

---

# Salesforce Order Management Implementation Guide for B2C Commerce

Salesforce, Spring '24

'24





# CONTENTS

- ORDER MANAGEMENT FOR B2C COMMERCE ..... 1**
- Prepare Your Org ..... 1
- Configure Order Management ..... 3
- Configure Fulfillment ..... 8
- Configure Payment Processing ..... 10
- Configure Access and Permissions ..... 12
- Integrate Order Management with B2C Commerce ..... 17
- Import Historical Order Data ..... 25



# ORDER MANAGEMENT FOR B2C COMMERCE

Order Management helps retailers fulfill, manage, and service orders with real-time inventory, payment and invoicing, order allocation, and fulfillment features. This guide describes the steps to implement Order Management with B2C Commerce.

## [Prepare Your Org](#)

Configure person accounts, currencies, and other org settings to work with your product.

## [Configure Order Management](#)

Configure the features and flows of Order Management.

## [Configure Fulfillment](#)

Configure Order Management objects to support your order fulfillment process. We offer sample fulfillment packages that include flows, processes, and Apex classes that show how to implement Order Management functionality with Salesforce Flow. You can customize them or use them as inspiration to create your own flows and processes.

## [Configure Payment Processing](#)

Processing payments is an integral part of the order management process. To let Order Management execute payment authorizations, captures and issue refunds, integrate a payments component. You can view payment activity in the order payment summary for each order.

## [Configure Access and Permissions](#)

Configure field-level security settings and users and permissions.

## [Integrate Order Management with B2C Commerce](#)

The provided integration imports order data from B2C Commerce and creates records in Salesforce, but doesn't do any other processing. For example, if you import orders from another source, you're responsible for checking for duplicate data.

## [Import Historical Order Data](#)

After you start using Order Management in production, you can import historical order data from a legacy system for reference purposes. This functionality is available only with an Order Management Starter or Growth license.

## Prepare Your Org

---

Configure person accounts, currencies, and other org settings to work with your product.

### [Prerequisites](#)

Make sure that you've met the prerequisites before implementing Order Management.

### [Enable Admin Settings for Order Management](#)

Order Management requires specific admin settings.

### [Enable Person Accounts \(Optional\)](#)

You can represent shoppers using person accounts instead of using contacts on standard business accounts. Enabling person accounts is optional.

### [Enable Multiple Currencies \(Optional\)](#)

To use multiple currencies, enable multiple currencies and then add each currency.

## Prerequisites

Make sure that you've met the prerequisites before implementing Order Management.

- Make sure that Lightning Experience is enabled. Order Management isn't supported on Salesforce Classic or Mobile. You can view record pages on Mobile, but can't use Order Management features.
- Purchase and activate an Order Management license. A license is included with a developer org or when you're using a partner-approved Order Management template ID to create a trial org.
- If you're using Salesforce Omnichannel Inventory, your corresponding Order Management org can integrate with only one instance of B2C Commerce. If you want to integrate with more than one B2C Commerce instance, log a case with Salesforce Support.
- For general information about Order Management concepts, objects, and records, see [Salesforce Order Management Help](#).

## Enable Admin Settings for Order Management

Order Management requires specific admin settings.

1. From Setup, in the Quick Find box, find and select **Order Settings**.
2. Select **Enable Orders**, **Enable Negative Quantity**, **Enable Zero Quantity**, **Enable Enhanced Commerce Orders**, and **Enable Order Events**.
3. Deselect **Enable Reduction Orders**  
Reduction orders can conflict with Order Management change orders. If you use reduction orders elsewhere, we recommend creating an order record type for Order Management orders. Use an Apex trigger or validation rule to prevent the record type from being used for reduction orders. See [Tailor Business Processes to Different Users](#).
4. (Optional) To use orders in Salesforce while managing price books in an external platform, select **Enable Optional Price Books for Orders**.
5. Save the settings.
6. Log out of Salesforce, and log back in.
7. From Setup, in the Quick Find box, search for and select **Order Management**.
8. Activate **Order Management**.  
Turning on **Order Management** can take several minutes. If you don't see the Order Management app in the App Launcher, wait a few minutes and refresh the page.

## Enable Person Accounts (Optional)

You can represent shoppers using person accounts instead of using contacts on standard business accounts. Enabling person accounts is optional.

 **Warning:** Enabling person accounts is irreversible and changes your database structure. Carefully consider the implications. See [Person Accounts](#).

1. Enable Person Accounts.
  - a. From Setup, in the Quick Find box, find and select **Person Accounts**.
  - b. Go through the steps listed on the Setup page.
  - c. From Setup, in the Quick Find box, search for and select **Order Management** and turn on Person Accounts.  
Turning on person accounts can take up to 1 hour.

2. From Setup, in the Quick Find box, find and select **Profiles**.
3. In the list of profiles, select **System Administrator**.
4. Scroll to the Record Type Settings section, find Accounts, and click **Edit**.
5. In the Selected Record Types section, move **Master** to the Available Record Types box, and move **Business Account** and **Person Account** to the Selected Record Types box.

 **Note:** Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid affecting customer implementations.

6. Set both the Default Record Type and the Business Account Default Record Type to **Business Account**.
7. Set the Person Account Default Record Type to **Person Account**.
8. Click **Save**.

## Enable Multiple Currencies (Optional)

To use multiple currencies, enable multiple currencies and then add each currency.

 **Warning:** After you enable multiple currencies, you can't disable it.

1. From Setup, in the Quick Find box, find and select **Company Information**.
2. Click **Edit**.  
You can click either Edit button.
3. Set the Currency Locale to the default currency that you want to use.
4. Select **Activate Multiple Currencies** and save your changes.
5. To add a currency, from Setup, in the Quick Find box, enter *Currencies*, and then select **Manage Currencies** and [enable each currency](#).

SEE ALSO:

[Manage Multiple Currencies](#)

## Configure Order Management

---

Configure the features and flows of Order Management.

### [Configure Delivery Methods](#)

Configure the delivery methods in your org that correspond to your storefront shipping methods. This process ensures that orders are consistently mapped from B2C Commerce to Order Management. Delivery methods must be configured, even when B2C Commerce isn't the source of the order, such as when a customer representative places an order on behalf of a customer. Every storefront shipping method must have a corresponding delivery method.

### [Order Management Flows](#)

Order Management works with Salesforce Flow, which makes it easy for you to create manual and automated functions in your org. It includes flow templates for processes such as order summary creation, placing orders on behalf of shoppers, order fulfillment, and order servicing. Some of the flow templates are available only with a Starter or Growth license.

