

Last Updated: 15 November, 2012

Initial Setup

After installing the Nexternal Commerce Connector from the Salesforce AppExchange you will need to make two adjustments to your Salesforce account settings:

1. [Add the Nexternal tab](#) to your Salesforce account page.
2. [Add a Remote Site Setting](#) to enable the Commerce Connector to download information from your Nexternal storefront.

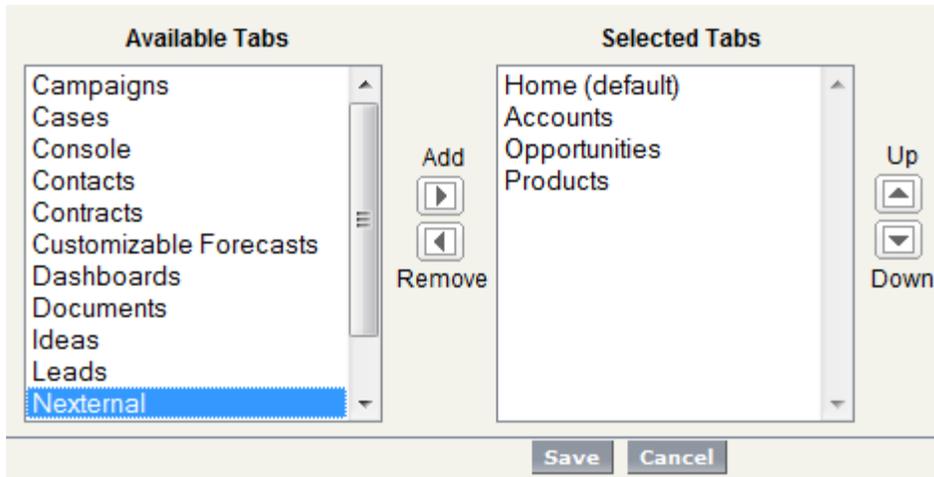
Prior to performing the initial import from your Nexternal storefront please review the [Importing Your Orders](#) section to make sure the import dates have been set (or reset, if you have been testing the addin before installing it in your production Salesforce environment), along with the [What Happens During An Import](#) section and the [Advanced Configuration](#) section below to ensure the import process won't conflict with any existing information you have entered into Salesforce.

Add the Nexternal tab

Click on the  tab at the far right of your Salesforce account page to display the **All Tabs** configuration settings.

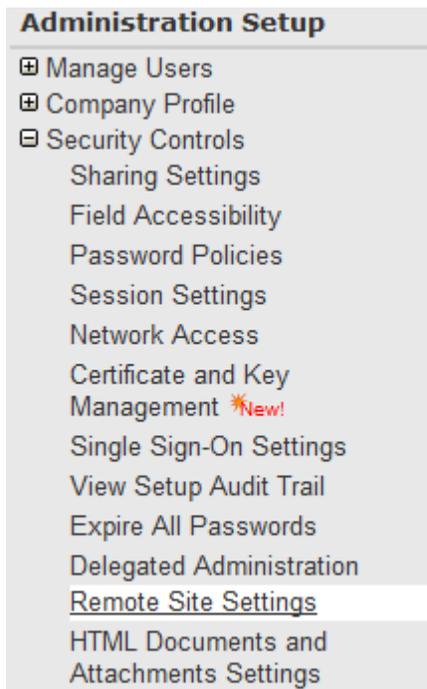


Click on the **Customize My Tabs** button to display the **Customize My Tabs** configuration settings. Select the **Nexternal** tab from the list of available tabs, then click the  button to add the Nexternal Commerce Connector to your list of **Selected Tabs** and **Save** to save your configuration.



Add a Remote Site Setting

Click on the **Setup** link at the top of your Salesforce account page, then open **Administration Setup** > **Security Controls** and click the Remote Site Settings link.



Click the **New Remote Site** button, and enter the following information on the **Remote Site Edit** page, and **Save** to save your configuration.

Remote Site Edit		Save	Save & New	Cancel
Remote Site Name	<input type="text" value="Nexternal"/>			
Remote Site URL	<input type="text" value="https://www.nexternal.com"/>			
Disable Protocol Security	<input type="checkbox"/> 			
Description	<input type="text"/>			

Importing Your Orders

Before you can import your storefront orders into Salesforce you will need to visit the Nexternal Order Management System and enable the Salesforce Commerce Connector under Settings > Compatible Software.

Please note that the export starting date might have to be reset by editing the date displayed in the Sync Start Date control if you have been testing the Salesforce Commerce Connector prior to installing it in your production Salesforce environment.

Information about which orders, customers, and products have already been retrieved from your storefront by the Nexternal Commerce Connector is stored on the Nexternal servers. You can reset this information by logging into your Nexternal Order Management System and changing the Sync Start Date under Settings > Compatible Software > Salesforce.

Once the Sync Start Date has been changed you can log back into Salesforce and click the button to restart the import from the Sync Start Date you have selected.

By default the Nexternal Commerce Connector will initially set the import synchronization start date to cover the previous six months' worth of orders in your storefront. If you would like to change the start date to retrieve more (or less) orders you can do so from the Nexternal Order Management System, under Settings > Compatible Software > Salesforce.

