

**SFCC - ESW Cartridge**

+

INTEGRATION GUIDE

Version: 4.8

Published date: September 2025

Copyright © 2025 by ESW. All rights reserved. Information contained herein is subject to change without notice.



Contents

[1 About This Document 5](#_Toc204601649)

[2 Overview 7](#_Toc204601650)

[2.1.1 Additional information 8](#_Toc204601651)

[3 Supported Features 10](#_Toc204601652)

[3.1 ESW Storefront Components 10](#_Toc204601653)

[3.2 Pricing Integration 10](#_Toc204601654)

[3.3 Shipping Methods Integration 10](#_Toc204601655)

[3.4 Checkout Integration 11](#_Toc204601656)

[3.5 Order Confirmation Integration 11](#_Toc204601657)

[3.6 Catalog Integration 11](#_Toc204601658)

[3.7 Package Integration 11](#_Toc204601659)

[3.8 Order Cancellation Integration 12](#_Toc204601660)

[3.9 Global Product Restriction 12](#_Toc204601661)

[3.10 Global Product Returns Prohibition 12](#_Toc204601662)

[3.11 Promotions Integration 12](#_Toc204601663)

[3.12 Checkout Shopper Registration 12](#_Toc204601664)

[3.13 ESW Share Marketing Opt-In 13](#_Toc204601665)

[3.14 Checkout Metadata Integration 13](#_Toc204601666)

[3.15 Order Appeasement Integraion 13](#_Toc204601667)

[3.16 Global Returns Integration 13](#_Toc204601668)

[3.17 Share service/webhooks logs on retailer platform 13](#_Toc204601669)

[3.18 Over the Counter Payment Method Integration 14](#_Toc204601670)

[3.19 Page Designer integration 14](#_Toc204601671)

[3.20 ESW Integration Monitoring 14](#_Toc204601672)

[3.21 Synchronize Package from ESW Integration 14](#_Toc204601673)

[3.22 Embedded Checkout 14](#_Toc204601674)

[3.23 ESW JWKS Integration 14](#_Toc204601675)

[3.24 Order Status Updates from ESW to SFCC 14](#_Toc204601676)

[4 Working with SFCC Business Manager 16](#_Toc204601677)

[4.1 Merchant Tools 16](#_Toc204601678)

[4.1.1 Site Preferences > Custom Preferences 16](#_Toc204601679)

[4.1.2 Site Preferences > Privacy 17](#_Toc204601680)

[4.1.3 Site Preferences > Order 18](#_Toc204601681)

[4.1.4 Custom Objects 18](#_Toc204601682)

[5 ESW Cartridge Installation 20](#_Toc204601683)

[5.1 Import Metadata 20](#_Toc204601684)

[5.2 Cartridge Setup 25](#_Toc204601685)

[5.2.1 SiteGenesis 25](#_Toc204601686)

[5.2.2 Storefront Reference Architecture 25](#_Toc204601687)

[5.2.3 Headless (OCAPI) 26](#_Toc204601688)

[6 SFCC Reference Information 27](#_Toc204601689)

[6.1 ESW APIs in the Cartridge 27](#_Toc204601690)

[6.1.1 Authentication API 27](#_Toc204601691)

[6.1.2 Pricing Advisor 28](#_Toc204601692)

[6.1.3 ESW Checkout 31](#_Toc204601693)

[6.1.4 Advanced Shipping Notification (ASN) 42](#_Toc204601694)

[6.2 New Countries or Markets Expansion 48](#_Toc204601695)

[6.2.1 Verifying the new country-currency selection 49](#_Toc204601696)

[6.3 Headless (OCAPI) Attributes Mapping with Cart 51](#_Toc204601697)

[6.3.1 ESW Headless (OCAPI) Attributes 51](#_Toc204601698)

[6.3.2 Product Line Item 51](#_Toc204601699)

[6.3.3 Cart (Basket) 51](#_Toc204601700)

[6.3.4 Mapping Examples 52](#_Toc204601701)

[6.4 How to Upgrade? 62](#_Toc204601702)

[6.4.1 Full Version Upgrade 62](#_Toc204601703)

[6.4.2 Recommended end-to-end approach 63](#_Toc204601704)

[7 Go-Live Readiness 65](#_Toc204601705)

[8 SFCC FAQs 67](#_Toc204601706)

[8.1 How does pricing conversion work? 67](#_Toc204601707)

[8.2 How can I add a header bar, footer bar, and welcome landing page? 67](#_Toc204601708)

