

Salesforce Provisioning Plugin

The Salesforce Provisioning Plugin provisions Contacts to [Salesforce](#) via the [Force.com REST API](#).

- [Operations](#)
- [Configuration](#)
 - [CoPerson Custom Object](#)
- [Attributes](#)
- [Understanding API Usage](#)
- [See Also](#)


Operations



| Registry CO Person Transaction | Changelog Action |
|--------------------------------|---|
| Add | Synchronize the CO Person with Salesforce |
| Edit | Synchronize the CO Person with Salesforce |
| Enter Grace Period | No changes (unless attributes change as part of grace period) |
| Expiration / Becomes Inactive | Synchronize the CO Person with Salesforce |
| Unexpire / Becomes Active | Synchronize the CO Person with Salesforce |
| Delete | No changes |
| Manual Provision | Synchronize the CO Person with Salesforce |

| Registry CO Group Transaction | Changelog Action |
|-------------------------------|------------------|
| Add | No changes |
| Edit | No changes |
| Delete | No changes |
| Manual Provision | No changes |

| Registry CO Email List Transaction | Changelog Action |
|------------------------------------|------------------|
| Add | No changes |
| Edit | No changes |
| Delete | No changes |
| Manual Provision | No changes |

Configuration

1. This is a non-core plugin, see [Installing and Enabling Registry Plugins](#) for more information.
2.  If you already have a Salesforce Server defined with the appropriate configuration (perhaps for use with the [Salesforce Source](#)) you can reuse it instead of defining a new one. You only need to complete step 6.
3. Add a new Server, via *Servers > Add a New Server*
 - a. Set the server type to *OAuth2*.
 - b. After the configuration has been saved, a Redirect URI will be available via the server configuration page. Keep this handy for the next step.
4. In another browser tab or window, login to Salesforce. [Add a new Connected App](#) via *Setup > Quick Links > Manage Apps*. Click the *New* button in the *Connected Apps* section.
 - a. Set the *Connected App Name* and *Contact Email*.
 - b. Under *API*, tick *Enable OAuth Settings*.
 - c. Set the *Callback URL* to the URI provided in step 1, above.
 - d. Added at least these two *OAuth Scopes*:
 - i. *Access and manage your data (api)*
 - ii. *Perform requests on your behalf at anytime (refresh_token, offline_access)*
 - e. Click *Save*. (You may need to scroll up to see the confirmation message.)
 - f. On the next page, a *Consumer Key* and *Consumer Secret* will be made available. Keep these handy for the next step.
5. Return to the OAuth2 Server configuration and complete the configuration.

- a. **Server URL:** The base URL of your Salesforce instance with `/services/oauth2` appended, eg <https://test.salesforce.com/services/oauth2>
 - i.  The plugin will work with either the generic service name (test.salesforce.com) or a specific instance (cs123.salesforce.com), but note that Salesforce periodically migrates customers to new instances (in Salesforce terms, an *instance refresh*). In such an event, when configured with the generic service name the plugin should detect the new instance automatically, though it may be necessary to obtain a new token (described below).
 - b. **Client ID:** The Consumer Key obtained in step 2.
 - c. **Client Secret:** The Consumer Secret obtained in step 2.
 - d. **Access Token Grant Type:** Authorization Code
 - e. **Scope:** (Leave blank)
 - f. Click **Save**.
6. Define a new CO Provisioning Target using the SalesforceProvisioner plugin. On the configuration page, select the Server set up above and click **Save**.
 7. Finally, return to the OAuth2 Server configuration to obtain an OAuth token.
 - a. The configuration should indicate that the Access Token is "Not Set", and there should now be a button "Obtain New Token".
 - b. Upon clicking that button, you will be taken to the Salesforce login page. Log in as a sufficiently authorized user.
 - c. After successful login, you should be returned to the OAuth2 Server configuration page, and the Access Token should now be "Set".
 - d.  Should it ever be necessary to obtain a new token (eg: if the administrator who performed the initial setup no longer has a valid Salesforce account), simply return to the configuration page and click the "Obtain New Token" button again.

