action can be active at a time.
   - Use response actions to send data to Salesforce based on response rules configured in your Oracle Eloqua instance. When a response occurs, Oracle Eloqua records the response based on priority and the response is sent to Salesforce.

   > ⓘ **Important**: The response actions feature is currently released under our Controlled Availability program. To request access to this feature, please log in to My Oracle Support and create a service request.

5. Create and schedule imports to import data from Salesforce to Oracle Eloqua periodically. The Salesforce Integration app executes imports every 15 minutes. If an import needs to be performed off schedule, it can be executed on demand.

- Use standard imports to import contact record data from Salesforce into Oracle Eloqua. The following imports are recommended: get leads, get contacts, get accounts.

- Use campaign imports to import custom object data from Salesforce into Oracle Eloqua campaigns to allow for bi-directional updates between the two applications.

6. Setup email notifications to be alerted if there are problems with your imports or actions.

7. Review your reports. The Salesforce Integration app provides reporting on the records being sent between Eloqua and Salesforce.

8. Use the app in a campaign or program to update or create contact records. View recommended programs here.

2.1 Installing the Salesforce Integration app

ⓘ **Important**: The installation process must be done by a user with client administrator rights.

To install the Salesforce Integration app:

1. Follow the URL provided.

   https://cloudmarketplace.oracle.com/marketplace/app/AppSalesforce

   💡 **Note**: If you have IP whitelisting enabled, add internal Eloqua IPs to your whitelist.

2. Log in with your Eloqua credentials, if required.

3. Click **Accept and Install** to add the app to your *Apps* list.

4. Click **Sign In**.

5. Confirm your credentials and click **Accept** to grant permission for the app to communicate with Eloqua on your behalf.

The app is now installed and can be viewed in your *Apps* list (**Settings** ⚙ > **Apps**, under the *Platform Extensions* section). You can configure your app now or at a later time.

You can return to the *Apps* list at any time and view the app description, modify the configuration settings, reinstall, or uninstall it. You can also check its status and dependencies.

> ⚠ **Warning**: To modify configuration settings, navigate to *Apps*, select your app, and click the **Configure** icon ⚙. Selecting the **Reinstall** icon ⬇ will force you to go through the entire configuration process again. The app will be unavailable for all users during the reinstall.

2.2 Configuring connections

You can configure one or more connections and modify connections on the app's *Connections* tab.

Prerequisites

- Install the Salesforce Integration app.

- **Salesforce connected app**: Create a connected app in Salesforce so that you can get the following values to enter when creating a connection:

- **Consumer key**: Displayed in the Salesforce connected app after creation.

- **Consumer secret**: Displayed in the Salesforce connected app

2.2.1 Creating a connected app

The following instructions are based on Salesforce Classic. For Lightning Experience and Classic details, see the Salesforce's connected app documentation.

To create a connected app in Salesforce:

1. Go to https://login.salesforce.com/ and log in to your Salesforce account.

2. From the profile drop-down list, select **Setup**.

3. In the *App Setup* section, select **Create** > **Apps**. The *Apps* page is displayed.

4. In the *Connected Apps* section, click **New**. The *New Connected App* page is displayed.

5. Enter the following required values and any optional values if needed:
   - **Connected App Name**: Enter a descriptive name such as `Eloqua Integration`.
   - **API Name**: You can accept the default value or enter a different name.
   - **Contact Email**: Enter the email address of the user who will manage the connected app.

6. In the *API (Enable OAuth Settings)* section, select **Enable OAuth Settings**. Additional options are displayed. Keep the **Require Secret for Web Server Flow** option, which is selected by default.

7. Enter your Oracle Eloqua**Callback URL**, which depends on your Eloqua pod number:

Pod	Callback URL
Pod 1	https://apps.p01.eloqua.com/eloqua/salesforce/callback
Pod 2	https://apps.p02.eloqua.com/eloqua/salesforce/callback

Pod	Callback URL
Pod 3	https://apps.p03.eloqua.com/eloqua/salesforce/callback
Pod 4	https://apps.p04.eloqua.com/eloqua/salesforce/callback
Pod 6	https://apps.p06.eloqua.com/eloqua/salesforce/callback
Pod 7	https://apps.p07.eloqua.com/eloqua/salesforce/callback

8. In the *Selected OAuth Scopes* section, select the appropriate scopes or select all available scopes.

9. Click **Save**.

10. Select **Create** > **Apps**.

11. In the *Connected Apps* section, find your connected app in the list and click its name.

12. In the connected app's *API (Enable OAuth Settings)* section, record the *Consumer Key* and *Consumer Secret* values.

> 🖉 **Learn more**: Salesforce connected app documentation

2.2.2 Configuring a connection in the Salesforce Integration app

To configure a connection:

1. Log in to Oracle Eloqua.

2. Click the settings icon ⚙.

3. In the *Platform Extensions* section, click **Apps**.

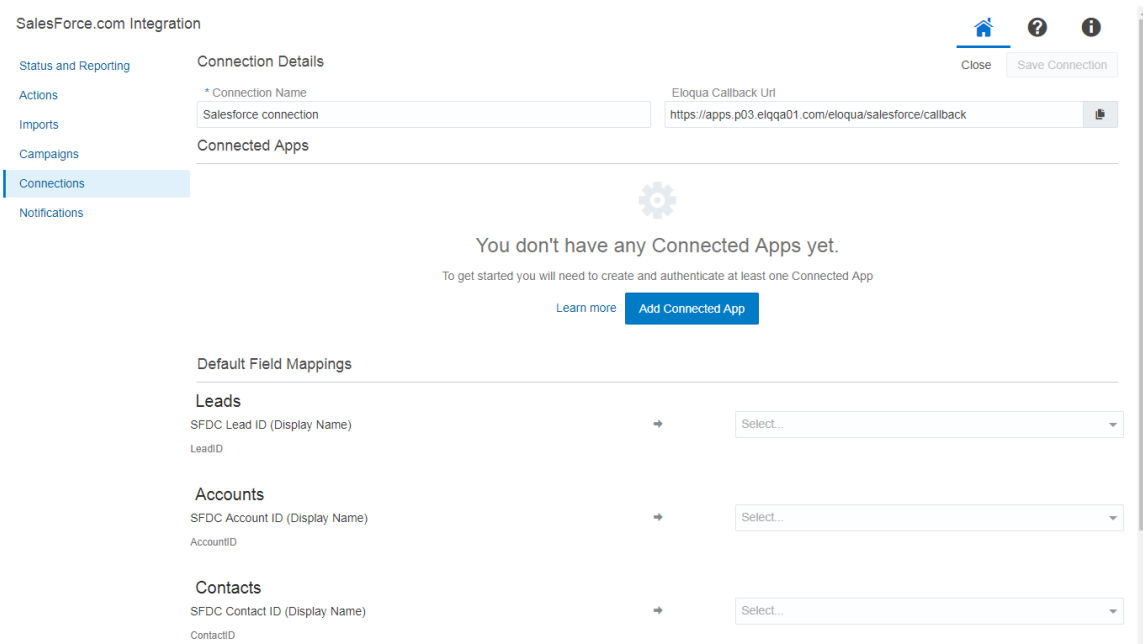4. From the apps list, click **Salesforce Integration for Eloqua**. The *Salesforce Integration for Eloqua* page is displayed.

5. Click the configure icon ⚙. The *App Configuration for Salesforce Integration for Eloqua* page is displayed.

6. Click **Connections**.

7. Click **Create Connection**.

8. Enter a connection name and click **Create Connection**. The **Add Connected App** button and default field mappings are displayed.

9. Click **Add Connected App**.

10. In the *Add a Connected App* section, enter the Consumer Key and Consumer Secret from your Salesforce connected app.

11. Click **Add App**.

12. In the *Default Field Mappings* section, select the Oracle Eloqua fields to map to the corresponding Salesforce Unique Identifier fields.

a. **Leads**: Select an Eloqua field to map to the lead ID field for Salesforce contacts. By default, it is named *SFDC Lead ID*.

b. **Accounts**: Select an Eloqua field to map to the registry ID field for Salesforce accounts. By default, it is named *SFDC Accounts ID*.

c. **Contacts**: Select an Eloqua field to map to the registry ID field for Salesforce contacts. By default, it is named *SFDC Contact ID*.