[8.3 What happens during a failover? 68](#_Toc204601709)

[8.4 What cookies does ESW create? 68](#_Toc204601710)

[8.5 Why can’t I see the sales tax information on the Cart page? 68](#_Toc204601711)

[8.6 How long does it take for an order to be marked as ‘Failed’? 69](#_Toc204601712)

[8.7 Why can’t I see the Shipping Method dropdown on the Cart page? 69](#_Toc204601713)

[8.8 Does SFCC reduce inventory when an order is created during the 'Cart to eSW' handover? 69](#_Toc204601714)

[8.9 Why does an order show as ‘Created’ during ‘Cart to ESW' handover? 69](#_Toc204601715)

[8.10 When does the SFCC native checkout gets displayed to the shoppers? 69](#_Toc204601716)

[8.11 Can only the fixed pricing model be supported without FX rates, uplifts, and rounding data being present in SFCC? 70](#_Toc204601717)

[8.12 Why do discounted prices require page refresh on the Cart page? 70](#_Toc204601718)

[8.13 Why am I stuck on cart page and unable to redirect to ESW Checkout? 70](#_Toc204601719)

[8.14 Why my SFCC order level system attribute values do not match with ESW order confirmation custom attribute? 71](#_Toc204601720)

[8.15 Why don’t I see stored cards on the checkout? 71](#_Toc204601721)

About This Document

Salesforce Commerce Cloud (SFCC) is an ecommerce platform that allows retailers to build and manage online stores. This document describes the end-to-end functionality of the eShopWorld-SFCC cartridge, including features, installation details, Business Manager configuration, and integration.

For an up-to-date version of this document, see the [ESW Knowledge Centre](https://know.eshopworld.com/space/AE/176947412/Salesforce+Commerce+Cloud).

Overview

The Salesforce Commerce Cloud (SFCC) - ESW cartridge allows you to integrate with the ESW services, such as hosted checkout, local payment methods, and international logistics network. The cartridge also enables developers to self-manage site configuration, including the welcome mat configuration, landing page, country & currency widgets, and product prices.

Using the cartridge, you can also configure shipping method prices, fixed price books, enable or disable countries, languages, currencies, and manage other global preferences. The following are some of the key features of the cartridge.

* **Localized Experience**: The cartridge offers functionalities, such as welcome mat and header/footer widgets. These functionalities are GEO IP powered to provide a localized experience to international shoppers. The cartridge also supports multiple currencies per country/locale to enhance the shopper experience.
* **IP Geolocation**: IP geolocation determines the shopper’s location. If geolocation is enabled, the shopper’s country is chosen by default. Shoppers can update the shipping country, language, and/or currency from the welcome mat.
* **Localized Prices**: The cartridge is integrated with ESW’s Pricing Advisor. This allows you to set up product prices for mass markets. The prices include uplifts, FX rates, duties and taxes.
* **Prices Display**: The cartridge maximizes conversion by displaying product prices that can be either inclusive or exclusive of duties and taxes.
* **Duty and Taxes**: For each order, duties and taxes are calculated using ESW’s checkout calculators.
* **Price Books**: The cartridge offers single website support for managing multiple price books. This simplifies managing multiple websites. Additionally, the cartridge allows you to use the master price book and generate prices for various markets at once instead of creating price books individually for each market.
* **Localized Price Books**: You can generate localized price books using the FX rates, uplifts, duties, taxes, and rounding rules from the ESW Pricing Advisor. For more information, see “Localized Pricebooks” section in the “eShopWorld\_Pricing\_Integration.docx” guide.
* **Multiple FX Rates**: You can set up multiple base/settlement currencies per tenant. For more information on this feature, see “Multiple FX Rates Pricing” section in the “eShopWorld\_Pricing\_Integration.docx”.
* **Promotions**: The ESW cartridge supports promotions in the shopper currency. By default, SFCC supports promotions in the default currency only.
* **Shipping Overrides and Shipping Promotion**: For certain countries and regions, the cartridge offers a custom shipping override. You can also offer shipping promotions in selected local markets.
* **Auto Catalog Sync**: The cartridge is integrated with Catalog 2.0, which means that you can send your product catalog details to ESW using the Catalog file transfer method or via an API call. For more information on this feature, see “eShopWorld\_Catalog\_Integration.docx” guide.
* **Auto Sync Package ASN Details in the Hub Model**: The cartridge is integrated with [Package API v4.0](https://know.eshopworld.com/space/DT/177012963/Package+API+Overview) allowing you to send your package details to ESW using an SFCC job.
* **Product Restriction**: The cartridge offers product restrictions, which means that you can restrict the product to certain countries only. For more information on this feature, see “eShopWorld\_Global\_Product\_Restriction.docx” guide.