### Order Lifecycle Statuses

Each object type has status categories with default statuses that reflect the order lifecycle. Find these statuses in Object Manger, in the Statuses section of an object type. You can't change the status categories, but you can customize and add statuses as long as you associate each status with a status category. Having status categories that never change is helpful if you have custom processes that depend on a record's status category.

### Configure Return Reasons for Order Servicing Workflows

The Cancel Item, Discount, Return Item, and Even Exchange flows require you to select a return reason from a dropdown list. Some examples of reasons are *Wrong Item* or *Customer Changed Mind*. Define a list of reasons, and add them as picklist values.

### Configure Return and Rejection Reasons for Return Merchandise Authorization

The RMA Return Items flow requires you to select a return or rejection reason from a dropdown list. Some examples of reasons are *Wrong Item* or *Customer Changed Mind*. Define a list of reasons and add them as picklist values. The RMA flow is available only with an Order Management Growth or Starter license.

### Configure Tax Display

If you only do business in a locale that uses net taxation, such as sales tax, no extra configuration is needed. If you do business in a locale that uses gross taxation, such as value-added tax (VAT), you need to configure your page layouts to display amounts that combine prices and taxes. If you're using multiple currencies, add a user profile for service agents and create page layouts that use gross tax fields.

## Configure Delivery Methods

Configure the delivery methods in your org that correspond to your storefront shipping methods. This process ensures that orders are consistently mapped from B2C Commerce to Order Management. Delivery methods must be configured, even when B2C Commerce isn't the source of the order, such as when a customer representative places an order on behalf of a customer. Every storefront shipping method must have a corresponding delivery method.

1. Create an order delivery method.
  - a. From the App Launcher, find and select **Order Delivery Methods**.
  - b. Click **New**.
  - c. Enter a name.
  - d. For Reference Number, enter the delivery or shipping method ID used in your storefront.

B2C Commerce uses the term "shipping method" instead of "delivery method." The Reference Number value is the shipping method ID used in your B2C Commerce storefront.

You can also add a delivery method that isn't associated with your storefront. In that case, you don't need to assign it a reference number. However, if you do assign the new method a reference number, use a number that doesn't conflict with the shipping method IDs in your storefront.
  - e. Select **Is Active**.
  - f. Optionally, enter a description.
2. Associate either an existing product record to the order delivery method or create a product.

You can create one product record for each delivery method or associate a product record with multiple delivery methods.

  - a. From the dropdown menu, select either an existing product or **New Product**.
  - b. If you selected **New Product**, enter a descriptive name and product code. Select **Active**, and click **Save**.
3. Save your work.
4. Add price book entries.

If Optional Price Books is enabled, this step is optional.

- a. From your new order delivery method, click the link for the associated product.
- b. On the product page, select the **Related** tab.
- c. In the Price Books section, click **Add Standard Price**.
- d. If you have multiple currencies enabled, select a currency from the dropdown.

If your org supports multiple currencies and the Optional Price Books feature isn't enabled, each delivery method product requires a standard price book entry for each supported currency.

- e. For List Price, enter 0. If you leave this field empty, an error occurs.
- f. To add a standard price book entry for another currency, click **Save & New**, and then repeat the steps for the next currency. When you've created entries for all supported currencies, click **Save**.

## Order Management Flows

Order Management works with Salesforce Flow, which makes it easy for you to create manual and automated functions in your org. It includes flow templates for processes such as order summary creation, placing orders on behalf of shoppers, order fulfillment, and order servicing. Some of the flow templates are available only with a Starter or Growth license.

For information about configuring Order Management flows, see [Salesforce Order Management: Configure Order Management Flows](#).

## Order Lifecycle Statuses

Each object type has status categories with default statuses that reflect the order lifecycle. Find these statuses in Object Manger, in the Statuses section of an object type. You can't change the status categories, but you can customize and add statuses as long as you associate each status with a status category. Having status categories that never change is helpful if you have custom processes that depend on a record's status category.

Object Type	Status Category	Default Statuses
Fulfillment Order	Draft	<ul style="list-style-type: none"> <li>• Draft</li> </ul>
	Activated	<ul style="list-style-type: none"> <li>• Allocated</li> </ul>
	Fulfilling	<ul style="list-style-type: none"> <li>• Assigned</li> <li>• Pickpack</li> </ul>
	Closed	<ul style="list-style-type: none"> <li>• Fulfilled</li> </ul>
	Canceled	<ul style="list-style-type: none"> <li>• Canceled</li> <li>• Rejected</li> </ul>
Order	Draft	<ul style="list-style-type: none"> <li>• Draft</li> </ul>
	Activated	<ul style="list-style-type: none"> <li>• Activated</li> </ul>

Object Type	Status Category	Default Statuses
Order Summary  <b>Note:</b> The Order Summary object includes a Status field, but not a Status Category field. Standard Order Management functionality doesn't use order summary statuses, though they can be useful for implementing custom business processes. We recommend that you configure the order summary statuses to match your order statuses.	—	<ul style="list-style-type: none"> <li>Created</li> </ul>
Process Exception	Active	<ul style="list-style-type: none"> <li>New</li> <li>Triaged</li> </ul>
	Inactive	<ul style="list-style-type: none"> <li>Ignored</li> <li>Paused</li> <li>Voided</li> </ul>
	Resolved	<ul style="list-style-type: none"> <li>Resolved</li> </ul>
Return Order	Activated	<ul style="list-style-type: none"> <li>Activated</li> <li>Submitted</li> </ul>
	Canceled	<ul style="list-style-type: none"> <li>Canceled</li> </ul>
	Closed	<ul style="list-style-type: none"> <li>Closed</li> </ul>
	Draft	<ul style="list-style-type: none"> <li>Draft</li> </ul>

## Configure Return Reasons for Order Servicing Workflows

The Cancel Item, Discount, Return Item, and Even Exchange flows require you to select a return reason from a dropdown list. Some examples of reasons are *Wrong Item* or *Customer Changed Mind*. Define a list of reasons, and add them as picklist values.

1. From Setup, in the Quick Find box, find and select **Object Manager**.
2. Select the **Order Product Summary Change** object.
3. Click **Fields & Relationships**.
4. Select **Reason**.
5. To add a new reason to the Reason Picklist Values, click **New**, create your reason, and click **Save**.
6. Select a default value.

- a. Click **Edit** for the value.
- b. Select **Default**.
- c. Click **Save**.