13. Click **Save Connection**.

14. Click **Close**. The *Connections* page lists existing CRM connections and associated connected apps.

15. Next to the new connected app, click **Authenticate App**. A Salesforce login dialog is displayed.
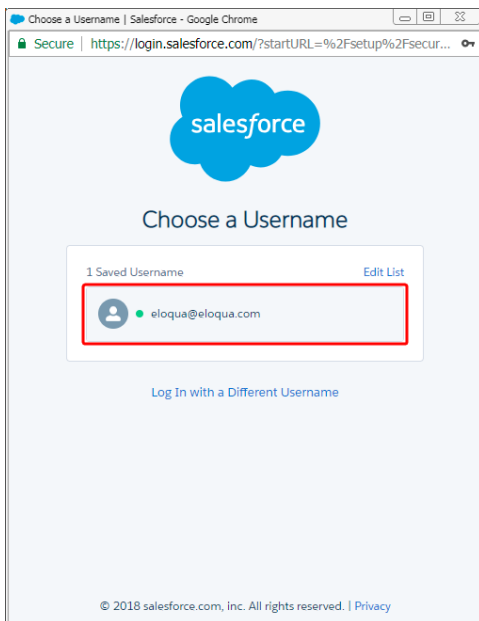
16. Click the default user name or log in with a different user name. The *Allow Access?* dialog is displayed.

17. Click **Allow**. If authentication succeeds, your connected app is ready to be used.

2.2.3 Modifying a connection

To modify an existing connection:

1. In theSalesforce Integration app, click the **Connections** tab.

2. Click the connection's menu icon ☰ and select any of the following options:

   - **Add Action**: Click to create an action and automatically associate it with the current connection.

   - **Add Import**: Click to create an import and automatically associate it with the current connection.

   - **Add New Connected App**: Add another connected app to the connection.

   - **Delete**: You can delete connections if there are no associated actions or imports. A confirmation dialog will list any dependencies. The dependent actions and imports need to be disabled and deleted before deleting the connection.

   - **Edit**: Open the connection in Edit mode so that you can modify its field mappings or connected app values. You cannot modify the callback URL.
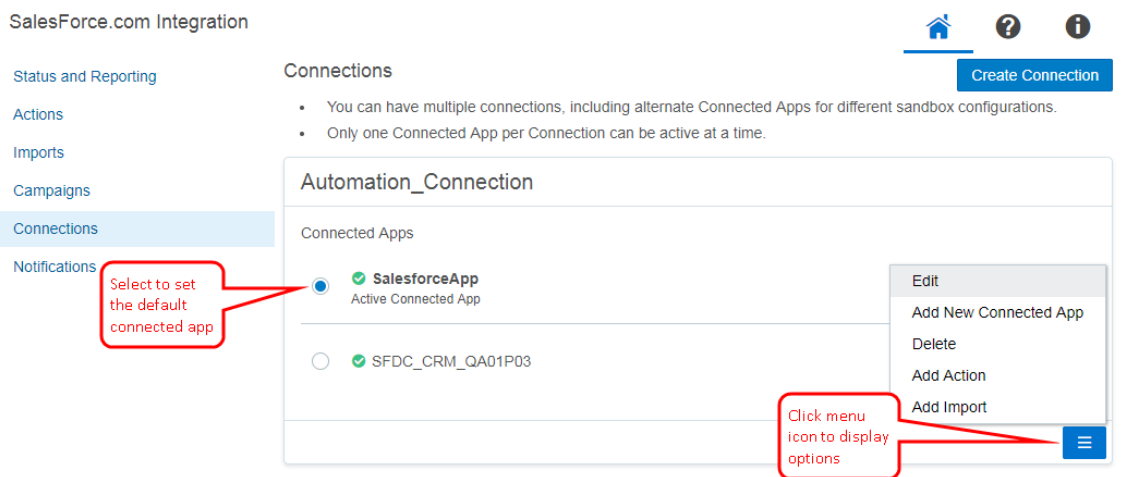
3. If a connection has multiple credentials, you can select the radio button to set the default credential. Only one credential per connection can be active at once. For example, one connected may have production credentials while another connected app has credentials linked to a sandbox or test environment.

2.3 Creating step actions

> **Note**: This page outlines how to create step actions in the Salesforce Integration app. Learn how to create campaign actions and response actions.

Actions are data transactions between Oracle Eloqua and Salesforce. The *Step Actions* page is where you create the types of actions to be used in a program or campaign when using the app. You can create two types of actions: *Create* and *Update*. *Create* will create a new record in Salesforce, and *Update* will take an existing Oracle Eloqua contact and update their information in Salesforce.

Prerequisites

- Configured connection

To create a step action:

1. In the Salesforce Integration app, click **Actions**.

2. Click **Add Action** and select a connection for which you want to create a new action.

3. Complete the action details:
   - **Name**: Enter a name for the action.
   - **Action Type**: Select an action type from the list to either create or update contact records.
   - **Connection**: The connection cannot be modified.

- **Object Mapping**: To see available fields for mapping, select an Eloqua Object and Salesforce Object and click **Fetch**. The unique identifiers for field mappings will be determined once the *Fetch* button is clicked. Top-level objects from your Salesforce instance are available in the Salesforce Object drop-down.

- **Update Salesforce Fields**: For each set of fields, specify the Salesforce field that will be mapped to Oracle Eloqua contacts. If you are updating contact records, the unique identifier is the first mapping shown in this list. Add more mappings by clicking **Add Export Field Mapping**. Click the menu ☰ to delete the mapping, or switch the field type to static text, static date, or dynamic date.

- **Update Eloqua Fields**: For each set of fields, specify the Oracle Eloqua field that will be mapped to Salesforce contacts. If you are creating contact records, the unique identifier is the first mapping shown in this list. Add more mappings by clicking **Add Import Field Mapping**. Click the menu ☰ to delete the mapping or switch a *From Salesforce* field type to static text, static date, or dynamic date.

- **Import Rules**: Specify data priority.
  - **Data Priority**: Select the priority of data to resolve conflicts on imports, such as Bulk API and CRM Integration. If leads and contacts are being imported to Oracle Eloqua contacts, there may be situations where a lead and contact with the same email address exist. Eloqua's data priority can be used to instruct Eloqua which source, lead or contact, is higher priority. If the contact import is set to have a higher priority than the lead import, the data imported from the contact will not be overwritten by the lower priority lead data. Higher priority sources are never overwritten by lower priority sources. Data priority needs to be specified as part of an action to ensure that higher priority sources are not overwritten by lower priority sources from other imports.

4. Click **Save**.

Your action is created and can be used in a program or campaign.

Click the drop-down list to modify the action. You can edit, enable, disable, or view report data for the action. If an action is disabled, you can delete it. If you attempt to delete an action that is used on a canvas, a warning message will display, but it can be overwritten if desired.

View Reports

Edit

Disable

≡

> ℹ **Important**: If an action is being used within a program or campaign and the action is disabled, contacts will not be processed until the action is enabled. Once the action has been re-enabled, the app will process all contacts.
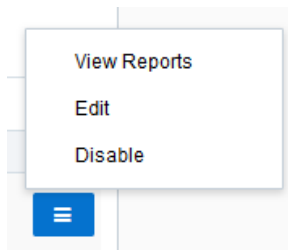
2.4 Creating campaign actions

Actions are data transactions between Oracle Eloqua and Salesforce. The *Campaign Actions* page is where you create actions to send data to Salesforce when campaigns change status. These actions are used for sending newly created campaigns to Salesforce, as well as sending updates to existing campaigns to Salesforce (provided the Eloqua campaign has a Salesforce campaign ID).

> 👍 **Tip**: To sync your campaign with Salesforce, check the **Sync with CRM** box in your campaign's advanced settings (open a campaign > Options > Campaign Settings > Advanced). The CRM Campaign ID field populates after Oracle Eloqua syncs with Oracle Sales Cloud for the first time. This field (CRM Id) is used as a unique identifier for campaign actions. Map at least one more field in addition to

the CRM Id.

Campaigns in Salesforce may also be imported into Eloqua to allow for bi-directional updates between the two applications. When a campaign action is enabled, any campaign imports using different connections are automatically disabled.