Additional information

Supported versions

* Site Genesis/Controllers: Supports the latest standard version
* Site Genesis/Pipelines: Core eShopWorld controllers can be used for ESW backend functionality such as preOrder request, order confirmation, order cancellation and validate inventory etc.   
  ESW SFCC plugin does not support the SG pipeline version, however retailer can compare and customize the required pipelines in-line with sitegenesis controllers provided in ESW SFCC plugin.
* SFRA: Supports SFRA v7.0.1 with the compatibility mode 22.7 and under.
  + ESW Supports Backwards compatibility till SFRA version 4.4.0

Partner profile

Click [here](https://www.salesforce.com/products/commerce-cloud/partner-marketplace/partners/eshopworld/)to view the SFCC- ESW partner marketplace.

Roadmap

ESW regularly introduces new features and enhancements to the cartridge. The following is the roadmap of the SFCC-ESW supported services:



Supported Features

ESW Storefront Components

ESW Cartridge supports Welcome mat, footer and header widgets. The welcome mat is a modal window that is displayed when shoppers visit your online store for the first time. Using the welcome mat, shoppers can change their shipping country, language, and currency. When shoppers click the header or footer widget, the welcome mat appears again. The country, language, and currency preferences can then be updated via the welcome mat and applied to the storefront.

For more details, please refer to **“eShopWorld\_Storefront\_Components.docx”** document.

Pricing Integration

ESW Cartridge supports Pricing Integration with eShopWorld:

* Pricing Models (Fixed, Calculated);
* ESW rounding prices (fixed, multiple);
* Multiple FX Rates Pricing (allows to set up multiple base and settlement currencies), Localized Pricebooks (allows you to generate localized pricebooks using the the FX rates, uplifts, duties, taxes, and rounding rules from the ESW Pricing Advisor);
* Localized Promotions (effective way of creating international promotions for countries that use the ESW Fixed Pricing model);
* **Localized Front-end Prices** (the feature used to enhance shopper’s localized experience by converting front-end prices configured in Content Slots/ Assets & Promotional Call-out Messages).

For more details, please refer to **“eShopWorld\_Pricing\_Integration.docx”** document.

Shipping Methods Integration

The shipping methods are displayed on the ESW checkout depending on how the ESW Override Shipping custom preference is set.

For more details, please refer to **“eShopWorld\_Shipping\_Methods\_Integration.docx”** document.

Checkout Integration

The ESW Checkout page is displayed when shoppers click **Checkout** on the Cart page. On the ESW Checkout page, shoppers can enter their delivery address, contact details, and payment details. Shoppers can also select their preferred shipping method.

For more details, please refer to **“eShopWorld\_Checkout\_Integration.docx”** document.

Order Confirmation Integration

The Order Confirmation page is displayed after the order is placed. The screen displays the items that the shopper ordered, the total price they paid, and the delivery and payment addresses.

In case **Enable ESW Inventory Check** preference is enabled in checkout configurations, inventory validation will take place on order confirmation webhook before placing the order.

For more details, please refer to **“eShopWorld\_Order\_Confirmation\_Integration.docx”** document.

Catalog Integration

ESW requires a product catalog from retailers as a part of the onboarding process. A product catalog includes data about the products that you want to sell internationally through the ESW Checkout and ship using the ESW logistics services. The data typically includes Product Code, Product Description, Country of Origin, Weight, HS Code, and so on. Sending the product catalog information is crucial as it allows ESW to accurately calculate the duties & taxes and ensure export compliance and customs clearance.

For more details, please refer to **“eShopWorld\_Catalog\_Integration.docx”** document.

Package Integration

ESW provides a range of models, such as Hub and Ship from Store (Hubless) for retailers to fulfill the orders. Each model has a different order process flow.

The ESW-SFCC cartridge allows retailers to synchronize their shipment/package data with the ESW Package Service through the Hub model.

For more details, please refer to **“eShopWorld\_Package\_Integration.docx”** document.

Order Cancellation Integration

ESW plugin allows retailers to synchronize cancelled orders status between SFCC & ESW regardless of where order cancellation triggered first, whether in SFCC or ESW.