CoPerson Custom Object



The SalesforceProvisioner plugin supports the use of a Salesforce Custom Object in order to provide additional attributes (see below) beyond what is supported in the Salesforce Contact record. To enable support for the CoPerson Custom Object:

1. Login to Salesforce and navigate to *Setup > Create > Objects*.
2. Click *New Custom Object*. Use the follow (exact) values:
 - a. Label: *CoPerson*
 - b. Plural Label: *CoPeople*
 - c. Object Name: *CoPerson*
 - d. Record Name: *CoPerson ObjectID*
 - e. Data Type: *Auto Number*
 - f. Display Format: *C{0}*
 - g. Starting Number: *100000* (or whatever number you prefer)
 - h. Deployment Status: *Deployed*
3. Create the following Custom Fields, again using the exact configuration specified:
 - a. Field Label and Field Name: *Contact*
 - i. Data Type: *Master-Detail*
 - ii. Related To: *Contact*
 - iii. Child Relationship Name: *CoPeople*
 - b. Field Label and Field Name: *Application ID*
 - i. Data Type: *Text*
 - ii. Length: *255*
 - c. Field Label and Field Name: *ORCID*
 - i. Data Type: *Text*
 - ii. Length: *255*
 - d. Field Label and Field Name: *Platform ID*
 - i. Data Type: *Text*
 - ii. Length: *255*
 - iii. External ID: *Checked*
 - e. Field Label and Field Name: *Status*
 - i. Data Type: *Text*
 - ii. Length: *32*
4. In the CManage plugin configuration, tick *Enable CoPerson Custom Object Support*.
 - a. Setting an Identifier type for *CoPerson Platform ID Identifier Type* is required. This identifier is considered by Salesforce as the External ID, and so must always be populated and unique. [Identifier Assignments](#) may be useful.

 The above instructions will not cause the Custom Object to appear on the Contact page, though it is visible if directly addressed.

Attributes

The Salesforce Provisioner uses the following attribute mapping:

| Registry Attribute | Salesforce Attribute | Notes |
|---|----------------------|--|
| Address:country (type <code>postal</code>) | MailingCountry | First Address of type <i>postal</i> associated with the first CoPersonRole  Values must be from Salesforce picklist. Attribute Enumerations may be useful. |
| Address:locality (type <code>postal</code>) | MailingCity | First Address of type <i>postal</i> associated with the first CoPersonRole |
| Address:postal_code (type <code>postal</code>) | MailingPostalCode | First Address of type <i>postal</i> associated with the first CoPersonRole |
| Address:state (type <code>postal</code>) | MailingState | First Address of type <i>postal</i> associated with the first CoPersonRole  Values must be from Salesforce picklist. Attribute Enumerations may be useful. |

| | | |
|---|-------------------------------|--|
| Address:street (appended with Address:room, if set) (type <code>postal</code>) | MailingStreet | First Address of type <i>postal</i> associated with the first CoPersonRole |
| CoPerson:status | CoPerson__c:Status__c | If CoPerson custom object is enabled |
| CoPersonRole:ou | Department | Department associated with the first CoPersonRole, sorted by CoPersonRole order. |
| CoPersonRole:title | Title | Title associated with the first CoPersonRole, sorted by CoPersonRole order. |
| EmailAddress:mail | Email | If <i>Email Address Type</i> is set, the first Email Address of the specified type is used. Otherwise, the first Email Address of any type is used. |
| Identifier:identifier | CoPerson__c:Application_ID__c | If CoPerson custom object is enabled, type as configured in <i>CoPerson Application ID Identifier Type</i> |
| Identifier:identifier | CoPerson__c:Platform_ID__c | If CoPerson custom object is enabled, type as configured in <i>CoPerson Platform ID Identifier Type</i> |
| Identifier:identifier (type <code>orcid</code>) | CoPerson__c:ORCID__c | If CoPerson custom object is enabled |
| Name:honorific | Salutation | Primary Name is used, regardless of type |
| Name:given | FirstName | Primary Name is used, regardless of type |
| Name:middle | Middlename | Primary Name is used, regardless of type ⚠ Middle Name is not enabled by default in Salesforce. Once enabled there, tick <i>Provision Middle Name</i> in the Provisioner configuration. |
| Name:family | LastName | Primary Name is used, regardless of type |
| Name:suffix | Suffix | Primary Name is used, regardless of type |
| TelephoneNumber:number (type <code>mobile</code>) | MobilePhone | First TelephoneNumber of type <i>mobile</i> associated with the first CoPersonRole |
| TelephoneNumber:number (type <code>office</code>) | Phone | First Telephonumbe of type <i>office</i> associated with the first CoPersonRole |

Understanding API Usage



The Salesforce API has [request limits](#) that vary according to the service tier and available licenses. To view current API usage, use the [Salesforce Source](#) plugin if configured, or login to Salesforce and navigate to *Reports > Administrative Reports > API Usage Last 7 Days*.



Note that the API call required to determine the current API limits counts against the API limit.

In general, each provisioning operation (create or update) requires 1 API call, plus an additional API call if the CoPerson custom object is enabled.

See Also

- [cm_co_salesforce_provisioner_targets](#)