Prerequisites

- Configured connection

- Custom object in Salesforce for Eloqua campaigns

To create a campaign action:

1. In the Salesforce Integration app, click **Campaigns**.

2. On the *Campaign Actions* tab, click **Add Campaign Action** and select a connection for which you want to create a new action.

3. Complete the action details:
    - **Action Name**: Enter a name for the action.

    - **Connection**: The connection cannot be modified.

- **Object Mapping**: The Eloqua Object is set to *Campaign* and cannot be modified. To see available fields, select a top-level object from the Salesforce Object list and click **Fetch**. The unique identifiers for field mappings are determined once the *Fetch* button is clicked.

- **Update Salesforce Fields**: The unique identifiers are automatically determined by the Eloqua and Salesforce objects in the Object Mapping section. The Eloqua unique identifier is *CRM Id*. You must add at least one additional mapping by clicking **Add Field Mapping**. For each set of fields, specify the Salesforce field that will be mapped to Eloqua contacts. Click the menu ☰ to delete the mapping, or switch the field type to static text, static date, or dynamic date.

- **Update Eloqua Fields**: The unique identifiers are automatically determined by the Eloqua and Salesforce Objects in Object Mapping. The Eloqua unique identifier is *CRM Id*. Add more mappings by clicking **Add Import Field Mapping**. For each set of fields, specify the Eloqua field that will be mapped to Salesforce contacts. Click the menu ☰ to delete the mapping or switch a *From Salesforce* field type to static text, static date, or dynamic date.

4. Click **Create Campaign Action** and then click **Close** to return to the *Campaign Actions* page.

To modify an action, click its menu and choose and option. You can edit, enable or disable, duplicate, or view report data for the action. Only one campaign action can be enabled at a given time. Enabling a new action will disable an existing one. You can delete disabled actions.

2.5 Creating response actions

> ⓘ **Important**: The response actions feature is currently released under our Controlled Availability program. To request access to this feature, please log in to My Oracle Support and create a service request.

Actions are data transactions between Eloqua and Salesforce. The *Campaigns > Response Actions* tab is where you create actions to send data to Salesforce based on response rules configured in your Oracle Eloqua instance. When a response occurs, Oracle Eloqua records the response based on priority and the response is sent to a Salesforce custom object.

Prerequisites

- A configured connection

- A campaign object or a custom object in Salesforce for Eloqua campaigns

- A response ruleconfigured in your Oracle Eloqua instance

To create a response action:

1. In the Salesforce Integration app, click **Campaigns**.

2. Click the **Response Actions** tab.

3. Click **Add Response Action** and select a connection for which you want to create a new action. The *Create Response Action* page is displayed.
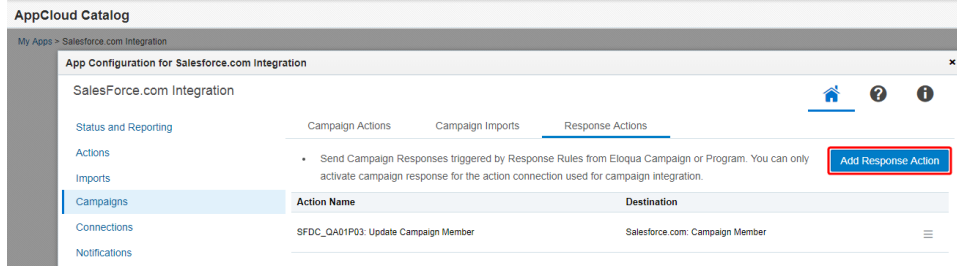
4. Complete the response action details:

- **Action Name**: Enter a name for the action.

- **Campaign Response**: The Eloqua object is set to *CampaignResponse* and cannot be modified.

- **Salesforce.com Object**: Select a top-level object from the list, such as **Campaign**.

- **Update Salesforce.com Fields**: The Eloqua unique identifier is *Integration Return Value*. The Salesforce unique identifier is determined by the Salesforce Object selected in *Object Mapping*. For example, if you select Campaign, the unique identifier is *Campaign ID*. Add more mappings by clicking **Add Field Mapping**. For each set of fields, specify the Salesforce field that will be mapped to Eloqua contacts. Click the menu ≡ to delete the mapping, or switch the field type to static text, static date, or dynamic date.

5. (Optional) Click **+Add Related Object**, select an object from the list, and click **Add Related Object**.

6. Click **Create Response Action** and click **Close** to be returned to the *Response Actions* page.

To modify the action, click the menu to display the options. You can edit, enable/disable, or view report data for the action. Only one campaign response action can be enabled at a given time. Enabling a new action will disable an existing one. You can delete disabled actions.
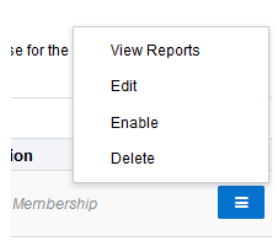
2.6 Creating imports

Imports are data transactions from Salesforce into Oracle Eloqua. The Salesforce Integration app executes imports every 15 minutes. After an import has been scheduled, an import schedule does not display because imports are executed so often. If an import needs to be performed off schedule, it can be executed on demand. An import may fail due to various reasons such as connection or data related issues. Some of these issues will cause an entire import to fail. You can setup email notifications to alert you when import errors occur.

The following imports are recommended:

- Leads

- Contacts

- Accounts

- Opportunities

When viewing your imports, disabled imports are dimmed to indicate their status. There are several indicators in your list of imports to help you quickly get an overview of their status.

Prerequisites

- Configured connection

To create an import:

1. In the Salesforce Integration app, click **Imports**.

2. Click **Add Import** and select a connection for which you want to create a new import.

3. Complete the import details:

   - **Name**: Enter a name for the import.

   - **Object Mapping**: To see available fields for mapping, select an Oracle Eloqua object and Salesforce object and click **Fetch**. The unique identifiers for field mappings will be determined once *Fetch* is clicked. Top-level objects from your Salesforce instance are available in the **Salesforce Object** drop-down.

   - **Import Only Deleted Records**: Select this option if you want to recognize only deleted Salesforce records in Eloqua by creating imports for deleted objects.

   - **Filter Details**: Enter a Salesforce Object Query Language (SOQL) WHERE clause filter to use on the CRM source field when querying contacts to import and then click Test to test the filter. For details about creating SOQL filters, refer to the SOQL developer documentation.

   - **Update Eloqua Fields**: Map your unique identifier to specify the object in Oracle Eloqua to update or create. For each set of fields, select the Oracle Eloqua field that will be mapped to Salesforce contacts. Add more mappings by clicking **Add Import Field Mapping**. Click the menu
     ≡
     to delete the mapping or switch the field type to static text, static date, or dynamic date.

- **Import Rules**: Specify data priority and deduplication rules.

- **Data Priority**: Select the priority of data to resolve conflicts on imports, such as Bulk API and CRM Integration. If leads and contacts are being imported to Oracle Eloqua contacts, there may be situations where a lead and contact with the same email address exist. Oracle Eloqua's data priority can be used to instruct Oracle Eloqua which source, lead or contact, is higher priority. If the contact import is set to have a higher priority than the lead import, the data imported from the contact will not be overwritten by the lower priority lead data. Higher priority sources are never overwritten by lower priority sources.

- **Deduplication type**: Specify how to handle duplicate import records. For example, you can chose the most recently updated record or import all records. Deduplication runs per import execution. If multiple records with the same match field value are imported in a single execution, those records will be deduplicated based on the setting specified here.

- **Post Processing**: Specify an action to be performed on a contact after import.

- **Add to Shared List / Remove from Shared List**: Specify whether after the import, contacts should be added to or removed from a shared list.

- **Subscribe to Email Group / Unsubscribe from Email Group**: Specify whether after the import, contacts should be subscribed to or unsubscribed from an email group.

- **Global Subscribe**: Specify whether after the import, contacts should be globally subscribed globally unsubscribed.

- **Add Action**: Click to add another action to be performed after the import.

4. Click **Save**.

Click the import's drop-down list to modify the import or view reports. You can edit and enable or disable an import. Once an import is disabled, you can delete it.
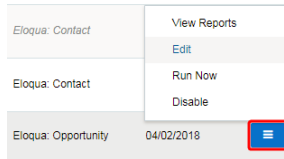
Clicking **Disable** temporarily stops the imports. If the import is currently running and has not yet started (it is in either the *Retrieve from Salesforce* or *Deduplicating* steps), the import is canceled and disabled. If the import has begun importing to Oracle Eloqua, the import will finish, then be disabled. Click **Enable** to enable your import.