For more details, please refer to “**eShopWorld\_Order\_Cancellation\_Integration.docx**” document.

Global Product Restriction

Considering different needs and availability of products over the global market, if you want to sell a product in a few targeted countries and not all, you can do so using the ESW Custom Product Level attribute. Using this attribute, you restrict the product from being sold in the countries that you specify within the cartridge.

For more details, please refer to **“eShopWorld\_Global\_Product\_Restriction.docx”** document.

Global Product Returns Prohibition

Considering different needs and availability of products over the global market, if you want to prohibit a product return in a few targeted or all countries, you can do so using the ESW Custom Product Level attribute. Using this attribute, you restrict the product from being returned in the countries that you specify within the cartridge.

For more details, please refer to **“eShopWorld\_Global\_Product\_Returns\_Prohibition.docx”** document.

## **Promotions Integration**

Native SFCC promotion engine threshold amount does not work with dynamic pricing. When dynamic price model is in use, localized prices are calculated on the fly in templates and SFCC backend engine does not see them. ESW cartridge allows retailers to unlock these limitation for international markets by using custom promotion attributes.

For more details, please refer to “**eshopWorld\_Promotions\_Integration.docx”**

## **Checkout Shopper Registration**

Following are some benefits of using the checkout shoppers’ registration feature.

This ESW SFCC feature allows for building up an international shoppers database.

Offer international shoppers promotions based on their purchase history.

Allow shoppers to track orders using the logged-in shopper’s dashboard.

Conversion rate of existing customers will be higher due to saving time on checkout.

In addition to checkout shoppers’ registration, the Chances of winning retailers will increase by adding this feature

For more details, please refer to “**eShopWorld\_Checkout\_Shoppers\_Registration\_ Integration.docx”**

## **ESW Share Marketing Opt-In**

This ESW opt-in allows retailers to create separate newsletters or intimidation emails for ESW opted-in shoppers. It also enabled retailers to publish Marketing SMS.There are two optins attributes for Email as well as SMS marketing. Registered Shoppers stored opt-in preferences value can be passed to the pre-order payload, which will turn on opt-in preferences on ESW checkout for already opted-in shoppers.

For more details, please refer to “**eShopWorld\_ESW\_Share\_Marketing\_Opt\_In\_Integration.docx”**

## **Checkout Metadata Integration**

By giving an option to configure Basket and Customer level metadata attributes directly in SFCC to minimize retailers specific code customizations.

For more details, please refer to “**eShopWorld\_Checkout\_Metadata\_Integration.docx”**

## **Order Appeasement Integraion**

* Appeasement integration with SFCC will allow the retailer to access the Appeasement information directly through SFCC and send them to their respective OMS/WMS/financial systems.

For more details, please refer to “**eShopWorld\_Order\_Appeasement\_Integration.docx**

## **Global Returns Integration**

This feature allows retailer to send Package ASN export using SFCC and return orders from ESW. Retailers will be able to return data in SFCC.

For more details, please refer to **eShopWorld\_Global\_Returns\_Integration.docs**

## **Share service/webhooks logs on retailer platform**

* This feature allow retailer to share service health related information with ESW through a specific JSON like file structure.

For more details, please refer to "**eShopWorld\_Specific\_Logs\_On\_Retailers\_Platform**”

## **Over the Counter Payment Method Integration**

* Over the Counter is a payment method that allows consumers to order goods online and pay for them over the counter in a convenience store.

For more details, please refer to"**eShopWorld\_Over\_The\_Counter\_Payment\_Method\_Integration.docx**”

## **Page Designer integration**

* This feature allow retailer to use localize pricing feature along with page designer

For more details, please refer to **”eShopWorld\_Page\_Designer\_Integration.docx”**

## **ESW Integration Monitoring**

* This feature allow retailer to download and share ESW related configuration with ESW through a specific JSON file structure.

For more details, please refer to " **eShopWorld\_Integration\_Monitoring.docx**”

## **Synchronize Package from ESW Integration**

* This feature allows retailers to sync package from ESW. For more details please refer to “eShopWorld\_Synchronize\_Package\_from\_ESW\_Integration.docx”

## **Embedded Checkout**

* This feature allow shoppers to complete checkout process on the retailer’s storefront which eliminates the need for redirecting the shopper to ESW checkout page. For detail integration of the feature please refer to “eShopWorld\_Embedded\_Checkout\_(EC)\_ Integration.docx”

## **ESW JWKS Integration**

* This feature allows retailers to authenticate the webhooks using JWKS. For more details please refer to “eShopWorld\_JWKS\_Authentication\_Integration.docx

## **Order Status Updates from ESW to SFCC**

* This feature allow retailers to sync order’s payment statuses from ESW to SFCC. For detailed integration please refer to “eShopWorld\_Order\_Status\_Updates.docx”.