Salesforce Commerce Connector  [Free Trial](#) [Install](#)

Use Salesforce Commerce Connector

Sync Start Date: 
 (estimated storage use: 1.83%)

Export orders based on: ▼

Note that the percentage of **estimated storage use** is a very rough approximation of the actual amount of storage that will be used by your initial import into your Salesforce account. The calculation is based on an estimated 20MB of storage per Salesforce account; the actual amount of storage space you have will depend on the Salesforce Edition you are using. You can check the storage limit from your Salesforce account Setup page and

adjust the percentages accordingly.

The screenshot shows the 'Administration Setup' sidebar on the left with 'Data Management' selected. Under 'Data Management', 'Storage Usage' is highlighted. The main content area is titled 'Storage Usage' and contains the text 'Your organization's storage usage is listed below.' Below this text is a table with two columns: 'Storage Type' and 'Limit'. The table lists 'Data Storage' and 'File Storage', both with a limit of 20.0 MB. Below the table is a section titled 'Current Data Storage Usage' with a table listing 'Record Type' for 'Opportunities' and 'Contacts'.

Storage Type	Limit
Data Storage	20.0 MB
File Storage	20.0 MB

Record Type
Opportunities
Contacts

Remember that most Salesforce editions come with a fixed amount of storage space, and any orders you import will have to share that space with all of the other CRM-related tasks that you are performing. **Please exercise prudence (and caution!) when you chose a synchronization start date.**

By default only the orders that are flagged as "Shipped" or "Sent" are exported from your storefront. If you prefer to export new ("Unshipped") orders as they come into your storefront you can change the "Export orders based on:" value to "Purchase Date".

Export orders based on:

Once you have configured the Salesforce Commerce Connector settings in the Nexternal Order Management System you're ready to being your initial import from your Salesforce account. Sign on to your Salesforce account, select the Nexternal tab, and enter your Nexternal Account Name and your Nexternal login and password in the Connection Credentials fields.

Select the Salesforce User you would like to **Assign as Owner** during the import process, then click to start importing the order, customer, and product information from your storefront.

Connection Credentials

Nexternal Account Name

User Name

Password

Assign as Owner

Enable automatic updates (once per hour)

Once you have completed the initial synchronization you can automate all future imports by selecting the **Enable automatic updates (once per hour)** checkbox.

Note that the *Enable Automatic Updates* checkbox will not be displayed until after you have completed your first successful synchronization!

What Happens During An Import

The Nexternal Commerce Connector is designed to use the standard fields in your Salesforce environment for storing the information imported from your Nexternal storefront. The import process itself is sequential (a section of imported data is read, a Salesforce record is created or updated and written back to the Salesforce database), and does not use any "event-driven" coding to complete any operations.

When data from your Nexternal storefront is imported into Salesforce the following relationships will be created.

- In general, a new Nexternal customer record will create a new Salesforce Account record and a new Salesforce Contact record that is associated with the Account. The only exceptions to this one-to-one mapping are:
 - If an existing Salesforce Account "Account Name" field matches the Nexternal customer Company name the existing Salesforce Account record will be used, and the Nexternal customer will be added to the Account record as a Salesforce Contact.
 - If an existing Salesforce Account contains a Contact record that has an email address that matches the Nexternal customer email address the existing Salesforce Contact record will be used.
- A Nexternal order record will create a new Salesforce Opportunity record.

- The order line items in the Nexternal order record will create new Salesforce OpportunityLineItem records, PricebookEntry records, and Product2 records that are associated with the Opportunity record.
- Any Nexternal customer activities will create new Salesforce Task records.

Note: If you are using the [NexternalFields.properties](#) configuration file as described in the [Nexternal Custom Field Properties](#) section the node value(s) will be written to the field(s) you have assigned in the properties file.

The Nexternal Commerce Connector is designed to always treat orders, customers, and products that are imported from the Nexternal storefront in this manner so that a one-to-one relationship between the data in the Nexternal Order Management System and the Salesforce environment is maintained. Please see the [Issues Encountered During An Import](#) section below for more information.

The Nexternal Commerce Connector will always try to preserve any existing Salesforce Account and/or Contact information, even if it means a technically incorrect assignment or association is created during the import.

For example, assume there is a pre-existing Salesforce Contact record for John Doe with the email address "test@test.com". An order from Jane Smith is imported from your Nexternal storefront that has the email address "test@test.com". After the import the order (Salesforce Opportunity) will be associated with John Doe. A Salesforce Contact record for Jane Smith will not be created, and the customer information for Jane Smith will not overwrite the Salesforce Contact information for John Doe.

Note: If you are going to merge customers in your Nexternal storefront, always try to merge the records into an existing Salesforce Contact record that has already been imported by the Nexternal Commerce Connector and assigned a "Nexternal" Lead Source value.

Contact Detail

Edit

Contact Owner	Demo User Demo User [Cha
Name	Max Martin
Account Name	Smart Kids Software
Title	
Department	
Birthdate	
Reports To	View Org Chart
Lead Source	Nexternal (1362-171)
Mailing Address	14303 Shannon Ridge Houston, TX 77062

- 165 from Allens, Inc., Ed
- Group #4 Select All/None**
- 144 Weise, Bill and Heidi
- 13 Gallagher, Mike
- 171 Martin, Max
- 186 Continental, Esmeralda
- Group #5 Select All/None**

If you have different requirements for data imported into your Salesforce environment please contact your Nexternal Account Manager. We can provide the source code for the addin to your developer or consultant so that they can tailor the import process to meet your needs.

The Imported Data

The import comes into the Nexternal Commerce Connector as an XML document, and certain fields in the document are used by the addin to prevent duplicate records from being created in your Salesforce account. An example of the Nexternal SalesforceReply is located in your Static Resources area (Salesforce Setup > App Setup Section > Develop > Static Resources) under the name [SalesforceReply](#), and can be used as a reference for the fields described below.