## Configure Return and Rejection Reasons for Return Merchandise Authorization

The RMA Return Items flow requires you to select a return or rejection reason from a dropdown list. Some examples of reasons are *Wrong Item* or *Customer Changed Mind*. Define a list of reasons and add them as picklist values. The RMA flow is available only with an Order Management Growth or Starter license.

1. From Setup, in the Quick Find box, enter *Object Manager*, and then select **Object Manager**.
2. Select **Return Order Line Item**.
3. Click **Fields & Relationships**.
4. Select **Reason for Return**.
5. To add a new reason to the Reason for Return Picklist Values, click **New**, create your reason, and click **Save**.
6. Select a default value.
  - a. Click **Edit** for the value.
  - b. Select **Default**.
  - c. Click **Save**.
7. Save your changes.
8. Verify that all the entries are also in the Reason picklist on Order Product Summary Change.  
All Reason for Return values in the Order Product Summary Change must have a matching value in the Return Order Line Item.
9. Go back to **Return Order Line Item > Fields & Relationships** and select **Reason for Rejection**.
10. Add your rejection reasons to the Reason for Rejection Picklist Values.  
Reasons for rejection are reasons you can't accept the return, such as *Item Is Perishable*.
11. Select a default value.
12. Save your changes.

## Configure Tax Display

If you only do business in a locale that uses net taxation, such as sales tax, no extra configuration is needed. If you do business in a locale that uses gross taxation, such as value-added tax (VAT), you need to configure your page layouts to display amounts that combine prices and taxes. If you're using multiple currencies, add a user profile for service agents and create page layouts that use gross tax fields.

1. From Setup, in the Quick Find box, find and select **Object Manager**.
2. For each object that displays prices, create a page layout that uses gross taxation fields that combine prices and tax amounts. See [Create Page Layouts](#).
3. Edit or clone the Standard User profile.
  - a. From Setup, in the Quick Find box, find and select **Profiles**.
  - b. If you only do business in a locale that uses gross taxation, edit the Standard User profile.

- c. If your org is enabled for multiple currencies, clone the Standard User profile. Give the new profile a name that identifies it as applicable for users working with gross tax.
  - d. Assign the gross taxation page layouts to the user profile.
4. After creating at least one Order Summary record, configure the gross tax fields. If necessary, return to this step later in the implementation process.
5. In Salesforce Order Management, open any Order Summary record details page.
  - a. From the Setup menu on that page, select **Edit Page**.
  - b. Configure the Fields for Gross Tax Orders section of the Totals component.
  - c. Save and activate the page.

## Configure Fulfillment

---

Configure Order Management objects to support your order fulfillment process. We offer sample fulfillment packages that include flows, processes, and Apex classes that show how to implement Order Management functionality with Salesforce Flow. You can customize them or use them as inspiration to create your own flows and processes.

### [Fulfillment Types](#)

Order Management supports three fulfillment categories. Each category is associated with one or more fulfillment types. You can't change the categories, but you can customize the list of types as long as you associate each category with at least one type.

### [Configure Fulfillment Locations](#)

Configure the location object to represent your locations and, optionally, location priorities. Then create a location record for each fulfillment location.

## Fulfillment Types

Order Management supports three fulfillment categories. Each category is associated with one or more fulfillment types. You can't change the categories, but you can customize the list of types as long as you associate each category with at least one type.

To customize fulfillment types, edit the picklist values for the Type field on the Fulfillment Order object.

Fulfillment Category	Default Fulfillment Types
Physical	<ul style="list-style-type: none"> <li>• Warehouse</li> <li>• Retail Store</li> <li>• In Store Pickup</li> </ul>
Drop Ship	<ul style="list-style-type: none"> <li>• Supplier</li> </ul>
Digital	<ul style="list-style-type: none"> <li>• Download</li> <li>• Email</li> </ul>

## Configure Fulfillment Locations

Configure the location object to represent your locations and, optionally, location priorities. Then create a location record for each fulfillment location.

1. From Setup, in the Quick Find box, find and select **Object Manager**.
2. Select **Location > Fields & Relationships > Location Type**.
3. Add your locations to the Location Type Picklist Values.  
Examples of a location type are *Warehouse* or *Retail Store*.
4. Select a default value.
  - a. Click **Edit** for the value.
  - b. Select **Default**.
  - c. Click **Save**.
5. (Optional for customers using routing) Create a location ranking system to prioritize locations.  
If you're using the Order Management license included with D2C Commerce, you can't use a priority ranking system.
  - a. Select **Location > Fields & Relationships > Priority**.
  - b. Add numeric values to the Priority Picklist Values.  
The lower the number, the more preferred the location is.
  - c. Select a default priority value.
6. Save your changes.
7. Create a location record for each fulfillment location.  
If you're using the Order Management license included with D2C Commerce, you can use only one location.
  - a. From the App Launcher, select **Locations**.
  - b. Click **New**.
  - c. Enter a name for the location, and select a value for **Location Type**.
  - d. If you plan to use Salesforce Omnichannel Inventory to manage your inventory locations, select **Sync with OCI** for locations and location groups that are part of your Omnichannel Inventory implementation.
  - e. If you're using a routing and have configured a location ranking system, set the priority for the location.
  - f. Optionally, limit the number of fulfillment orders assigned to a location by defining its fulfillment order capacity. If you're using the Order Management license included with D2C Commerce, these order capacity fields aren't available.
    - **Track Fulfillment Order Capacity**—To track the location's capacity, set this option to *true*. The values of Fulfillment Order Capacity and Assigned Fulfillment Order Count are then used to determine if the location has available capacity.
    - **Fulfillment Order Capacity**—Maximum number of orders that can be assigned to the location. If this value is null, the location's capacity is unlimited.
    - **Assigned Fulfillment Order Count**—To define the time period associated with the maximum capacity for this location, set this value to a certain interval. For example, to track capacity per day, run an automated job that sets this value to 0 for each location at midnight of the location's time zone.
  - g. Save your changes.

- h. Repeat for each location.

SEE ALSO:

[Order Management Flow Core Actions](#)

## Configure Payment Processing

---

Processing payments is an integral part of the order management process. To let Order Management execute payment authorizations, captures and issue refunds, integrate a payments component. You can view payment activity in the order payment summary for each order.

### [Manually Configure a Payment Gateway](#)

If you don't use Salesforce Payments, you can manually configure a payment gateway to view and manage payments in Order Management. Configure a gateway for each payment provider, for example, Stripe, PayPal, Adyen, or Cybersource. Payment packages are available on AppExchange for some vendors.