Working with SFCC Business Manager

Merchant Tools

The **Merchant Tools** section allows you to manage the site-specific aspects of the application. You can also manage the site-specific ESW configuration.

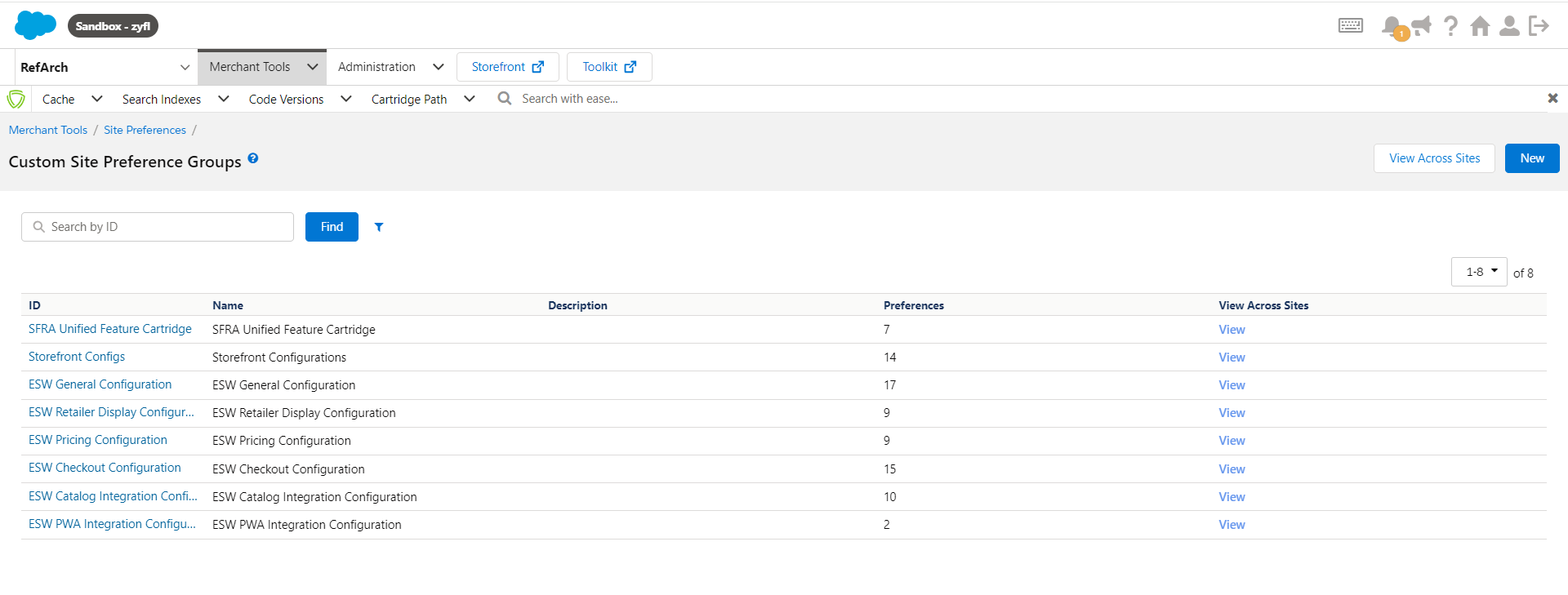
Graphical user interface, table

Description automatically generated with medium confidence

Site Preferences > Custom Preferences

You can use **Custom Preferences** to manage ESW-specific configurations of the cartridge. You can access Custom Preferences by navigating to **Merchant Tools** > **Site Preferences** > **Custom Preferences**.

After importing ESW SFCC plugin metadata mentioned below in **ESW Cartridge Installation,** following ESW configuration groups will be available.