<AccountID>

Nexternal storefront number combined with the Nexternal customer number (e.g. 1362-1084). The value in the AccountID node is written to the Salesforce Account.AccountId field, and is used to find existing Nexternal customers that have been created during a previous import. If this field is changed or removed a new Salesforce Account record will be created the next time the customer number is part of an import. There are two exceptions to this condition:

- (1) If the value in the <Account><Company> node matches an existing Salesforce Account.Name field and the Account.AccountId field contains a Nexternal company identifier (a "C" followed by an integer number) the matching Account record will be used, and a new Contact record will be created (or an existing record, as per exception (2) below, will be used).
- (2) If a matching Account record isn't found a secondary search will be performed for any Salesforce Contact records that have a Contact.Email field that matches the value contained in the imported <Account><EMail> node. If a match is found the Salesforce Account record associated with the matching Contact record will be used instead.

A modified version of the AccountID node is also used in the Salesforce Contact.LeadSource field to identify Contact records that have been created during an import and/or associated with an existing Salesforce Contact record during an import. If the Contact.LeadSource field is changed or removed a new Salesforce Contact record will be created the next time the customer number is part of an import (or an existing record, as per exception (2) above, will be used).

<OpportunityID>

Nexternal storefront number combined with the Nexternal order number (e.g. 1362-4148). The value in the OpportunityID node is written to the Salesforce Opportunity.Name field, and is used to find existing Nexternal orders that have been created during a previous import. If this field is changed or removed a new Salesforce Opportunity record will be created the next time the order customer number is part of an import.

The OpportunityID node value is also used to associate the Salesforce Opportunity with one or more Salesforce OpportunityLineItem records (OpportunityLineItem.OpportunityId), one or more Salesforce Task records (Task.WhatId), and with the Salesforce Contact that placed the original order in the Nexternal storefront (OpportunityContactRole.OpportunityId).

<LineItemID>

Nexternal storefront number combined with the Nexternal line item number (e.g. 1362-2174). The value in the LineItemID node is written to the Salesforce OpportunityLineItem.Description field, and is used to find existing Nexternal order line items that have been created during a previous import.

The LineItemID node value is also used to associate the Salesforce Opportunity with one or more Salesforce OpportunityLineItem records.

<ProductCode>

Nexternal storefront number combined with the Nexternal product number (e.g. 1362-32218). If the product has a product-level SKU value defined the SKU

value will be used instead of the product number (e.g. 1362-ff062). If the product is a gift certificate the gift certificate code will be used (e.g. 1362-fisher2320).

Note: If the "UseSKULevelCode" key is set to "true" in the [NexternalFields.properties](#) configuration file the individual SKU-level items from your storefront orders will be used to create Product records on Salesforce. Please contact your Nexternal Account Manager for additional information about this setting, as it might not be suitable for use with your storefront.

The value in the ProductCode node is written to the Salesforce Product2.ProductCode field, and is used to find existing Nexternal products that have been created during a previous import. If this field is changed or removed a new Salesforce Product2 record will be created the next time the product number is part of an import.

The ProductCode node value is also used to associate the Salesforce OpportunityLineItem with a Salesforce Product2 record via the Salesforce PricebookEntry record.

Note: Do not change or alter the Pricebook information for products imported into Salesforce from your Nexternal storefront. Nexternal products need to be associated with the Salesforce Standard Pricebook.

Note that changing the ProductCode value after an import might result in reassignment of any existing Salesforce linkages between new and existing OpportunityLineItems and loss of the PricebookEntry record linkages.

The Import Process

After each batch of data is imported the information within the XML document is parsed and processed as follows. Note that each step below will use the information described in [The Imported Data](#) section to retrieve pre-existing records if possible, and the Salesforce records that are created or updated during each step are committed to the Salesforce database before proceeding to the next step.

- (1) The <Account> node is selected, and a Salesforce Account record is retrieved or created.
- (2) All <Product> nodes within the <Account> node are selected, and Salesforce Product2 records are retrieved or created.
- (3) All <Opportunity> nodes within the <Account> node are selected, and Salesforce Opportunity records are retrieved or created. The <Product> nodes are reselected so that a relationship between the Opportunity and the Product2 records can be established via the Salesforce PricebookEntry record.
- (4) All <OpportunityLineItem> nodes within the <Account> node are selected, and Salesforce OpportunityLineItem records are retrieved or created. The <Opportunity> and <Product> nodes are reselected so that

a relationship between the OpportunityLineItem and PricebookEntry records can be established.

- (5) The selected <Account> node is used to retrieve or create a Salesforce Contact record, and the Contact record is used to establish the relationship between the Account and the Contact via the Salesforce AccountContactRole record and the relationship between the Contact and the Opportunity via the Salesforce OpportunityContactRole record.
- (6) The <ActivityHistory> node is selected, and all <Activity> nodes within the <ActivityHistory> node are added as Salesforce Task records.

Issues Encountered During An Import

Each of the preceding steps performs an update on your Salesforce data, so if you have any custom triggers, rules, filters, or third-party addins in your Salesforce environment that are "listening" for change events on Salesforce objects you might encounter problems due to data fields, dependencies, or relationships that have yet to be created.