2.6.1 Scheduling an import

Once you enable an import, it is scheduled to run automatically every 15 minutes.

You can also run the import on demand clicking **Run Now** and then selecting one of the following:

- **A specific date range**: When **Run Now** is used with a date and time filter, the current date and time is pre-populated. The date and time can be changed to a time in the past as desired. Selecting a specific date allows a partial import to be performed.

- **All Records**: If no date is selected, the import retrieves all matching records from Salesforce and imports them to Oracle Eloqua.

Once you select a specific date range, click **Run Import**.

> 💡 **Note**: The *Last Import* date indicates the last successful import.

Clicking **Disable** temporarily stops the imports. If the import is currently running and has not yet started (it is in either the *Retrieve from Salesforce* or *Deduplicating* steps), the import is canceled and disabled. If the import has begun importing to Oracle Eloqua, the

import will finish, then be disabled. Click **Enable** to schedule your import to run every 15 minutes.

2.7 Creating campaign imports

Imports are data transactions from Salesforce into Oracle Eloqua. The Salesforce Integration app executes imports every 15 minutes. After an import has been scheduled, an import schedule does not display because imports are executed so often. If an import needs to be performed off schedule, it can be executed on demand. An import may fail due to various reasons such as connection or data related issues. Some of these issues will cause an entire import to fail. You can setup email notifications to alert you when import errors occur.

Custom objects in Salesforce Integration app can be imported into Oracle Eloqua campaigns to allow for bi-directional updates between the two applications. When a campaign action is enabled, any campaign imports using different connections are automatically disabled.

Disabled imports are dimmed to indicate their status. There are several indicators in your list of imports to help you quickly get an overview of their status.
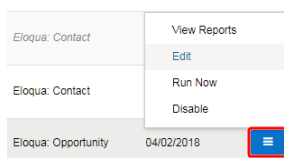
Prerequisites

- Configured connection
- A top-level custom object to store Oracle Eloqua campaigns in Salesforce must exist.

To create a campaign import:

1. In the Salesforce Integration app, click **Campaigns**.

2. Click the **Campaign Imports** tab.

3. Click **Add Import** and select a connection for which you want to create a new import.

4. Complete the import details:

   - **Name**: Enter a name for the import.

   - **Object Mapping**: The **Eloqua Object** is set to *Campaign* and cannot be modified. To see available fields for mapping, select a top-level object from the **Salesforce Object** list.

   - **Filter Details**: Enter a Salesforce Object Query Language (SOQL) WHERE clause filter to use on the CRM source field when querying contacts to import and then click **Test** to test the filter. For details about creating SOQL filters, refer to the Salesforce SOQL developer documentation.

   - **Update Eloqua Fields**: The unique identifiers are automatically determined by the **Eloqua Object** and **Salesforce Object** values in the *Object Mapping* section. For each set of fields, select the Oracle Eloqua field that will be mapped to Salesforce contacts. Add more mappings by clicking **Add Import Field Mapping**. Click the menu ≡ to delete the mapping or switch the field type to static text, static date, or dynamic date.

   - **Import Rules**: Specify deduplication rules.
     - **Deduplication type**: Specify how to handle duplicate import records. For example, you can chose the most recently updated record or import all records. Deduplication runs per import execution. If multiple records with the same match field value are imported in a single execution, those records will be deduplicated based on the setting specified here.

   - **Post Processing**: Post processing rules are not available for campaign imports.

5. Click **Save**.

Click the import's drop-down list to modify the import or view reports. You can edit and enable or disable an import. Once an import is disabled, you can delete it.

Once your import is scheduled, it automatically runs every 15 minutes. For more details, see scheduling an import.

> 💡 **Note**: The *Last Import* date indicates the last successful import.

Clicking **Disable** temporarily stops the imports. If the import is currently running and has not yet started (it is in either the *Retrieve from Salesforce* or *Deduplicating* steps), the import is canceled and disabled. If the import has begun importing to Oracle Eloqua, the import will finish, then be disabled. Click **Enable** to enable your import.

2.8 Creating opportunity imports

An opportunity is a potential sales deal created and maintained by the sales team. You can import Salesforce opportunities into Oracle Eloqua to enable marketers to report on opportunities and measure campaign effectiveness. Data imported by the Salesforce Integration app is required to populate Insight's closed-loop reporting data for Salesforce opportunities.

Opportunity imports have dependencies: they must be linked to contacts directly or to contacts via accounts. The *Imports* list will display the opportunity import as a child of its parent object.

✅	Account Import	Salesforce.com: Account	Eloqua: Account	Today at 9:29 AM	☰
✅	↳ Opportunity Import	Salesforce.com: Opportunity	Eloqua: Opportunity	Today at 9:26 AM	☰

You must specify at least one parent dependency so that the linked Contact object is created in Oracle Eloqua *before* the opportunity is imported. You can specify up to two parent dependencies (a contact import or contact and account import).

Prerequisites

- A configured connection

- Identify the correct names of the object and associated fields that you want to import from Salesforce and use for linking to an Oracle Eloqua object. The linking field is the Salesforce Contact ID or Account ID.

- A successful contact import for the connection, which must be enabled before the opportunity import can be run

- If you will select **Contacts via Account**, you will also need a successful account import for the connection and link the contact import to the account. The linked account and contact imports must both be enabled before the opportunity import can be run.

To create an opportunity import:

1. In the Salesforce Integration app, click **Imports**.

2. Click **Add Import** and select a connection for which you want to create a new import. The *Create Import* page displays a default name for the import.

3. Enter a unique name for the opportunity import.

4. **Import Only Deleted Records**: Select this option if you want to recognize only deleted Salesforce records in Eloqua by creating imports for deleted objects.

5. From the **Salesforce Object** list, select **Opportunity**.

6. From the **Eloqua Object** list select **Opportunity**. The *Opportunity Linking* section is displayed.

7. In the *Update Eloqua Fields* section, you can adjust the default field mappings if necessary. For each set of fields, select the Oracle Eloqua field that will be mapped to Salesforce contacts. You can add more mappings by clicking **Add Import Field Mapping**. Click the menu ≡ to delete a mapping or switch its field type if needed.
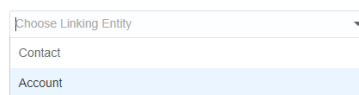
8. In the *Import Rules* section, adjust the default data priority or deduplication rules if needed.

9. From the **Eloqua Entity to link Opportunity records to** list, select one of the following:

   - **Contact**: Indicates that the Oracle Eloqua opportunity import should be linked to the contact import (one parent dependency).

   - **Account**: Indicates that the Oracle Eloqua opportunity import should be linked to the contact import, which is linked with its account import (two parent dependencies). The contact import should first be linked to the account import before the opportunity import is enabled.

10. In the *Dependencies* section, you can add any needed import dependencies by clicking **Add Import Dependency**.

    a. In the *Parent Import* column, select the name of the parent object to establish the opportunity import's dependency on an existing import, such as the name of an existing contact import. If you have two parent dependencies, click **Add Import Dependency** again and add the second dependency. You can establish the dependency while the parent import is disabled. However, the parent import must be enabled before you can enable its opportunities import.

11. Click **Save**. The *Imports* page displays the new opportunity import in the list.

12. If one or more of the opportunity import's parent objects are disabled, click the corresponding menu ☰ and select **Enable** to run the parent import.