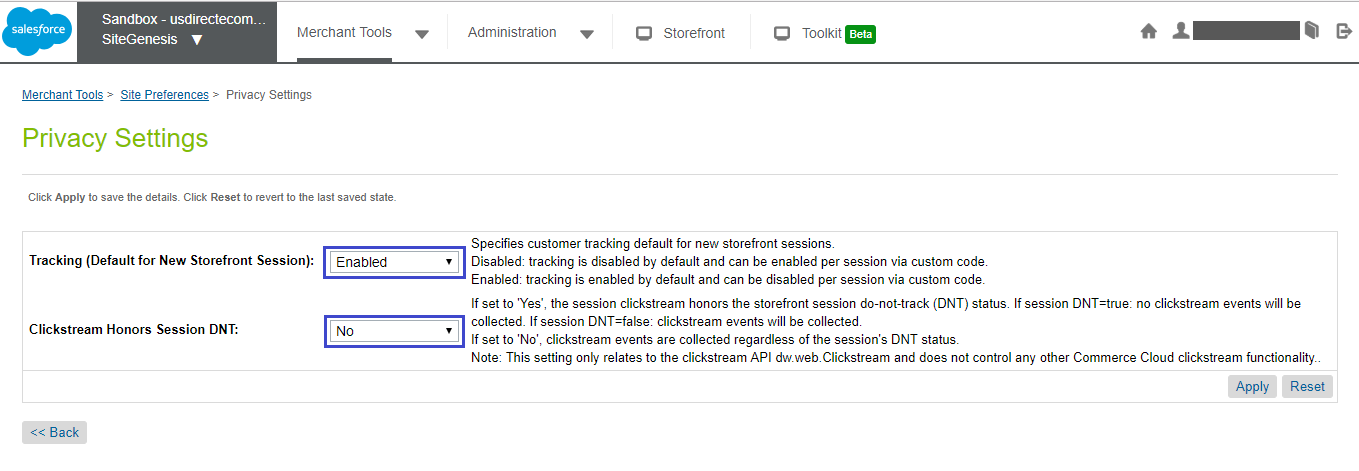


Custom Site Preference Groups

Site Preferences > Privacy

You can access**Privacy** settings by navigating to **Merchant Tools** > **Site Preferences** > **Privacy**.

The following screen is displayed when you click **Privacy**:



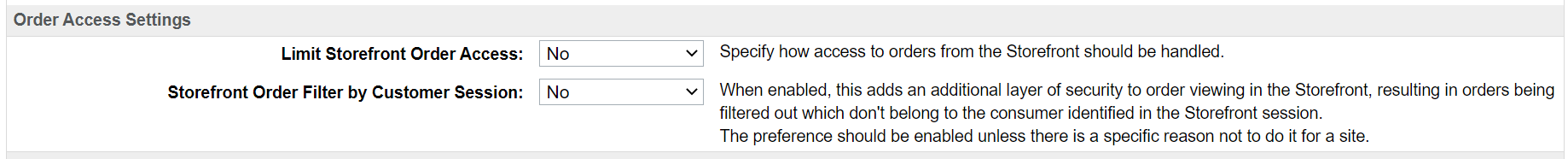
Privacy Settings

The following options are available on this screen:

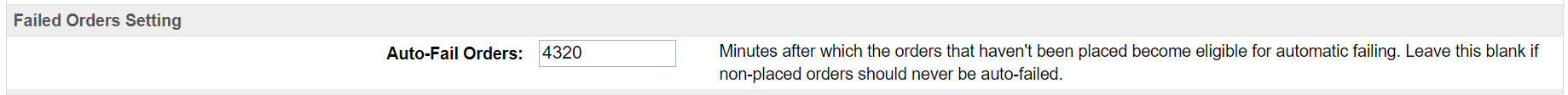
* **Tracking (Default for New Storefront Session)**: This is a mandatory attribute. Select Enabled to enable session tracking.
* **Clickstream Honors Session DNT**: This is a mandatory attribute. The clickstream events are collected regardless of the session's DNT status.

Site Preferences > Order

**Limit Storefront Order Access:** Please set the setting to “NO” (as shown on the screenshot below).



**Failed Orders Settings:** By default, this field is empty, which means that SFCC will not automatically fail orders in CREATED status, so please set it to the desired amount of time (minutes). For example, 4320 (3 days).



### **Custom Objects**

ESW cartridge uses custom objects to store ESW pricing advisor data, supported countries and currencies. This includes the countries’ base currency, default currency, Fx rates, country adjustments, rounding model and pricing model.

After running the **Retailer Pricing Feed** job will populate the ESW\_PA\_DATA data custom objects containing pricing advisor data for the countries configured on the tenant end.