If this is the case we *strongly* recommend that you install the Nexternal Commerce Connector to your Salesforce environment sandbox and conduct some test-runs before installing the addin in your production environment. In some cases the issues you might encounter during your tests can be mitigated by creating custom fields in your Salesforce environment and mapping one or more Nexternal fields to these custom fields. Please see the [Advanced Configuration](#) section below for more information.

If you have any of these kinds of interaction issues with the Nexternal Commerce Connector please contact your Nexternal Account Manager. We can provide the source code for the addin to your developer or consultant so that they can tailor the import process to meet your needs.

Importing Multiple Storefronts

If desired, the Nexternal Commerce Connector can import orders from multiple storefronts into a single Salesforce environment. All of the information related to an order (e.g. Account, Contact, Opportunity, etc.) is imported with a unique storefront number, which is used to create unique records in your Salesforce environment.

The only exception to this behaviour is for Nexternal customer records that have a Company name (see the "Company Field" options in the Nexternal Order Management System > Customers > Types > Edit Customer & Activity Types (Page 2)). If the same Company name is used in more than one of your Nexternal storefronts only one Company-type Salesforce Account record will be created and used for all Customers (Salesforce Contacts) that are imported. Please see [What Happens During An Import](#) for more information about how Company Account records are created.

If you want to import multiple storefronts using the automatic update

feature, each of the storefronts must be configured using a separate Salesforce Login, so that unique Salesforce CronTrigger tasks can be created to manage the information required by the addin when scheduling an import for each storefront.

In the example below, two different Salesforce Users ("First User" and "Second User") have logged into Salesforce, selected the "Nexternal" addin tab, entered the Connection Credentials for a Nexternal demonstration storefront ("fairway" and "winery"), and after successfully performing the initial synchronization they have checked the "Enable automatic updates" checkbox to create two new Scheduled Jobs within the Salesforce environment.

Administration Setup

- ☑ Manage Users
- ☑ Company Profile
- ☑ Security Controls
- ☑ Communication Templates
- ☑ Translation Workbench
- ☑ Data Management
- ☑ Monitoring
 - Imports
 - Outbound Messages
 - Time-Based Workflow
 - Automated Process Actions
 - Case Escalations
 - API Usage Notifications
 - Mass Emails
 - Debug Logs
 - Scheduled Jobs
 - Apex Jobs
 - Bulk Data Load Jobs

All Scheduled Jobs
[Help for this Page ?](#)

The All Scheduled Jobs page lists all of the jobs scheduled by your users. Multiple job types may display on this page. You can delete scheduled jobs if you have the permission to do so.

View: All Scheduled Jobs ▾ [Create New View](#)

A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | Other **All**

Action	Job Name ▲	Submitted By	Submitted	Started	Next Scheduled Run	Type
Del	Nexternal Import (fairway)	User, First	11/15/2012 3:22 PM	11/16/2012 8:22 AM	11/16/2012 9:22 AM	Scheduled Apex
Del	Nexternal Import (winery)	User, Second	11/15/2012 3:22 PM	11/16/2012 8:22 AM	11/16/2012 9:22 AM	Scheduled Apex

Advanced Configuration (Optional)

Nexternal Custom Field Settings

The Nexternal Commerce Connector is designed to use the standard fields in your Salesforce environment for storing the information imported from your Nexternal storefront. If you have existing CRM data entered into your Salesforce account, or if you intend to use the Nexternal Commerce Connector in conjunction with another Salesforce application or AppExchange addin, you might need to create a Nexternal Custom Fields configuration file.

The illustrations below in the [Salesforce Objects](#) section outline the Salesforce objects and standard fields used by the Nexternal Commerce Connector, and include a brief description of the storefront data that will be imported into each field. An example of the Nexternal SalesforceReply Object (XML) will be copied into your Static Resources area (Salesforce Setup > App Setup Section > Develop > Static Resources) under the name [SalesforceReply](#) during the installation of the Nexternal Commerce Connector, and can be used as a reference for the fields described below in the Nexternal Response Node columns.

The section following this one, [Nexternal Custom Field Properties](#), describes the contents of the Java "properties" file that can be used to change the Salesforce object field(s) where your Nexternal information will be stored.

Salesforce Objects

Salesforce Account Object

App Setup

- Customize
 - Tab Names and Labels
 - Home
 - Activities
 - Campaigns
 - Leads
 - Accounts
 - Fields**
 - Related Lookup Filters
 - Validation Rules
 - Triggers
 - Partner Roles
 - Contact Roles
 - Page Layouts
 - Field Sets New!
 - Search Layouts
 - Buttons and Links
 - Record Types
 - Account Teams
 - Contacts
 - Opportunities
 - Quotes
 - Forecasts
 - Cases
 - Entitlement Management
 - Self-Service
 - Call Center

Set History Tracking

Action	Field Label	Field Name	Data Type
	<u>Account Name</u>	Name	Name
Edit	<u>Account Number</u>	AccountNumber	Text(40)
Edit	<u>Account Owner</u>	Owner	Lookup(User)
Edit	<u>Account Site</u>	Site	Text(80)
Edit	<u>Annual Revenue</u>	AnnualRevenue	Currency(18, 0)
	<u>Billing Address</u>	BillingAddress	Address
	<u>Created By</u>	CreatedBy	Lookup(User)
Edit	<u>Description</u>	Description	Long Text Area(32000)
Edit	<u>Employees</u>	NumberOfEmployees	Number(8, 0)
Edit	<u>Fax</u>	Fax	Fax
Replace Edit	<u>Industry</u>	Industry	Picklist
	<u>Last Modified By</u>	LastModifiedBy	Lookup(User)
Replace Edit	<u>Ownership</u>	Ownership	Picklist
Edit	<u>Parent Account</u>	Parent	Lookup(Account)
Edit	<u>Phone</u>	Phone	Phone
Replace Edit	<u>Rating</u>	Rating	Picklist
	<u>Shipping Address</u>	ShippingAddress	Address
Edit	<u>SIC Code</u>	Sic	Text(20)
Edit	<u>Ticker Symbol</u>	TickerSymbol	Content(20)
Replace Edit	<u>Type</u>	Type	Picklist
Edit	<u>Website</u>	Website	URL(255)