### [Payment Methods](#)

For B2C Commerce, the Order Management integration with Salesforce Payments supports credit cards, digital wallets, and non-card options, such as SEPA and iDEAL.

### [Define an Alternative Payment Method](#)

You can define a payment method other than a credit card or digital wallet.

### [Set Up Payment Gateways with StoreIntegratedService](#)

Order on Behalf Of can take payment for the order and use the Commerce Webstore Payment Token and Commerce Webstore Checkout Payments APIs. To let your store access these APIs, use the StoreIntegratedService entity to associate your WebStore with the Payment Gateway.

### [Payment Sequencing](#)

You can control the sequence in which to apply refund amounts to payment methods. For example, if an order was partially paid with a gift card, you could refund to the gift card first, regardless of how the payment amounts were originally distributed.

## Manually Configure a Payment Gateway

If you don't use Salesforce Payments, you can manually configure a payment gateway to view and manage payments in Order Management. Configure a gateway for each payment provider, for example, Stripe, PayPal, Adyen, or Cybersource. Payment packages are available on AppExchange for some vendors.

1. Create a payment gateway adapter class using the Salesforce APEX connector framework.  
To get started, see these reference classes: <https://github.com/forcedotcom/Core-Payments-Reference-Gateway-Integration-Adapters>.
2. Create a payment gateway provider record that points to the adapter class that you created.
3. In Setup, define a named credential that contains the gateway provider's authentication and login information.  
The adapter class callout definition calls the named credential.

4. To link the payment gateway provider record and the named credential, create a payment gateway record.

SEE ALSO:

- [Use Cases for the CommercePayments Namespace](#)
- [Named Credentials](#)
- [Commerce Payments Resources](#)
- [CommercePayments Namespace](#)

## Payment Methods

For B2C Commerce, the Order Management integration with Salesforce Payments supports credit cards, digital wallets, and non-card options, such as SEPA and iDEAL.

For a credit card payment, the payment method ID in the order data must match an entry in the Card Type picklist on the Card Payment Method object. If it matches, a card payment method record is created in Salesforce. You can't customize the Card Type picklist.

For a digital wallet payment, the payment method ID in the order data must match this non-case-sensitive regular expression. If it matches, a digital wallet record is created in Salesforce.

```
paypal|visa_checkout|pay_by_check|.*(apple|google|android|amazon|ali).* (pay)*
```

To define a payment method other than a credit card or digital wallet, use the Alternative Payment Method object.

## Define an Alternative Payment Method

You can define a payment method other than a credit card or digital wallet.

1. Create a record type for the alternative payment method object that represents the payment method, for example, Square. For more information, see [Create Record Types](#).

2. Create a `GtwyProvPaymentMethodType` record.

You can't create a `GtwyProvPaymentMethodType` record using the UI. Insert the record using a tool, such as Postman, or Apex code. The API name of the object is `GtwyProvPaymentMethodType`, at the URL `/services/data/version/objects/GtwyProvPaymentMethodType`.

3. Set the `GtwyProvPaymentMethodType` fields.

- Comments—Optional description.
- Developer Name—The unique API name of the record.
- Master Label—A human-readable name for the record.
- Gateway Provider Payment Method Type—The name of the payment method in the storefront. This value must exactly match the payment method ID used in B2C Commerce.
- Payment Gateway Provider Id—Reference to the payment gateway provider record associated with the payment processor for the payment method.
- Payment Method Type—Select **AlternativePaymentMethod**. To use an alternative payment method, first create a corresponding Record Type record.
- Record Type Id—For an alternative payment method, this value is a reference to the record type record.

```
{ "DeveloperName" : "BankTransfer"
  "MasterLabel" : "Bank Transfer",
```

```
"GtwyProviderPaymentMethodType" : "directBanking",
"PaymentGatewayProviderId" : "0cJaa0000000001E67",
"PaymentMethodType" : "AlternativePaymentMethod",
"RecordTypeId" : "012aa000000008A34F" }
```

Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid affecting customer implementations.

## Set Up Payment Gateways with StoreIntegratedService

Order on Behalf Of can take payment for the order and use the Commerce Webstore Payment Token and Commerce Webstore Checkout Payments APIs. To let your store access these APIs, use the StoreIntegratedService entity to associate your WebStore with the Payment Gateway.

1. Obtain the Payment Gateway ID of the existing Payment Gateway used for the site you're connecting to.

```
sf data query --query
"SELECT Id, PaymentGatewayName, PaymentGatewayProviderId, Status
FROM PaymentGateway"
```

2. Ensure the Payment Gateway supports tokenize and authorization. If not, an update is required.
3. Obtain the ID of the WebStore that you're configuring. Add CurrencyIsoCode to multi-currency orgs to get the right ID.

```
sf data query --query
"SELECT Id, ExternalReference, Name, Type, DefaultTaxLocaleType
FROM WebStore"
```

4. Create a StoreIntegratedService entity.

```
StoreIntegratedService sis = new StoreIntegratedService(
    StoreId = '0ZERN00000009qu4AA', // WebStore Id (step 3)
    Integration = '0b0RO000000009FYAQ', // Payment Gateway Id (step 1)
    ServiceProviderType = 'Payment');
insert sis;
```

## Payment Sequencing

You can control the sequence in which to apply refund amounts to payment methods. For example, if an order was partially paid with a gift card, you could refund to the gift card first, regardless of how the payment amounts were originally distributed.

For information about implementing payment sequencing, see [Payment Sequencing](#) in the *Salesforce Order Management Developer Guide*.

## Configure Access and Permissions

---

Configure field-level security settings and users and permissions.

### [Order Management Users and Permissions](#)

To access Order Management features, users require the System Administrator profile, or either the Standard Platform User profile or the Identity User profile along with the Salesforce Order Management feature permission set license. Users also require access to the Order Management console app and to any customizations that you implement. Order Management includes some persona-based permission sets that you can use as is or clone and customize.

#### [Configure Access to the Console App](#)

Provide access to the console app for a permission set.

#### [Configure Field-Level Security](#)

Give the System Administrator profile permissions to access specific fields on order-related objects. These permissions allow users to access Order Management functionality.

#### [Configure Custom Field Permissions](#)

When you add a custom field to an object for use in Order Management, Order Management users need access to that custom field. Update each permission set that includes the custom field.

#### [Give Access to Person Accounts](#)

If you're using Person Accounts to represent shoppers, give access to the appropriate permission sets to access Person Accounts.