13. Click the menu for the new opportunity import and select **Enable** to run it.
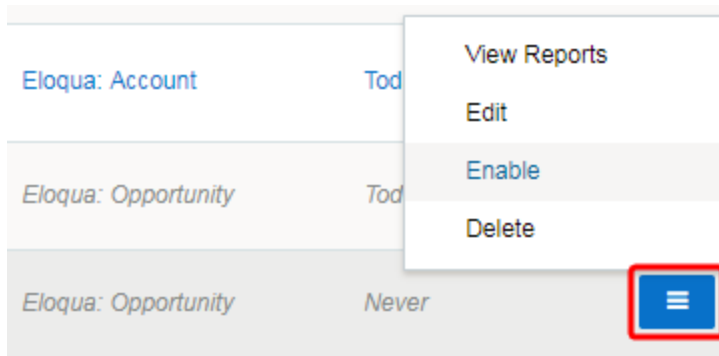
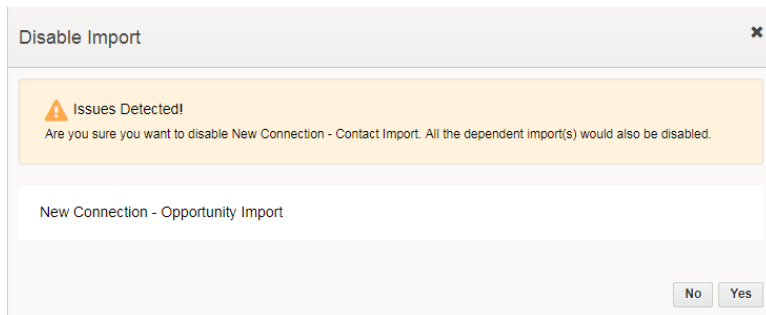Once your import is scheduled, it automatically runs every 15 minutes and is based on the parent import timing and success. If there is a failure on any parent, the child import will not execute.

You can determine whether an opportunity import was successful or partially succeeded by selecting **View Report** from the menu. Some links for the object dependencies should be created for an initial successful opportunity import.

> ⚲ **Note**: The opportunity import report may indicate 0 links after the initial successful import if no additional records were created.

Clicking **Disable** temporarily stops the imports. If the import is currently running and has not yet started syncing to Oracle Eloqua (if it is in either the *Retrieve from Salesforce* or *Deduplicating* steps), the import is canceled and disabled. If the import has begun importing to Oracle Eloqua, the import will finish, then be disabled.

If you click **Disable** or **Delete** for an import that has dependencies, a message will indicate those dependencies.

Disable Import   ✕

> ⚠ **Issues Detected!**
> Are you sure you want to disable New Connection - Contact Import. All the dependent import(s) would also be disabled.

New Connection - Opportunity Import

[No] [Yes]

Before clicking **Yes** to disable or delete the import, determine whether you also want to disable the linked imports (or delink the imports if you are deleting a parent import).

2.9 Creating user imports

Salesforce users can be imported into Oracle Eloqua. By importing, users can be created and updated, they can be placed in a folder, and added to or removed from a security group.

Imports are data transactions from Salesforce into Oracle Eloqua. The Salesforce Integration app executes imports every 15 minutes. After an import has been scheduled, an import schedule does not display because imports are executed so often. If an import needs to be performed off schedule, it can be executed on demand. An import may fail due to various reasons such as connection or data related issues. Some of these issues will cause an entire import to fail. You can setup email notifications to alert you when import errors occur.

Disabled imports are dimmed to indicate their status. There are several indicators in your list of imports to help you quickly get an overview of their status.

Prerequisites

- Configured connection

- A BI user report must exist in Salesforce.

To create a user import:

1. In the Salesforce Integration app, click **Imports**.

2. Click **Add Import** and select a connection for which you want to create a new import.

3. Complete the import details:
   - **Name**: Enter a name for the import.

   - **Connection**: The connection cannot be modified.

   - **Salesforce Object**: Select *User*.

   - **Report Path**: Enter in the path to the user report in Salesforce.

     > 💡 **Note**: A BI user report must first exist in Salesforce.

   - **Eloqua Object**: Select *User*.

   - Click **Fetch** to see available fields for mapping.

   - **Update Eloqua Fields**: Map your unique identifier to specify the object in Oracle Eloqua to update or create. For each set of fields, select the Oracle Eloqua field that will be mapped to Salesforce contacts. Add more mappings by clicking **Add Import Field Mapping**. Click the menu ≡ to delete the mapping or switch the field type to static text, static date, or dynamic date.

- **Import Rules**: Specify folder and deduplication rules.

- **Folder**: Select an Oracle Eloqua folder where the user should be saved. All existing user folders in your instance are available in this drop-down menu. New folders cannot be created from this menu.

- **Deduplication type**: Specify how to handle duplicate import records. For example, you can chose the most recently updated record or import all records. Deduplication runs per import execution. If multiple records with the same match field value are imported in a single execution, those records will be deduplicated based on the setting specified here.

- **Post Processing**: Click **Add Action** to specify an action to be performed on a user after import.

- **Add to Security Group / Remove from Security Group**: Specify whether after the import, users should be added to or removed from a security group.

4. Click **Save**.

Click the drop-down list to modify the import. You can edit, enable (to schedule your import) and disable, or view report data for an import. If an import is disabled, you can delete it.

Once your import is scheduled, it automatically runs every 15 minutes. For more details, see scheduling an import.

> 💡 **Note**: The *Last Import* date indicates the last successful import.

Clicking **Disable** temporarily stops the imports. If the import is currently running and has not yet started (it is in either the *Retrieve from Salesforce* or *Deduplicating* steps), the import is canceled and disabled. If the import has begun importing to Oracle Eloqua, the import will finish, then be disabled. Click **Enable** to enable your import.

2.10 Configuring notifications

An import or action may fail due to various reasons such as connection or data related issues. Use the *Notifications* page in the Salesforce Integration app to setup email notifications to alert you of import and action errors. The notification includes your client and import name, start and end times, and identifies the step at which the error occurred.

✖ An error has occurred with your import from CRM
ERROR: Retrieving from Sales Cloud
Import details:

Client:	ExampleClient
Import:	TestImport
Start Time:	2016-11-24T22:01:00Z
End Time:	2016-11-24T22:01:02.224Z

If the problem persists, please log into My Oracle Support and submit a case or call 1-800-223-1711 for technical support.

We apologize for the inconvenience.

To configure a notification:

1. In the Salesforce Integration app, click **Notifications**.

2. Click **Add Notification**.

3. Select the notification settings and recipients:

- *Name*: Provide a descriptive name that will appear in the list of notifications.

- *Connection*: Select a specific connection or all connections.

- *Event Source Type*: Notifications are currently available for imports and actions.

- *Event Source*: Select All Imports or All Actions.

- *Event Type*: Notifications are available for failed imports and actions.

- *Email(s)*: Enter recipient email addresses. You can enter multiple email addresses separated by commas.

4. Click **Save**.

You can return to the notification area to create new notifications, as well as edit, delete, and disable existing notifications.

3 Using the app in a program or campaign

The Salesforce Integration app enables you to update or create contact records within a program or campaign.

To use the Salesforce Integration app in a campaign or program:

1. Open an existing campaign or program, or create a new one.

2. Click on the *Actions* section to expand it, and view all available action elements, including installed apps.

3. Drag the **Salesforce Integration app** element onto the canvas, then double-click it to open the configure screen.

   The window that opens enables you to do the following:

   - Change the name of the step to something more meaningful.

   - Configure the element.

   - Choose to route contacts that resulted in an error to another step (see: configuring campaign elements for more information).

4. Click the **Edit** icon to configure the action.
   a. Click the **Actions** drop-down to select an existing action. If you do not see any existing actions, see configuring actions for more information.

   b. Click **Save**.

   c. (Optional) Click **Status** to see the status of contact processing. The amount of contacts that have been successfully processed, or were processed with warnings or errors will be displayed.

4 Viewing report data for actions and imports

Individual reporting for Salesforce Integration app actions and imports displays activity for a specific import or action for the current day (previous dates are selectable). A table of every import or action execution, including details of created, updated, and failed records and time of execution. Data is retained for a period of three months.

To view report data for actions or imports:

1. In the Salesforce Integration app, click **Actions** or **Imports**. To view reports for campaign actions, click the **Campaign Actions** tab.

2. Locate the action or import for which you want to view report data, click the list, and select **View Reports**.

4.1 Action report data

Action reports are available for step actions and campaign actions. These reports provide information about the records that are flowing from Oracle Eloqua into Salesforce.