Nexternal Response Node	Salesforce Object and Field	Description
Account/AccountID	Account.AccountNumber	Customer Number (1)
Account/Company	Account.Name	Company Name (2)
Account/Name	Account.Name	Customer Name
Account/EMail	Account.Site	Email Address
Account/Billing/Street	Account.BillingStreet	Billing Address
Account/Billing/City	Account.BillingCity	Billing City
Account/Billing/State	Account.BillingState	Billing State
Account/Billing/Zip	Account.BillingPostalCode	Billing Zip Code
Account/Billing/Country	Account.BillingCountry	Billing Country
Account/Shipping/Street	Account.ShippingStreet	Shipping Address
Account/Shipping/City	Account.ShippingCity	Shipping City
Account/Shipping/State	Account.ShippingState	Shipping State
Account/Shipping/Zip	Account.ShippingPostalCode	Shipping Zip Code
Account/Shipping/Country	Account.ShippingCountry	Shipping Country
Account/CustomerType	Account.Type	Customer Type (3)
Account/Description	Account.Description	Age Verification (4)
Account/Phone	Account.Phone	Phone Number

(1) If the Salesforce Account.Type field is "Residential" the Salesforce

Account.AccountNumber field will contain the Nexternal Customer Number. If the Salesforce Account.Type is "Business" the Salesforce.AccountNumber field will contain an auto-generated (and auto-incremented) numeric value prefixed with the letter "C". This auto-generated value is used by the Nexternal Commerce Connector to query for and select "Business" accounts that can be used as the parent Account for one or more Contacts. See (2) and (3) below for more information about how Salesforce Accounts and Contacts are processed.

- (2) If the Company Name field in a Nexternal Customer record is empty (blank) the Nexternal Commerce Connector will create a Salesforce Account and a matching Salesforce Contact using the Nexternal Customer record. Both Salesforce objects will be labelled with the Customer Name.

If the Company Name field is not empty the Nexternal Commerce Connector will try to find a Salesforce Account that matches the Company Name, and, if one is found, it will create a Salesforce Contact and attach it to the existing Salesforce Company Account. If a matching Account is not found a new Salesforce Account based on the Nexternal Customer information will be created using the Company Name, and a new Salesforce Contact will be created and attached to the new Salesforce Account.

- (3) The Nexternal Customer Type is used in the Salesforce AccountContactRole.Role field. When a new Salesforce Account object is created the Account.Type field is set to either "Business" or "Residential", depending on whether or not a Company Name is involved.
- (4) If Age Verification is enabled in the storefront Order Management System the Age Verification Flag and Age Verification Notes are passed to the Salesforce Account.Description field.

See also: [What Happens During An Import](#)

Salesforce Contact Object

App Setup

- Customize
 - Tab Names and Labels
 - Home
 - Activities
 - Campaigns
 - Leads
 - Accounts
 - Contacts**
 - Fields**
 - Related Lookup Filters
 - Validation Rules
 - Triggers
 - Page Layouts
 - Field Sets New!
 - Search Layouts
 - Buttons and Links
 - Record Types
 - Opportunities
 - Quotes
 - Forecasts
 - Cases
 - Entitlement Management
 - Self-Service
 - Call Center
 - Contracts
 - Solutions
 - Products
 - Partners
 - Salesforce to Salesforce
 - Customer Portal
 - Ideas
 - Answers
 - Assets
 - Users
 - Console
 - Salesforce CRM Content
 - Content Deliveries

Set History Tracking

Contact Standard Fields

Action	Field Label	Field Name	Data Type
Edit	Account Name	Account	Lookup(Account)
Edit	Assistant	AssistantName	Text(40)
Edit	Ast. Phone	AssistantPhone	Phone
Edit	Birthdate	Birthdate	Date
Edit	Contact Owner	Owner	Lookup(User)
	Created By	CreatedBy	Lookup(User)
Edit	Department	Department	Text(80)
Edit	Description	Description	Long Text Area(32000)
Edit	Do Not Call	DoNotCall	Checkbox
Edit	Email	Email	Email
Edit	Email Opt Out	HasOptedOutOfEmail	Checkbox
Edit	Fax	Fax	Fax
Edit	Fax Opt Out	HasOptedOutOfFax	Checkbox
Edit	Home Phone	HomePhone	Phone
	Last Modified By	LastModifiedBy	Lookup(User)
Edit	Last Stay-in-Touch Request Date	LastCUREquestDate	Date/Time
Edit	Last Stay-in-Touch Save Date	LastCUUpdateDate	Date/Time
Replace Edit	Lead Source	LeadSource	Picklist
	Mailing Address	MailingAddress	Address
Edit	Mobile	MobilePhone	Phone
	Name	Name	Name
Edit Replace	Salutation	Picklist	
	First Name	Text(40)	
	Last Name	Text(80)	
	Other Address	OtherAddress	Address
Edit	Other Phone	OtherPhone	Phone
Edit	Phone	Phone	Phone
Edit	Reports To	ReportsTo	Lookup(Contact)
Edit	Title	Title	Text(80)