## Order Management Users and Permissions

To access Order Management features, users require the System Administrator profile, or either the Standard Platform User profile or the Identity User profile along with the Salesforce Order Management feature permission set license. Users also require access to the Order Management console app and to any customizations that you implement. Order Management includes some persona-based permission sets that you can use as is or clone and customize.

To users assigned to the Identity User profile access Order Management features, configure the tab setting for the Identity User profile for the Account object to Default On. See [View and Edit Tab Settings in Permission Sets and Profiles](#).

## User Access

To view the included permission sets, in the Setup Quick Find box, enter *Permission Sets*, and then select **Permission Sets**.

Permission Set	Description
Order Management Operations Manager (Growth or Starter license only)	Ability to manage day-to-day Order Management processes. Limited ability to modify open fulfillment orders.
Order Management Agent (Growth or Starter license only)	Standard Order Management user. Can only modify an open fulfillment order by advancing its status through the workflow.
Order Management Store Associate (Growth or Starter license only)	Limited access that allows an in-store associate to process fulfillment orders.
Order Management Shopper (Growth or Starter license only)	Limited access for a service user to allow an external system to make self-service requests. Read-only access to fulfillment orders.
1Commerce Order Management User	Ability to manage day-to-day Order Management processes. Limited ability to modify open fulfillment orders. Only cancellation and simple returns allowed. No appeasements or unmanaged orders allowed.

If you use the included permission sets as is, you must also create at least one custom permission set to provide access to the console app and any customizations. Each cloned permission set counts against your org's limits.

You can organize your permission sets in different ways:

- (Recommended) Use one or more permission sets to provide the Order Management permission set license and persona-based permissions, and one permission set for the console app and any customizations. You can use the included permission sets or create custom permission sets with the Order Management permission set license. Assign one Order Management permission set and the custom permission set to each Order Management user.
- Use multiple Order Management permission sets. You can clone the included permission sets or create custom permission sets with the Order Management permission set license. Add the console app permission and all custom permissions to each set. Assign one of the sets to each Order Management user.
- Use a single permission set for all Order Management users. You can clone one of the included permission sets or create a custom permission set with the Order Management permission set license. Add the console app permission and all custom permissions to the set.

 **Important:** If you assign more than one permission set with the Order Management permission set license to a single user, that user consumes multiple Order Management licenses.

For more information, see [Permission Sets](#) and [User Licenses](#) in *Set Up and Maintain Your Salesforce Organization* in Salesforce Help.

## Data Protection and Privacy

Data protection and privacy regulations can require you to mask or delete customers' personal data. However, many fields on Order Management objects that contain personal data is read-only during normal processing. Only users with the System Administrator profile or the Edit Activated Orders permission can modify them. When designing your data protection and privacy policies, be sure to account for this restriction.

For more information, see [Data Protection and Privacy](#).

SEE ALSO:

[Permission Sets](#)

[User Licenses](#)

## Configure Access to the Console App

Provide access to the console app for a permission set.

1. From Setup, in the Quick Find box, search for and select **Permission Sets**.
2. To create a permission set, click **New**. To add permissions to an existing set, select it.
3. If you're creating a permission set:
  - a. Enter a unique label. We recommend *OM Console*.
  - b. Deselect **Session Activation Required**.
  - c. For License, select **None**.
  - d. Click **Save**.
4. In the Apps section, click **Assigned Apps**.
5. Click **Edit**.
6. For Available Apps, select **Order Management (standard\_\_OMConsole)**, and then click **Add**.

7. Click **Save**.
8. Configure the permission set to access Account records.
  - a. Click **Permission Set Overview**.
  - b. In the Apps section, click **Object Settings**.
  - c. Select **Accounts**.
  - d. Click **Edit**.
  - e. If you're using person accounts to represent shoppers, in the Account: Record Type Assignments section, select **Assigned Record Types** all the record types.
  - f. If your org restricts sharing for Account records, in the Object Permissions section, enable **View All**.  
To check whether sharing is restricted for account records, from Setup, in the Quick Find box, enter *Sharing Settings*, and then select **Sharing Settings**.
  - g. Save the changes.
9. Repeat the process for each permission set that you want to provide console app access to.

SEE ALSO:

[Sharing Settings](#)

## Configure Field-Level Security

Give the System Administrator profile permissions to access specific fields on order-related objects. These permissions allow users to access Order Management functionality.

1. If you're using person accounts to represent individual shoppers, verify that they're enabled and configured in your org.
2. From Setup, in the Quick Find box, find and select **Profiles**.
3. Select **System Administrator**.
4. In the Fields & Relationships section, find Order and click **View**.
5. Click **Edit**.
6. Assign Edit Access and Read Access for each object in the table. Some of the permissions and objects aren't available in every version of Order Management. Don't worry. Fill out the available objects and permissions that are visible in your organization.

Object	Read Access	Edit Access
Location (Growth Only)	N/A	<ul style="list-style-type: none"> <li>• Assigned Fulfillment Order Count</li> <li>• Fulfillment Order Capacity</li> <li>• Priority</li> <li>• Track Fulfillment Order Capacity</li> </ul>
Order	<ul style="list-style-type: none"> <li>• Order Amount</li> <li>• Order Management Reference Identifier</li> <li>• Total Adjusted Delivery Amount</li> </ul>	<ul style="list-style-type: none"> <li>• Billing Email Address</li> <li>• Billing Phone Number</li> <li>• Bill To Contact</li> <li>• Order Name</li> </ul>

Object	Read Access	Edit Access
	<ul style="list-style-type: none"> <li>• Total Adjusted Delivery Tax</li> <li>• Total Adjusted Delivery Amount with Tax</li> <li>• Total Adjusted Product Amount</li> <li>• Total Adjusted Product Tax</li> <li>• Total Adjusted Product Amount with Tax</li> <li>• Total Delivery Distributed Adjustments</li> <li>• Total Delivery Distributed Adjustment Tax</li> <li>• Total Delivery Distributed Adjustments with Tax</li> <li>• Total Product Distributed Adjustments</li> <li>• Total Product Distributed Adjustment Tax</li> <li>• Total Product Distributed Adjustments with Tax</li> <li>• Total Tax</li> <li>• Total with Tax</li> </ul>	<ul style="list-style-type: none"> <li>• Order Reference Number</li> <li>• Ordered Date</li> <li>• Sales Channel</li> <li>• Sales Store</li> <li>• Tax Locale Type</li> </ul>
Order Product	<ul style="list-style-type: none"> <li>• Adjusted Product Subtotal</li> <li>• Adjusted Product Subtotal Tax</li> <li>• Adjusted Product Subtotal with Tax</li> <li>• Distributed Order Adjustments</li> <li>• Distributed Order Adjustments Tax</li> <li>• Distributed Order Adjustments with Tax</li> <li>• Gross Unit Price</li> <li>• Product Adjustments with Tax</li> <li>• Total Adjustments</li> <li>• Total Adjustments Tax</li> <li>• Total Adjustments with Tax</li> <li>• Total Amount with Tax</li> <li>• Total Tax</li> </ul>	<ul style="list-style-type: none"> <li>• Line Number</li> <li>• Order Delivery Group</li> <li>• Reserved at Location</li> </ul>
Order Adjustment Group	N/A	<ul style="list-style-type: none"> <li>• Description</li> </ul>
Order Delivery Group	N/A	<ul style="list-style-type: none"> <li>• Deliver to Address</li> </ul>
Shipment (Growth Only)	N/A	<ul style="list-style-type: none"> <li>• Expected Delivery Date</li> </ul>