For step actions, view how many records were created or updated, how many were processed per hour, how long it took to complete, and whether or not it was successful.
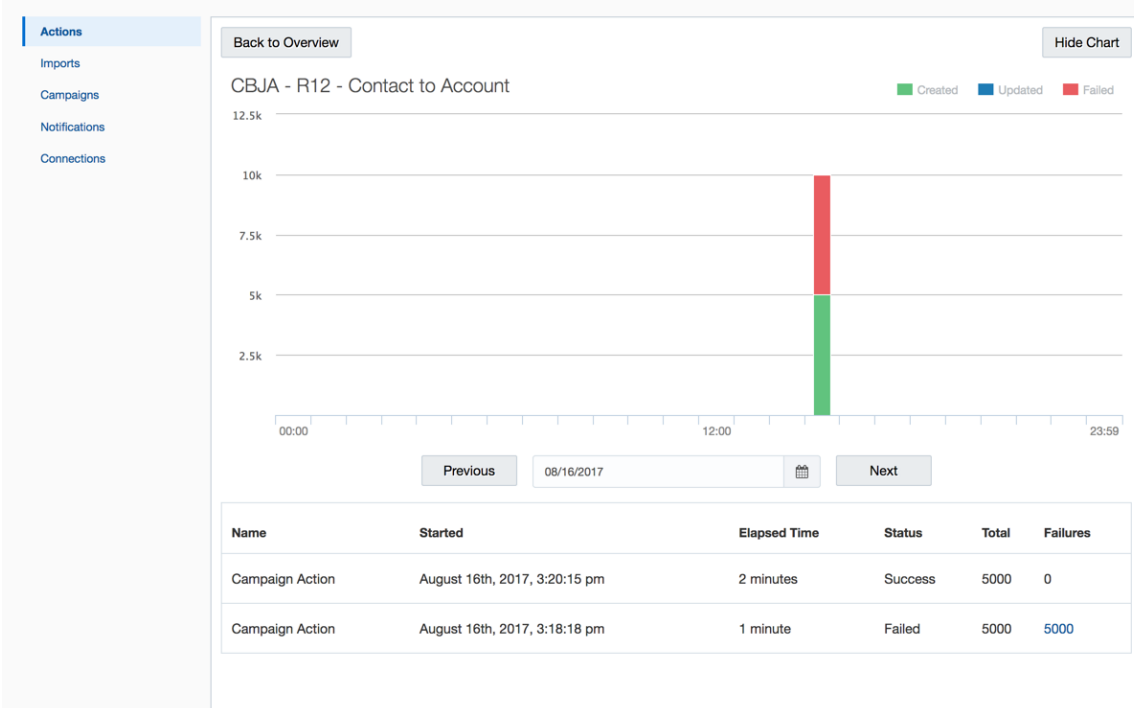
For campaign actions, view whether the campaigns exported from Oracle Eloqua to CRM were successful and a daily history of your campaign exports.
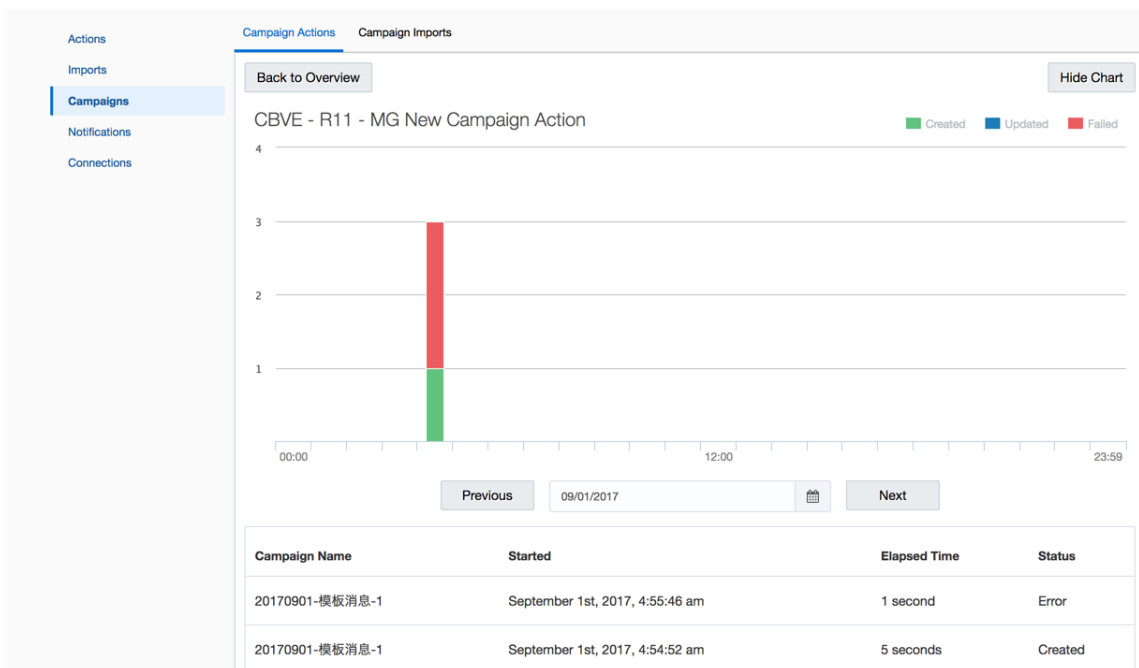
Click a table row to view more details of the data sent over, sent date, and error messages.

4.2 Import report data

Import reports provide insight into how your imports are performing by displaying a chart of how many records are being created, updated, or have failed to import. You can also view how many objects were imported into Oracle Eloqua campaigns from CRM, which ones were successful, and how many failed.

Each report displays an import's schedule start time, how long it took to complete, whether or not it was successful, and how many records were created, updated, or failed. When an import is running, click the import to view the progress.

The status of the import is displayed including:

- Import data

- When the import started

- How long the import took to complete

- The number of records with their status

There are three stages to a successful import; records are retrieved from Salesforce, deduplicated, and uploaded to Oracle Eloqua. The message *Retrieving from Salesforce* indicates your import is successfully running.

You can manually stop an import if it is running and has not yet reached the *Uploading to Eloqua* step. Once the *Uploading to Eloqua* step is started, the import cannot be stopped and will continue to completion. Stopped imports cannot be resumed. You must run the import again. Learn more about creating and scheduling imports.