Nexternal Response Node	Salesforce Object and Field	Description
Contact/Name/LastName (1)	Contact.LastName	Customer Last Name
Contact/Name/FirstName	Contact.FirstName	Customer First Name
Contact/Mailing/Street	Contact.MailingStreet	Billing Address
Contact/Mailing/City	Contact.MailingCity	Billing City
Contact/Mailing/State	Contact.MailingState	Billing State
Contact/Mailing/Zip	Contact.MailingPostalCode	Billing Zip Code
Contact/Mailing/Country	Contact.MailingCountry	Billing Country
Contact/Phone	Contact.Phone	Phone Number
Contact/Email	Contact.Email	Email Address (2)
Contact/LeadSource	Contact.LeadSource	Nexternal (Customer Number) (3)

(1) Note that in this instance the Nexternal Response Node column is a mapping placeholder for Contact-level information that actually comes from the "Account/*" nodes. See the notes for [Salesforce Account](#)

[Object](#) for more information.

- (2) When a Nexternal Customer record is initially imported into a Salesforce account the Nexternal Commerce Connector will use the Contact/Email node to try and find a matching (pre-existing) Salesforce Contact. If the same email address is used for more than one Contact (or more than one Nexternal Customer) only the first match found will result in a binding between that specific Nexternal Customer and that specific Salesforce Contact; all subsequent Nexternal Customers sharing that email address will create new Salesforce Contact objects.
- (3) The Salesforce Contact.Leadsource is used to store the source of the lead (your Nexternal storefront) and the Nexternal Customer Number.

See also: [What Happens During An Import](#)

Salesforce Opportunity Object

App Setup

- Customize
 - Tab Names and Labels
 - Home
 - Activities
 - Campaigns
 - Leads
 - Accounts
 - Contacts
 - Opportunities
 - Fields
 - Validation Rules
 - Triggers
 - Contact Roles
 - Competitors
 - Page Layouts
 - Field Sets **New!**
 - Search Layouts
 - Buttons and Links
 - Sales Processes
 - Record Types
 - Big Deal Alert
 - Opportunity Team Selling
 - Update Reminders

[set history tracking](#)

Opportunity Standard Fields

Action	Field Label	Field Name	Data Type
Edit	<u>Account Name</u>	Account	Lookup(Account)
Edit	<u>Amount</u>	Amount	Currency(16, 2)
Edit	<u>Close Date</u>	CloseDate	Date
	<u>Created By</u>	CreatedBy	Lookup(User)
Edit	<u>Description</u>	Description	Long Text Area(32000)
Edit	<u>Expected Revenue</u>	ExpectedRevenue	Currency(16, 2)
Edit	<u>Forecast Category</u>	ForecastCategoryName	Picklist
	<u>Last Modified By</u>	LastModifiedBy	Lookup(User)
Replace Edit	<u>Lead Source</u>	LeadSource	Picklist
Edit	<u>Next Step</u>	NextStep	Text(255)
Edit	<u>Opportunity Name</u>	Name	Text(120)
Edit	<u>Opportunity Owner</u>	Owner	Lookup(User)
Edit	<u>Primary Campaign Source</u>	Campaign	Lookup(Campaign)
Edit	<u>Private</u>	IsPrivate	Checkbox
Edit	<u>Probability (%)</u>	Probability	Percent(3, 0)
Edit	<u>Quantity</u>	TotalOpportunityQuantity	Number(16, 2)
Replace Edit	<u>Stage</u>	StageName	Picklist
Replace Edit	<u>Type</u>	Type	Picklist

Nexternal Response Node	Salesforce Object and Field	Description
Opportunity/OpportunityID	Opportunity.Name	Order Number
Opportunity/CloseDate	Opportunity.CloseDate	Ship Date
Opportunity/NextStep	Opportunity.NextStep	Payment Method and P.O. Number
Opportunity/StageName	Opportunity.StageName	Status (always "Closed Won")
Opportunity/Type	Opportunity.Type	Placed By
Opportunity/LeadSource	Opportunity.LeadSource	Lead Source (always "Nexternal")
Opportunity/OrderDate	Opportunity.Description(1)	Creation Date
Opportunity/OrderNet	Opportunity.Description	Net Price
Opportunity/Coupon	Opportunity.Description	Coupon Discount(s)
Opportunity/Discount	Opportunity.Description	Other Discount(s)
Opportunity/Tax	Opportunity.Description	Tax
Opportunity/Shipping	Opportunity.Description	Shipping
Opportunity/Amount	Opportunity.Description	Total Price
Opportunity/Comments	Opportunity.Description	Customer Comments

- (1) Note that the last eight fields are concatenated and stored in the Opportunity.Description field by default. See [Nexternal Custom Field Properties](#) for more information.