Object	Read Access	Edit Access
		<ul style="list-style-type: none"> <li>• Fulfillment Order</li> <li>• Return Order</li> <li>• Shipping Provider</li> <li>• Status</li> <li>• Tracking Number</li> <li>• Tracking URL</li> </ul>
Shipment Item	N/A	<ul style="list-style-type: none"> <li>• Return Order Line Item</li> </ul>

## Configure Custom Field Permissions

When you add a custom field to an object for use in Order Management, Order Management users need access to that custom field. Update each permission set that includes the custom field.

1. From Setup, in the Quick Find box, find and select **Permission Sets**.
2. Select the permission set that controls access for your Order Management users.
3. In the Apps section, click **Object Settings**.
4. Select the object that has the custom field.
5. Click **Edit**.
6. In the Field Permissions section, select **Edit Access** for the custom field.
7. Click **Save**.

## Give Access to Person Accounts

If you're using Person Accounts to represent shoppers, give access to the appropriate permission sets to access Person Accounts.

1. From Setup, in the Quick Find box, find and select **Permission Sets**.
2. Select the permission set that controls access for your Order Management users. Typically, it's named OM Console.
3. In the Apps section, click **Object Settings**.
4. Select **Accounts**.
5. Click **Edit**.
6. In the Account: Record Type Assignments section, select both **Business Account** and **Person Account**.
7. Click **Save**.
8. Repeat this process for the OM Management B2C Service permission set, which provides access to Salesforce records.

## Integrate Order Management with B2C Commerce

---

The provided integration imports order data from B2C Commerce and creates records in Salesforce, but doesn't do any other processing. For example, if you import orders from another source, you're responsible for checking for duplicate data.

If you don't use Salesforce Omnichannel Inventory, you can integrate Order Management with multiple instances of B2C Commerce. However, if you use Omnichannel Inventory, Order Management, and B2C Commerce together, your Order Management org can integrate with only one instance of B2C Commerce.

#### [Configure High Scale Order Management](#)

The high-scale system for ingesting orders from B2C Commerce handles much larger order volumes than the regular system. In most cases, we recommend enabling high-scale functionality. If your configuration stops the automated order creation process to add additional data or formatting before creating order summaries, you can't use the high-scale system.

#### [Connect to B2C Commerce for Order Integration](#)

Set up a connection between B2C Commerce and the Salesforce org where Order Management is configured. This connection enables Order Management to receive orders from B2C Commerce.

#### [Web Store Records](#)

Both the Even Exchange and the Order on Behalf Of flows require users to select a web store record that represents their B2C Commerce storefront. Order Management creates the required web store records for you. These system-generated web store records set the LocationType to Virtual. If you want to control the location or change other fields, you can create your own web store records.

#### [Troubleshoot the B2C Commerce Integration](#)

If the integration isn't working, refer to the troubleshooting tips in Salesforce Help.

## Configure High Scale Order Management

The high-scale system for ingesting orders from B2C Commerce handles much larger order volumes than the regular system. In most cases, we recommend enabling high-scale functionality. If your configuration stops the automated order creation process to add additional data or formatting before creating order summaries, you can't use the high-scale system.

For instructions on configuring high-scale Order Management, see [Enable High-Scale Orders for Order Management](#).

## Connect to B2C Commerce for Order Integration

Set up a connection between B2C Commerce and the Salesforce org where Order Management is configured. This connection enables Order Management to receive orders from B2C Commerce.

#### [Connect B2C Commerce and Order Management](#)

Set up a connection between B2C Commerce and the Salesforce org where Order Management is configured. This connection enables Order Management to receive orders from B2C Commerce and get product and pricing data for creating and updating orders.

#### [Enable Order Management for B2C Commerce](#)

When you're ready to send order data from B2C Commerce to Order Management, use Business Manager to enable the integration.

#### [Organization-Wide Sharing Defaults](#)

The B2C Integration service user is an internal Salesforce user. If you set internal organization-wide sharing defaults for an object that needs to be accessible to the integration, set the defaults to Public Read/Write access.

#### [Create an API Client](#)

Creating a dedicated API client ID for your Order Management integration facilitates troubleshooting issues and gathering metrics because you can more easily identify the actions associated with the client ID. Creating a dedicated API client is optional. You can also use an existing API client and share it with other functions.

### [Use Matching Rules for Shopper Account Identification](#)

Order Management identifies an existing shopper by matching the shopper's name or email address to an existing Account, Contact, or Person Account record. However, you can configure Order Management to use your org's duplicate and matching rules instead when looking for existing shoppers.

### [Add a Sales Channel](#)

Create Sales Channel records to represent the sales channels that provide orders to your org.

### [State and Country Codes](#)

The values that you use in your store order data must match in Order Management. The values that you can use depend on whether you enable state and country/territory picklists in Salesforce.

### [Custom B2C Commerce Storefront Data Integration](#)

You can pass custom data on certain B2C Commerce objects to Order Management by adding custom attributes to objects in B2C Commerce and matching custom fields to the corresponding objects in Order Management. When an attribute ID and field name match, and the data types also match, the values are included in the created Order Management records. If you also create a matching field on the corresponding summary object, Order Management includes the custom data when it creates a summary record.

### [Update a B2C Permission Set After Adding a Custom Field](#)

The Order Management B2C Service permission set provides access to Salesforce records. When you add a custom field to a Salesforce object, update the permission set by adding read and edit access to the new field. You also must add edit access to the permission set that provides access for your Order Management users.

### [Product Bundle Associations](#)

Order Management doesn't support custom attributes on B2C product options. To use product bundles, you can pass the information as custom data.