5 Example programs

The Salesforce Integration app can be used for various marketing qualification processes and lead generation. Numerous programs can be designed using Oracle Eloqua's program canvas to update and create leads and contacts in Salesforce.
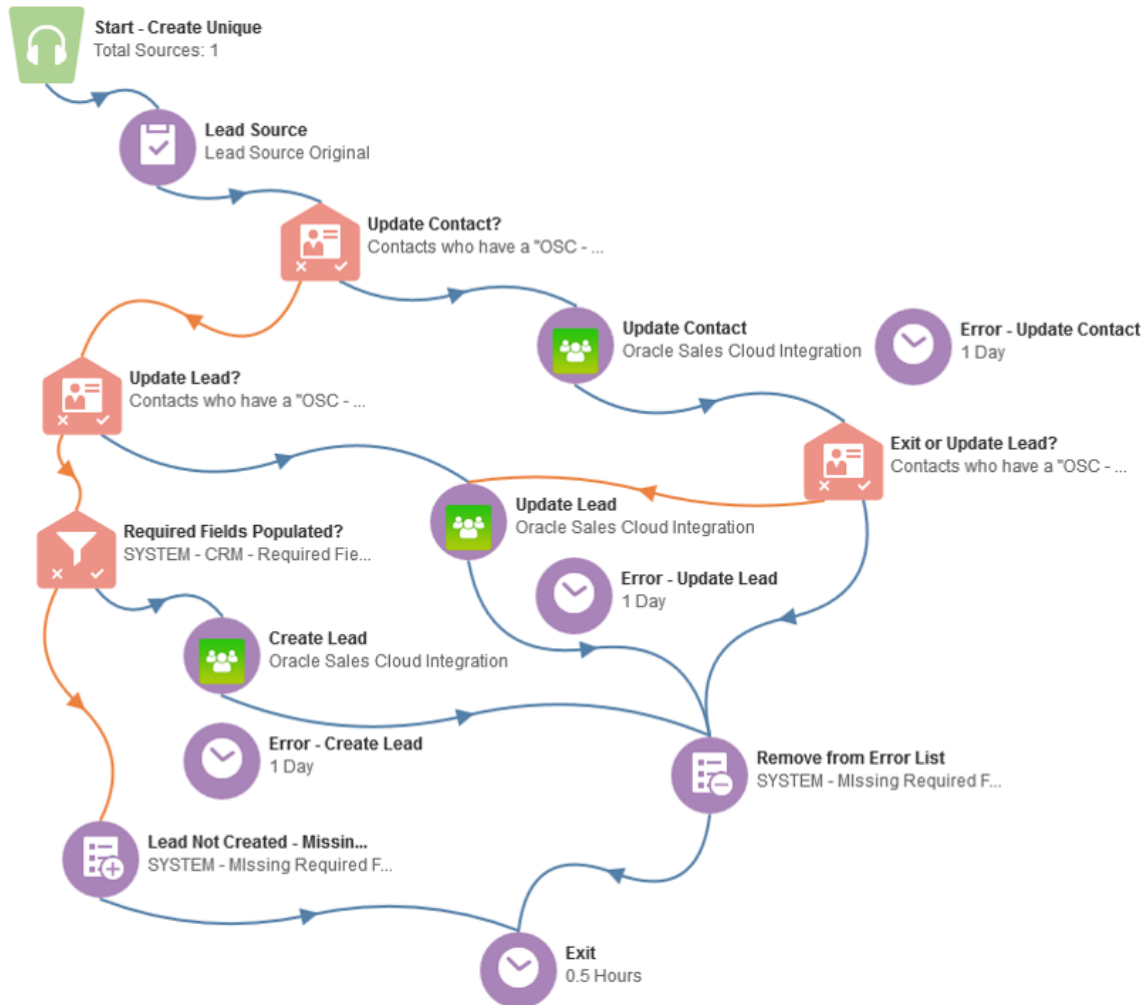
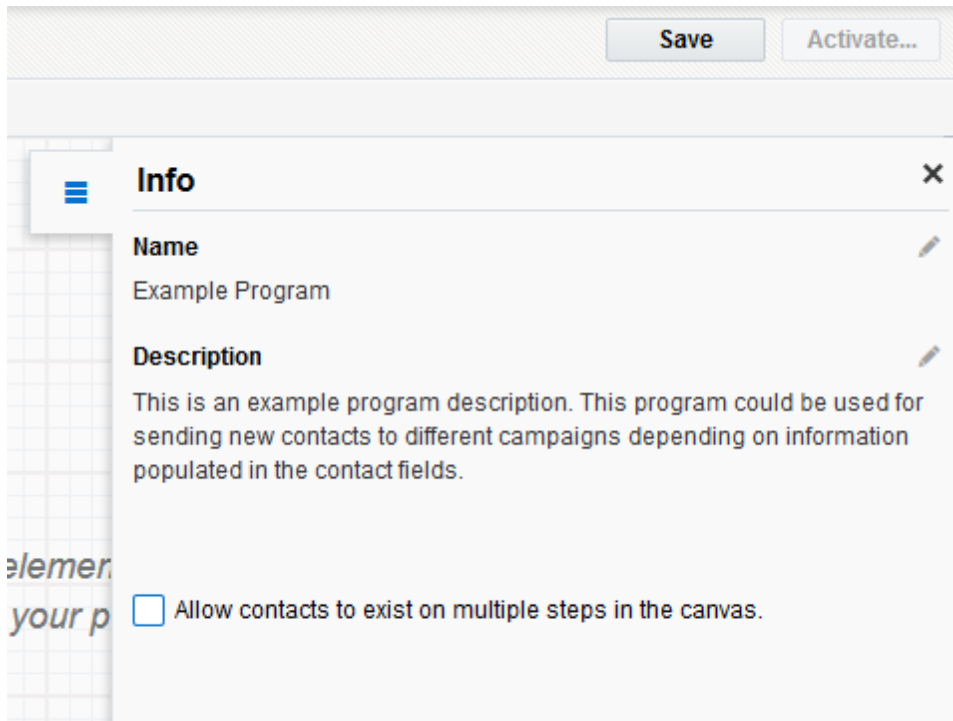These are two recommended programs:

- Create Unique: In this program, a new lead record is created in Salesforce only if there is no contact or lead record with the same email address in Salesforce. If there is an existing contact record, the contact record is updated with contact information as defined in the *Update Contact* action.

- Point of Interest: A new lead record is created if there are no leads with the same email address in Salesforce, but an Salesforce contact already exists. If there is an existing contact record, the contact record is updated with contact information from Oracle Eloqua as defined in the *Update Contact* action.

5.1 Creating a Salesforce Create Unique program

Program Canvas is a drag-and-drop whiteboard that allows you to build data management workflows inside of Oracle Eloqua. The following outlines how to create a *Create Unique* program for Salesforce.

Start - Create Unique
Total Sources: 1

Lead Source
Lead Source Original

Update Contact?
Contacts who have a "OSC - ...

Update Contact
Oracle Sales Cloud Integration

Error - Update Contact
1 Day

Update Lead?
Contacts who have a "OSC - ...

Exit or Update Lead?
Contacts who have a "OSC - ...

Required Fields Populated?
SYSTEM - CRM - Required Fie...

Update Lead
Oracle Sales Cloud Integration

Error - Update Lead
1 Day

Create Lead
Oracle Sales Cloud Integration

Error - Create Lead
1 Day

Remove from Error List
SYSTEM - MIssing Required F...

Lead Not Created - Missin...
SYSTEM - MIssing Required F...

Exit
0.5 Hours

In this program, a new lead record is created in Salesforce only if there is no contact or lead record with the same email address in Salesforce. If there is an existing contact record, the contact record is updated with contact information as defined in the *Update Contact* action.

> 💡 Note: Determination of whether a Salesforce contact or lead exists is dependent on the configuration of an import for both Salesforce leads and contacts with the appropriate record IDs.

To create a Salesforce Create Unique program:

1. Navigate to **Orchestration** ⊞, then click **Programs**.

2. Click **Create a Contact Program**.

3. Select a blank template.

4. Configure basic information about the program:

   a. Click **Info** ☰ on the left side of the canvas, and add a **Name** and **Description** of the program.

   b. To allow a contact to enter the program more than once, select the **Allow contact to exist on multiple steps in the canvas** check box. If you do not select this option, once a contact enters a program, the contact cannot enter the program again from another entry point.

   c. Click **X** to close the *Info* window.

5. Add the **Listener Step** to the canvas:

   - Name the step *Start - Create Unique*.

   - To subscribe to data using a listener step, double-click 🎧 **Listener Step**.

   - Add any data sources that are part of your lead flow. For example, form submissions, lead nurturing programs or campaigns that generate leads.

6. Add an **Update Rule** action step. Name the step *Lead Source*. Configure the *Rule Set* option to **Lead Source Original**.

7. Add a **Compare Contact Fields** decision step. Name the step *Update Contact?* This step evaluates and routes contacts based on a specified value in a selected contact field. Set the step to route contacts based on whether they have a *Salesforce Contact Registry ID*.

   a. For the *yes* path, add a **Salesforce Integration** action to update information for existing Salesforce contacts. Name this step *Update Contact*.

   b. For the *no* path, add a **Compare Contact Fields** decision to route contacts based on whether they have a *Salesforce Lead ID*. Name this step *Update Lead?*

8. For the Update Lead? step, add the following paths:

   a. For the *yes* path, add a **Shared Filter Member?** action. Name this step *Required Fields Populated?*. This step evaluates whether shared filter members have required contact fields (such as name, company, and so on). Add a **Salesforce Integration** action named *Create Leads* and connect it to the *yes* path to create leads for filter members with the required contact fields.

   b. For the *no* path, add a **Salesforce Integration** action to update information for existing Salesforce leads. Name this step *Update Lead*.

9. Add any necessary wait steps, error flows, shared list actions, and Salesforce Integration app actions as depicted in the workflow.

10. Click **Save**.

11. Activate your program.

12. Monitor the performance of the program using operational reports. If you need to change your program, you must disable the program.