Salesforce OpportunityLineItem Object

- Opportunities
 - Fields
 - Validation Rules
 - Triggers
 - Contact Roles
 - Competitors
 - Page Layouts
 - Field Sets New!
 - Search Layouts
 - Buttons and Links
 - Sales Processes
 - Record Types
 - Big Deal Alert
 - Opportunity Team Selling
 - Update Reminders
 - Settings
 - Similar Opportunities New!
- Opportunity Products
 - Fields

Opportunity Product Standard Fields			
Action	Field Label	Field Name	Data Type
	<u>Created By</u>	CreatedBy	Lookup(User)
Edit	<u>Date</u>	ServiceDate	Date
Edit	<u>Discount</u>	Discount	Percent(3, 2)
	<u>Last Modified By</u>	LastModifiedBy	Lookup(User)
Edit	<u>Line Description</u>	Description	Text(255)
Edit	<u>List Price</u>	ListPrice	Currency(16, 2)
Edit	<u>Opportunity</u>	Opportunity	Lookup(Opportunity)
Edit	<u>Product</u>	Product2	Lookup(Product)
Edit	<u>Product Code</u>	ProductCode	Text(255)
Edit	<u>Quantity</u>	Quantity	Number(10, 2)
Edit	<u>Sales Price</u>	UnitPrice	Currency(16, 2)
Edit	<u>Subtotal</u>	Subtotal	Currency(16, 2)
Edit	<u>Total Price</u>	TotalPrice	Currency(16, 2)

Nexternal Response Node	Salesforce Object and Field	Description
OpportunityLineItem/Quantity	OpportunityLineItem.Quantity	Line Item Quantity
OpportunityLineItem/ServiceDate	OpportunityLineItem.ServiceDate	Line Item Ship Date
OpportunityLineItem/UnitPrice	OpportunityLineItem.UnitPrice	Line Item Unit Price
OpportunityLineItem/LineItemID	OpportunityLineItem.Description(1)	Line Item Number
OpportunityLineItem/ShipTo	OpportunityLineItem.Description	Line Item ShipTo Address
OpportunityLineItem/TrackingNo	OpportunityLineItem.Description	Line Item Tracking Number

- (1) Note that the last three fields are concatenated and stored in the OpportunityLineItem. Description field by default. See [Nexternal Custom Field Properties](#) for more information.

Salesforce Product2 Object

Product Standard Fields			
Action	Field Label	Field Name	Data Type
Edit	Active	IsActive	Checkbox
	Created By	CreatedBy	Lookup(User)
	Last Modified By	LastModifiedBy	Lookup(User)
Edit	Product Code	ProductCode	Text(255)
Edit	Product Description	Description	Text Area(4000)
Replace Edit	Product Family	Family	Picklist
Edit	Product Name	Name	Text(255)

Nexternal Response Node	Salesforce Object and Field	Description
Product2/ProductName	Product2.Name	Product Name
Product2/ProductCode	Product2.ProductCode(1)	Product Number
Product2/Family	Product2.Family	Product Category
Product2/Vendor	Product2.Description(2)	Product Vendor
Product2/Description	Product2.Description	Product Description (abbreviated)

- (1) See <Product Code> in [The Imported Data](#) for more information about alternative field values.
- (2) Note that the last two fields are concatenated and stored in the Product2.Description field by default. See [Nexternal Custom Field Properties](#) for more information.

Nexternal Custom Field Properties

If you are using the Nexternal Commerce Connector with a brand new Salesforce account, and your Salesforce account will never be used for anything except importing order, customer, and product information from your Nexternal storefront into the Salesforce CRM, then

You DO NOT need to have a `NexternalFields.properties` file!

Please do not proceed beyond this point, and please do not follow any of the steps or procedures outlined below!

The information in this section is intended for any combination of the following:

- (1) Companies and users that have an existing Salesforce account, and that Salesforce account already contains CRM data.
- (2) Companies and users that are signing up with Salesforce and intend to use additional addins from the Salesforce AppExchange in combination with the Nexternal Commerce Connector addin.
- (3) Companies and users that already are using or will be using any kind of programming service(s) to modify their Salesforce data and / or views (e.g. triggers, rules, etc.).

The content of the next section should be copied in its entirety and pasted into a plain-ASCII text file named `NexternalFields.properties`. This is the configuration file you will modify and upload to the Static Resources area (Salesforce Setup > App Setup Section > Develop > Static Resources) as `NexternalFields` if you want to create your own Custom Fields and use them for storing Nexternal data.

The properties file itself serves as the documentation for what kind of changes can be made. Remember that you must create your Salesforce Custom field(s) **first**, and then transfer the Custom Fields API Name(s) to the Salesforce object field(s) **to the right of the period**.

Information about creating Salesforce Custom Fields can be found in the [Force.Com Workbook](#) (PDF) and [How do I create a custom field?](#) (video).