## Connect B2C Commerce and Order Management

Set up a connection between B2C Commerce and the Salesforce org where Order Management is configured. This connection enables Order Management to receive orders from B2C Commerce and get product and pricing data for creating and updating orders.

If you've changed your Business Manager certificate, open a Salesforce Support case to update the B2C Commerce URL before connecting Order Management and B2C Commerce. You can use a custom URL for B2C Commerce only in a production instance.

1. Log in to Salesforce as a System Administrator.
2. From Setup, in the Quick Find Box find and select **B2C Commerce Connections**.  
If the link doesn't appear, try logging out and back in again.
3. On another tab,
4. Click **New Connection**.
5. In the Connection Request section, enter the connection name and the B2C instance ID.  
Find the B2C instance ID in Business Manager. Go to **Administration > Global Preferences > Salesforce Order Management Integration Administration** and copy the tenant ID.
6. Click **Save**.
7. In the Allow Tenant Linkage window, click **Allow**.
8. Move the Available Connection Type that you want to use to Selected Connection Type.

Option	Description
<b>Get orders from B2C Commerce</b>	Select <b>Order Integration</b>

Option	Description
<b>Get product and pricing data for creating and updating orders</b>	Select <b>Cart and Checkout Data</b> (Requires a Growth license)
<b>Connect your instance to core</b>	select <b>Order Self-Service</b>

9. Click **Save** and confirm the connection request.
10. Wait to receive an email at your System Administrator email address indicating that the provisioning is complete.
11. In the Tenant Connections section, find your connection.
  - If the connection was provisioned successfully, it appears with Status Provisioned.
  - If the provisioning failed, it appears with Status Failed and more information in the Failed Reason column. If you can't solve the issue, open a case with Salesforce Support. Include your B2C Commerce instance ID, Salesforce org ID, the information in the Failed Reason column, and whether the connection is new or you clicked its **Retry** link.
12. In a separate tab, log in to B2C Commerce Business Manager.
13. In B2C Commerce Business Manager, go to **Administration > Global Preferences > Salesforce Order Management Integration Configuration**.
14. Select **Active**.
15. Click **Save**, and in the confirmation window, click **OK**.

To connect multiple instances of B2C Commerce to your Salesforce org, repeat these steps for each B2C Commerce instance.

## Enable Order Management for B2C Commerce

When you're ready to send order data from B2C Commerce to Order Management, use Business Manager to enable the integration.

1. Log in to Business Manager.
2. Select your site.
3. Navigate to **Merchant Tools > Site Preferences > Order**.
4. In the Order Management Settings section, set **Include in Order Management** to **Yes**.
5. Click **Apply**.

Repeat these steps for each instance.

## Organization-Wide Sharing Defaults

The B2C Integration service user is an internal Salesforce user. If you set internal organization-wide sharing defaults for an object that needs to be accessible to the integration, set the defaults to Public Read/Write access.

SEE ALSO:

[Organization-Wide Sharing Defaults](#)

## Create an API Client

Creating a dedicated API client ID for your Order Management integration facilitates troubleshooting issues and gathering metrics because you can more easily identify the actions associated with the client ID. Creating a dedicated API client is optional. You can also use an existing API client and share it with other functions.

1. Log in to Account Manager as an account administrator at <https://account.demandware.com/>.
2. Click **API Client**.
3. Click **Add API Client**.
4. For Display Name, enter a descriptive name.
5. Click **Add**.
6. Click **Add** again.

## Use Matching Rules for Shopper Account Identification

Order Management identifies an existing shopper by matching the shopper's name or email address to an existing Account, Contact, or Person Account record. However, you can configure Order Management to use your org's duplicate and matching rules instead when looking for existing shoppers.

When configuring matching rules, complex rules can affect the integration's performance.

1. From Setup, in the Quick Find box, find and select **Order Management**.
2. Switch the B2C Integration Data Matching Rules option to **Active**.

If your matching rules identify multiple potential duplicate records, Order Management selects the one with the highest match score. If multiple records tie for the highest match score, you can't guarantee which is selected.

## Add a Sales Channel

Create Sales Channel records to represent the sales channels that provide orders to your org.

1. For each B2C Commerce Site, create a Sales Channel record in Salesforce.
2. Set the Sales Channel's Name field to match the Site ID.  
In a B2C Commerce order payload, the Site ID is the `site/site-id` value.

## State and Country Codes

The values that you use in your storefront order data must match in Order Management. The values that you can use depend on whether you enable state and country/territory picklists in Salesforce.

If Salesforce state and country/territory picklists aren't enabled, Order Management copies each code to the State or Country field on records in Salesforce.

- The state code value in your storefront order data must match a standard 2-character ISO state or province code.
- The country code value in your storefront order data must match a standard 2-character ISO country code.

If Salesforce state and country/territory picklists are enabled, Order Management copies each code to the State Code or Country Code field on records in Salesforce. The corresponding State or Country field is set to the matching integration value from the picklist.

- The state code value in your storefront order data must match the state code of an entry in the Salesforce state picklist.

- The country code value in your storefront order data must match the country or territory code of an entry in the Salesforce country/territory picklist.

SEE ALSO:

[Let Users Select State and Country from Picklists](#)

## Custom B2C Commerce Storefront Data Integration

You can pass custom data on certain B2C Commerce objects to Order Management by adding custom attributes to objects in B2C Commerce and matching custom fields to the corresponding objects in Order Management. When an attribute ID and field name match, and the data types also match, the values are included in the created Order Management records. If you also create a matching field on the corresponding summary object, Order Management includes the custom data when it creates a summary record.

Only the B2C Commerce attribute ID and the Order Management field name at the API level are compared. UI labels aren't considered.

If order data includes a custom attribute value but the corresponding Salesforce object has no matching custom field, the attribute is ignored. If a custom field exists in Salesforce and order data doesn't include a matching custom attribute value, the field is ignored.

Also consider datatype compatibility between your storefront and Salesforce. All text values in order data that exceed the size of the corresponding field in Salesforce by three characters less than the datatype size limit and appends three periods (...). For example, if order data includes the value `Albuquerque` for a custom field with a maximum length of 10 characters, the value is stored as `Albuque...` in Salesforce.

**!** **Important:** If a custom field in Salesforce is required, and it corresponds to a custom storefront attribute, order data must include a value for that attribute. Otherwise, Order Management can't create the corresponding Salesforce record and an error occurs.