5.2 Creating a Salesforce Point of Interest program

Program Canvas is a drag-and-drop whiteboard that allows you to build data management workflows inside of Oracle Eloqua. The following outlines how to create a *Point of Interest* program for Salesforce.
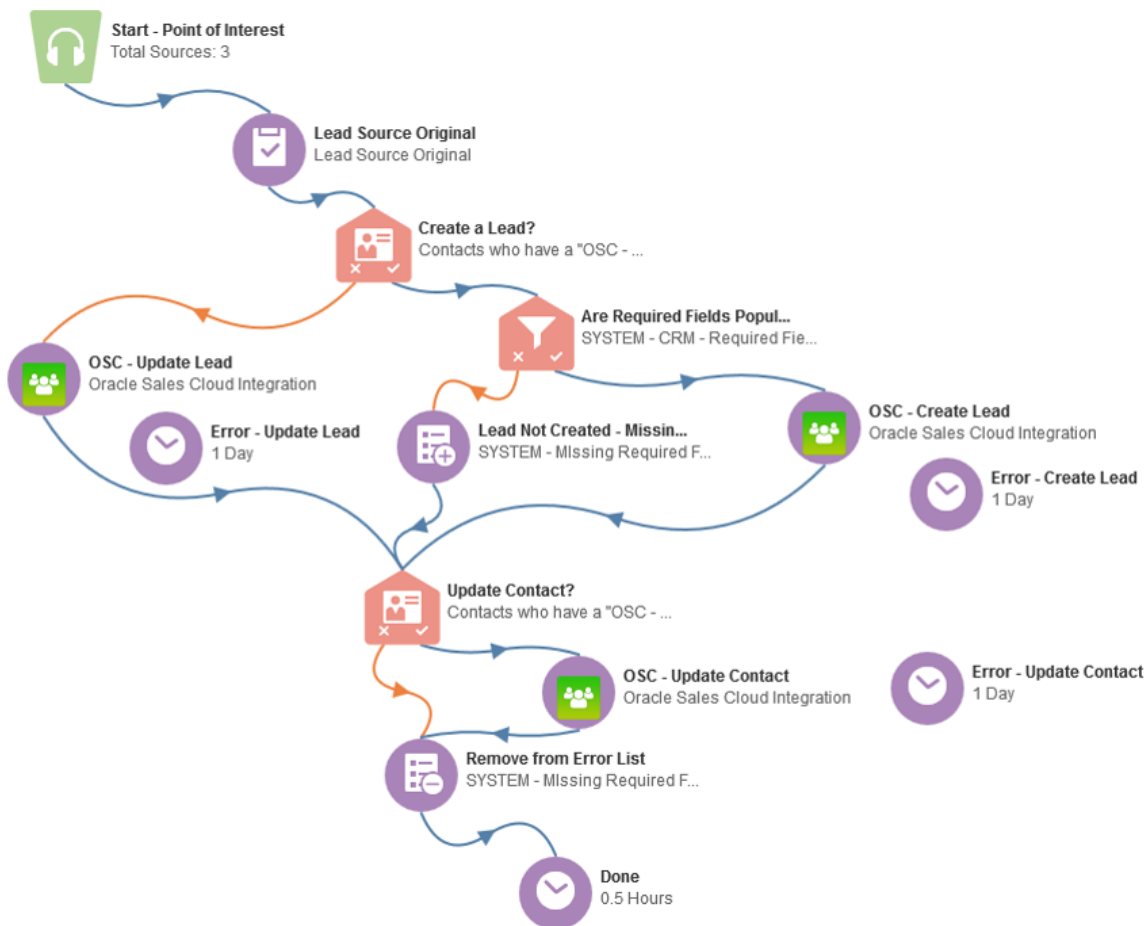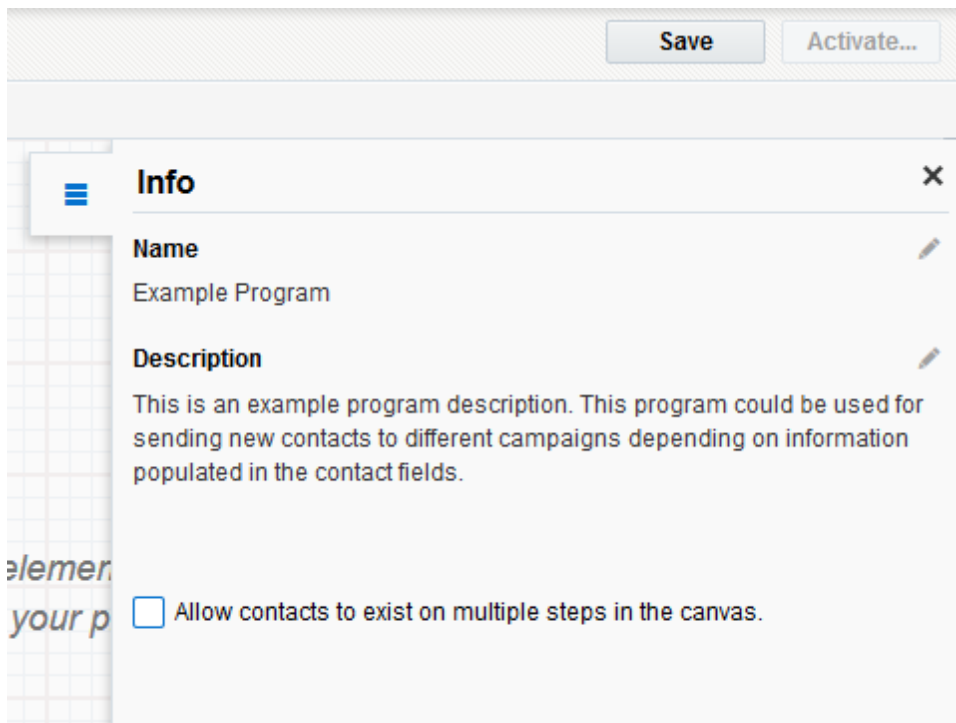

A new lead record is created if there are no leads with the same email address in Salesforce, but an Salesforce contact already exists. If there is an existing contact record, the contact record is updated with contact information from Oracle Eloqua as defined in the *Update Contact* action.

> 💡 **Note**: Determination of whether a Salesforce contact or lead exists is dependent on the configuration of an import for both Salesforce leads and contacts with the appropriate record IDs.

To create an Salesforce Point of Interest program:

1. Navigate to **Orchestration** ⊞, then click **Programs**.

2. Click **Create a Contact Program**

3. Select a blank template.

4. Configure basic information about the program:
   a. Click **Info** ≡ on the left side of the canvas, and add a **Name** and **Description** of the program.

b. To allow a contact to enter the program more than once, select the **Allow contact to exist on multiple steps in the canvas** check box. If you do not select this option, once a contact enters a program, the contact cannot enter the program again from another entry point.

c. Click **X** to close the *Info* window.

5. Add the **Listener Step** to the canvas:

- Name the step *Start - Point of Interest*.

- To subscribe to data using a listener step, double-click **Listener Step**.

- Add any data sources that are part of your lead flow. For example, form submissions, lead nurturing programs or campaigns that generate leads.

6. Add an **Update Rule** action step. Name the step *Lead Source*. Configure the *Rule Set* option to **Lead Source Original**.

7. Add a **Compare Contact Fields** decision step. Name the step *Create a Lead?* This step evaluates and routes contacts based on a specified value in a selected contact field. Set the step to route contacts based on whether they have a *Salesforce Lead ID*.

a. For the *yes* path, add a **Shared Filter Member?** action. Name this step *Are Required Fields Populated?*. This step evaluates whether shared filter members have required contact fields (such as name, company, and so on). Add a **Salesforce Integration** action named *SFDC - Create Lead* and connect it to the *yes* path to create leads for filter members with the required contact fields. For the *no* path, move members to an **Add to Shared List** action named *Lead Not Created - Missing Fields*.

b. For the *no* path, add a **Salesforce Integration** action to update information for existing Salesforce leads. Name this step *SFDC - Update Lead*.

8. Add a **Compare Contact Fields** decision step. Name this step *Update Contact?*

a. For the *yes* path, add an **Salesforce Integration** action to update information for existing Salesforce contacts. Name this step *SFDC - Update Contact*. Connect this step to the *Remove from Error List* step below.

b. For the *no* path, add a **Remove from Shared List** action. Name this step *Remove from Error List*.

9. Add any necessary wait and error steps.

10. Click **Save**.

11. Activate your program.

12. Monitor the performance of the program using operational reports. If you need to change your program, you must disable the program.