`NexternalFields.properties`

```
# START OF PROPERTIES
#
# In general, the keys below represent nodes in the Nexternal XML response and
# the values represent Salesforce objects and fields. The value will be 'read'
# from the XML node as a String, converted to the appropriate Salesforce data
# type, and stored into the Salesforce object.
#
# (1) Values on the left-hand side of the equal sign (the 'key') cannot be
#     changed in any way. These are the nodes in the XML response, and are used
```

```

# by the Nexternal addin to locate and retrieve data.
#
# (2) Values on the right-hand side of the equal sign (the 'value') represent
# the Salesforce object and field where data from the XML node will be
# stored.
#
# (a) The Salesforce object on the left-hand side of the period (e.g.
# 'Account.') CANNOT be changed.
#
# (b) The Salesforce object field on the right-hand side of the period
# (e.g. 'AccountNumber') CAN be changed to a custom field that you have
# created within your Salesforce object, but it is YOUR RESPONSIBILITY
# to ensure the custom field is of the SAME SALESFORCE DATA TYPE as the
# field it will replace. The following Salesforce DisplayTypes are
# currently supported in the Nexternal addin: Currency, Date, Double,
# Integer, String, and PickList.
#
# (3) The values stored in the following nodes cannot be altered in any way:
#
# Account/AccountID
# Contact/LeadSource
# Opportunity/OpportunityID
# OpportunityLineItem/LineItemID
# Product2/ProductCode
#
# The information in these fields is used to identify Nexternal information
# imported into Salesforce so that changes to a customer, order, or product
# in your Nexternal Order Management System will be accurately reflected in
# your Salesforce account.
#
# All of the above identification values (except for OpportunityLineItem/
# LineItemID) must be mapped to a Salesforce field that contains ONLY the
# Nexternal record value, so that SOQL queries can be used to retrieve
# Salesforce records that match the records imported from your Nexternal
# Order Management System.
#
# (4) Almost all Salesforce objects have required fields. For example, the Account
# object requires an entry for the Account.Name field, so if you change the
# Account/Name node to point to a Salesforce field other than Account.Name the
# Nexternal addin will write the value of Account/Name to your custom field
# AS WELL AS the Account.Name field to fulfill the Salesforce requirement for
# that field.
#
# Note that the above condition will only be applied to NEW Salesforce objects
# that are created by the Nexternal addin during an import.
#
Account/AccountID=Account.AccountNumber
Account/Company=Account.Name
Account/Name=Account.Name
Account/EMail=Account.Site
Account/Billing/Street=Account.BillingStreet
Account/Billing/City=Account.BillingCity
Account/Billing/State=Account.BillingState
Account/Billing/Zip=Account.BillingPostalCode
Account/Billing/Country=Account.BillingCountry
Account/Shipping/Street=Account.ShippingStreet
Account/Shipping/City=Account.ShippingCity
Account/Shipping/State=Account.ShippingState
Account/Shipping/Zip=Account.ShippingPostalCode
Account/Shipping/Country=Account.ShippingCountry
Account/CustomerType=Account.Type
Account/Description=Account.Description
Account/Phone=Account.Phone
#
# Salesforce Contact records are created using the child nodes of the Account

```

```

# container node from the Nexternal XML response.
#
Contact/Name/LastName=Contact.LastName
Contact/Name/FirstName=Contact.FirstName
Contact/Mailing/Street=Contact.MailingStreet
Contact/Mailing/City=Contact.MailingCity
Contact/Mailing/State=Contact.MailingState
Contact/Mailing/Zip=Contact.MailingPostalCode
Contact/Mailing/Country=Contact.MailingCountry
Contact/Phone=Contact.Phone
Contact/EMail=Contact.Email
Contact/LeadSource=Contact.LeadSource
#
Opportunity/OpportunityID=Opportunity.Name
Opportunity/CloseDate=Opportunity.CloseDate
Opportunity/NextStep=Opportunity.NextStep
Opportunity/StageName=Opportunity.StageName
Opportunity/Type=Opportunity.Type
Opportunity/LeadSource=Opportunity.LeadSource
#
# The directions below apply to the following section ONLY.
#
# The values for the next 8 nodes are normally combined into one String as
# 'key: value' pairs separated by linefeeds and stored in the Opportunity.
# Description field on Salesforce. You can change any (or all) of the values
# below to store the XML node value in a different (or custom) Salesforce
# field within the Opportunity object, and any remaining nodes mapped to
# Opportunity.Description will be combined into the String and stored.
#
Opportunity/OrderDate=Opportunity.Description
Opportunity/OrderNet=Opportunity.Description
Opportunity/Coupon=Opportunity.Description
Opportunity/Discount=Opportunity.Description
Opportunity/Tax=Opportunity.Description
Opportunity/Shipping=Opportunity.Description
Opportunity/Amount=Opportunity.Description
Opportunity/Comments=Opportunity.Description
#
OpportunityLineItem/Quantity=OpportunityLineItem.Quantity
OpportunityLineItem/ServiceDate=OpportunityLineItem.ServiceDate
OpportunityLineItem/UnitPrice=OpportunityLineItem.UnitPrice
#
# The directions below apply to the following section ONLY.
#
# The values for the next 3 nodes are normally combined into one String as
# 'key: value' pairs separated by linefeeds and stored in the OpportunityLineItem.
# Description field on Salesforce. You can change any (or all) of the values
# below to store the XML node value in a different (or custom) Salesforce
# field within the OpportunityLineItem object, and any remaining nodes mapped to
# OpportunityLineItem.Description will be combined into the String and stored.
#
OpportunityLineItem/LineItemID=OpportunityLineItem.Description
OpportunityLineItem/ShipTo=OpportunityLineItem.Description
OpportunityLineItem/TrackingNo=OpportunityLineItem.Description
#
Product2/ProductName=Product2.Name
Product2/ProductCode=Product2.ProductCode
Product2/Family=Product2.Family
#
# The directions below apply to the following section ONLY.
#
# The values for the next 2 nodes are normally combined into one String as
# 'key: value' pairs separated by linefeeds and stored in the Product2.
# Description field on Salesforce. You can change any (or all) of the values

```

```
# below to store the XML node value in a different (or custom) Salesforce
# field within the Product2 object, and any remaining nodes mapped to
# Product2.Description will be combined into the String and stored.
#
Product2/Vendor=Product2.Description
Product2/Description=Product2.Description
#
# Additional Configuration Settings: Contact your Nexternal Account Manager before
# enabling any of the following configuration values.
#
# UseSKULevelCode=true
#
# END OF PROPERTIES
```