At the API level, Salesforce custom field names always end in `__c`. Don't include `__c` in the names of the matching attributes on your storefront objects. However, if you include a custom namespace in a custom field name, also include it in the name of the matching storefront object attribute. For example, if the API name of your custom field in Salesforce is `mynamespace_FieldName__c`, name your matching storefront object attribute `mysnamespace_FieldName`. Limit your custom field names to 40 characters (37 plus `__c`).

You can use matching custom attributes and fields on these objects.

We don't support custom object mapping or custom object lookup. If you try to map a custom attribute with a custom object lookup, the mapping fails.

B2C Commerce	Salesforce Order Management
Gift Certificate Line Item, Order Line Item, and Shipping Line Item	OrderItem and OrderItemSummary
Order	Order and Order Summary
Order Payment Instrument and Payment Transaction	AlternativePaymentMethod, CardPaymentMethod, DigitalWallet, Payment, PaymentAuthorization, and PaymentGatewayLog
Shipment	OrderDeliveryGroup and OrderDeliveryGroupSummary
Buyer Address on Order	Account (including PersonAccount) and Contact

SEE ALSO:

[Custom Field Types](#)

## Update a B2C Permission Set After Adding a Custom Field

The Order Management B2C Service permission set provides access to Salesforce records. When you add a custom field to a Salesforce object, update the permission set by adding read and edit access to the new field. You also must add edit access to the permission set that provides access for your Order Management users.

1. From Setup, in the Quick Find box, find and select **Permission Sets**.
2. Select **Order Management B2C Service**.
3. In the Apps section, click **Object Settings**.
4. Select the object that has the custom field.
5. Click **Edit**.
6. Under Field Permissions, select **Edit Access** for the custom field.
7. Click **Save**.
8. From Setup, select **Permission Sets** again.
9. Select the permission set that controls access for your Order Management users. Usually, it's named OM Console.
10. Under Apps, click **Object Settings**.
11. Select the object that has the custom field.
12. Click **Edit**.
13. Under Field Permissions, select **Edit Access** for the custom field.
14. Click **Save**.

## Product Bundle Associations

Order Management doesn't support custom attributes on B2C product options. To use product bundles, you can pass the information as custom data.

To set up product bundle associations, define a custom guid attribute for the line item object in your storefront. In Salesforce, define a matching field on Order Product and Order Product Summary. When creating an order, assign the same value to the custom attribute on line items that belong together. You can define another custom attribute to identify a line item as part of a bundle. Then build logic into your fulfillment process to handle them accordingly.

When using custom attributes to support bundles, customize your return and cancel flows to handle them according to your business rules. For example, design your cancel process so that when you cancel a bundle product, the associated bundle products are also canceled. Likewise, you can design your return process to disallow the return of only some of the products in a bundle.

## Web Store Records

Both the Even Exchange and the Order on Behalf Of flows require users to select a web store record that represents their B2C Commerce storefront. Order Management creates the required web store records for you. These system-generated web store records set the LocationType to Virtual. If you want to control the location or change other fields, you can create your own web store records.

If your store supports multiple currencies, create a web store record for each possible combination of currency and locale that can appear in order data.

An order's currency code and locale ID are matched with the web store IsoCurrencyCode and DefaultLanguage fields, respectively. Thus, in Business Manager Site Settings, you can select only locales that match valid values for the web store DefaultLanguage. You can't select "default." For details about supported languages, see [WebStore](#) in the Object Reference for Salesforce and Lightning Platform.

To create a web store record, you must be assigned the Operations Manager permission set. You can't create a web store with the required values in the Salesforce UI, so use a tool like Postman or Apex code. Define these properties.

### Name

Any unique name

### CurrencyIsoCode

Required for storefronts that support multiple currencies. This value matches the currency code in order data.

### DefaultLanguage

Required for storefronts that support multiple languages. This value matches the locale ID in order data.

### DefaultTaxLocaleType

*NET* or *GROSS*

### ExternalReference

Use the format *instanceId + @ + siteId*. For example, *abc\_123@SiteGenesis*

### LocationId

Optional. If you don't specify a location, the system creates a location and sets this value to its ID. In that case, it sets the new location's LocationType field to Virtual and its ExternalReference field to null. If you include any custom validations for the Location object, they must account for these values.

### SupportedCurrencies

Required for storefronts that support multiple currencies. Include only one value, which must match CurrencyIsoCode.

### SupportedLanguages

Required for storefronts that support multiple languages. Include only one value, which must match DefaultLanguage.

### Type

*B2CE*

Here's an Apex code example for creating a web store.

```
WebStore createWebStore( String name, String instanceId, String siteId ) {
    WebStore store = new WebStore(
        Name = name,
        // Either NET or GROSS
        DefaultTaxLocaleType = 'NET',
        // Format expected is instanceId@siteId; example abc_123@SiteGenesis
        // If multicurrency is enabled then uncomment these lines:
        //CurrencyIsoCode = 'USD',
        //DefaultLanguage = 'en_US',
        //SupportedCurrencies = 'USD',
        //SupportedLanguages 'en_US',
        ExternalReference = instanceId + '@' + siteId,
        Type = 'B2CE');

    return store;
}

WebStore webStoreRecord = createWebStore('Apex1 WebStore', 'abc_123', 'SiteGenesis');
insert webStoreRecord;
```

Here's a multicurrency example. Let's say that you have one site that supports US orders in US dollars. A second site supports UK orders in pounds or euros, and also supports French orders in euros. Create four web stores with these values.

Name	External Reference	Currency IsoCode	Default Language	DefaultTax LocaleType	Supported Currencies	Supported Languages	Type
SiteGenesis USD	abc_123@ SiteGenesis	USD	en_US	NET	USD	en_US	B2CE
SiteGenesis Global GBP en	abc_123@ SiteGenesis Global	GBP	en_GB	GROSS	GBP	en_GB	B2CE
SiteGenesis Global EUR fr	abc_123@ SiteGenesis Global	EUR	fr_FR	GROSS	EUR	fr_FR	B2CE
SiteGenesis Global EUR en	abc_123@ SiteGenesis Global	EUR	en_GB	GROSS	EUR	en_GB	B2CE

## Troubleshoot the B2C Commerce Integration

If the integration isn't working, refer to the troubleshooting tips in Salesforce Help.

See [Troubleshoot the B2C Commerce Integration](#).

## Import Historical Order Data

---

After you start using Order Management in production, you can import historical order data from a legacy system for reference purposes. This functionality is available only with an Order Management Starter or Growth license.

Typically, for orders that are no longer subject to servicing, you import them as unmanaged orders. For information about importing order data, see [Importing Order Data](#).