Text

Description automatically generated

*ESW PA Data Custom Object*

Running the **ESW Retailer Auto Configurator** job will populate ESW Countries and ESW Currencies and custom objects with esw supported countries and currencies respectively. This job is also responsible to create ESW\_DIAGNOSTIC\_DATA custom objects which holds the information regarding cartridge version and SFRA version. This job should be scheduled **after** execution of the Retailer Pricing feed job.

Table

Description automatically generated

*ESW Countries Custom Objects*

Table

Description automatically generated

*ESW Currencies Custom Objects*

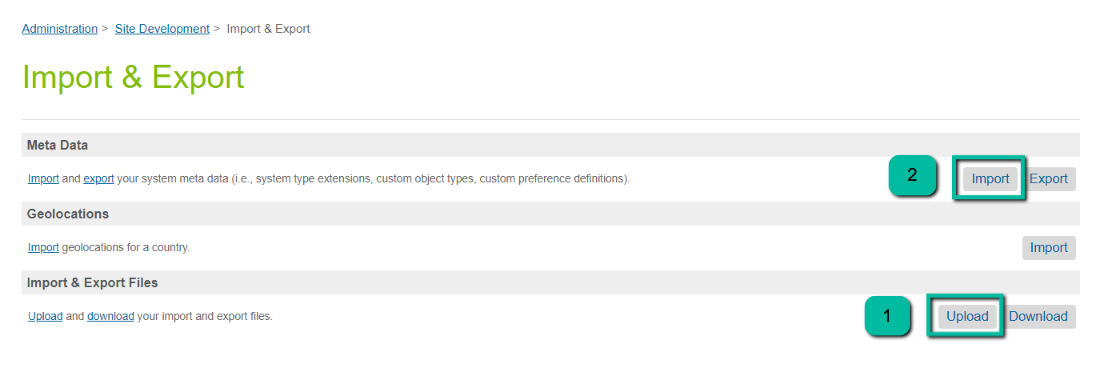
ESW Cartridge Installation

Import Metadata

Before you will start using the ESW cartridge functionality you have to import the metadata files.

|  |  |  |
| --- | --- | --- |
| **Metadata File Name** | **Path** | **Description** |
| customobjects.xml | /link\_eshopworld/Metadata/meta/ | ESW custom objects |
| customobjects\_pwa.xml | /link\_eshopworld/Metadata/meta/ | ESW additional custom objects for PWA |
| systemobjects.xml | /link\_eshopworld/Metadata/meta/ | ESW system objects extensions |
| systemobjects\_headless.xml | /link\_eshopworld/Metadata/meta/ | ESW additional system objects for Headless |
| systemobjects\_pwa.xml | /link\_eshopworld/Metadata/meta/ | ESW additional system objects for PWA |
| services.xml | /link\_eshopworld/Metadata/ | ESW services |
| jobs.xml | /link\_eshopworld/Metadata/ | ESW jobs |
| content-assets.xml | /link\_eshopworld/Metadata/sites/SiteGenesis  or  /link\_eshopworld/Metadata/sites/RefArch | ESW content assets |
| payment-methods.xml | /link\_eshopworld/Metadata/sites/SiteGenesis  or  /link\_eshopworld/Metadata/sites/RefArch | ESW payment methods |
| slots.xml | /link\_eshopworld/Metadata/sites/SiteGenesis  or  /link\_eshopworld/Metadata/sites/RefArch | ESW slots |

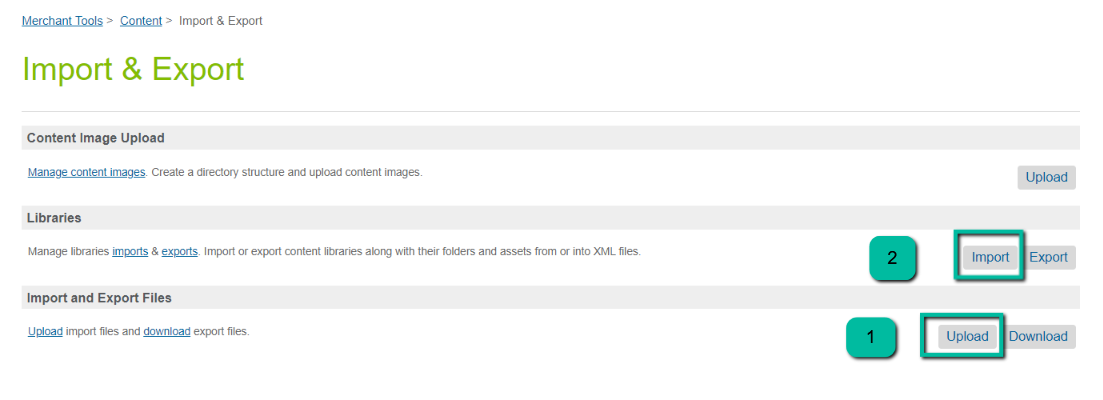
* + 1. Import **customobjects.xml**, **systemobjects.xml**, **systemobjects\_headless.xml.xml** files in in MERGE mode:



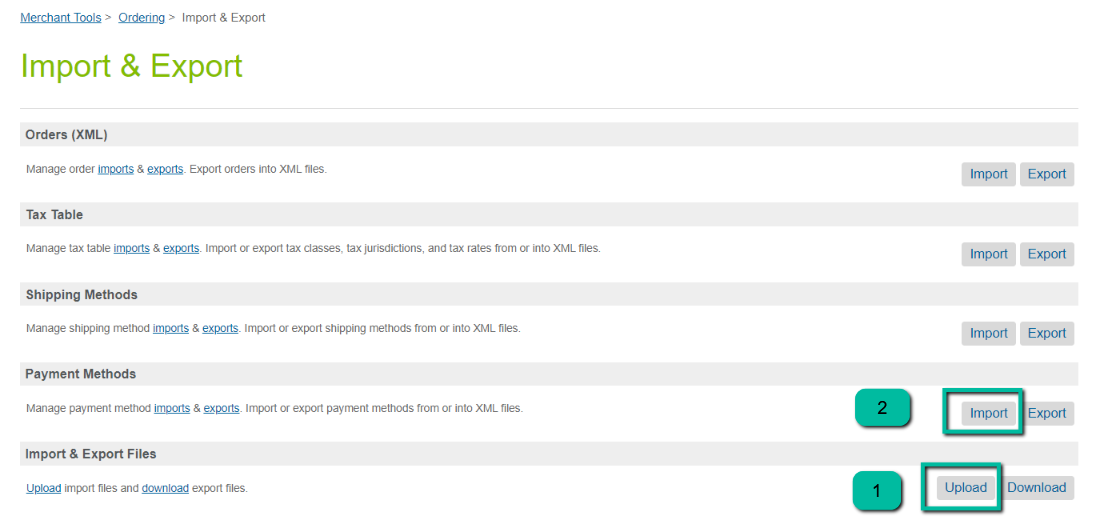
* + 1. Import **services.xml**, **jobs.xml** files in in MERGE mode:

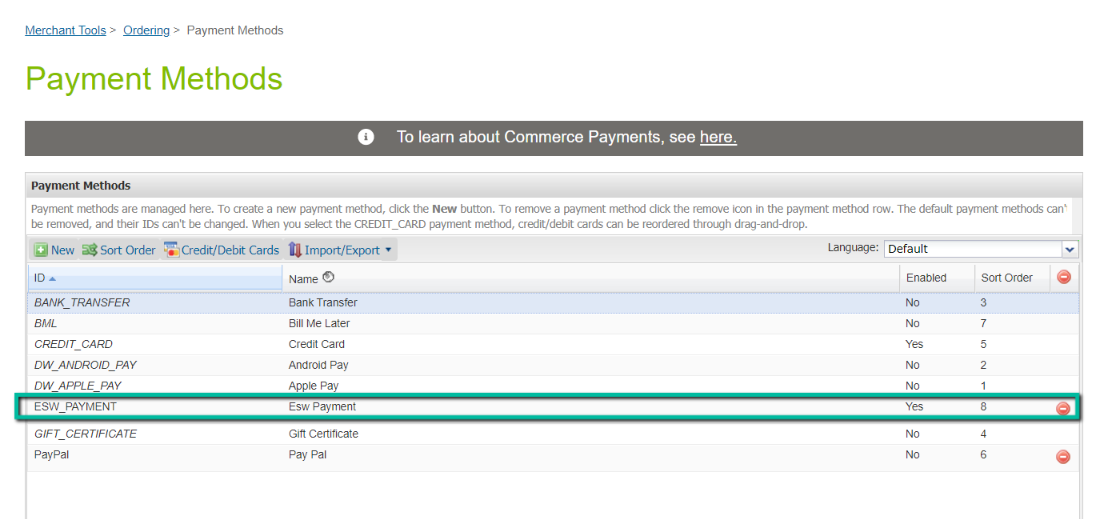


* + 1. Import **content-assets.xml** file in in MERGE mode:

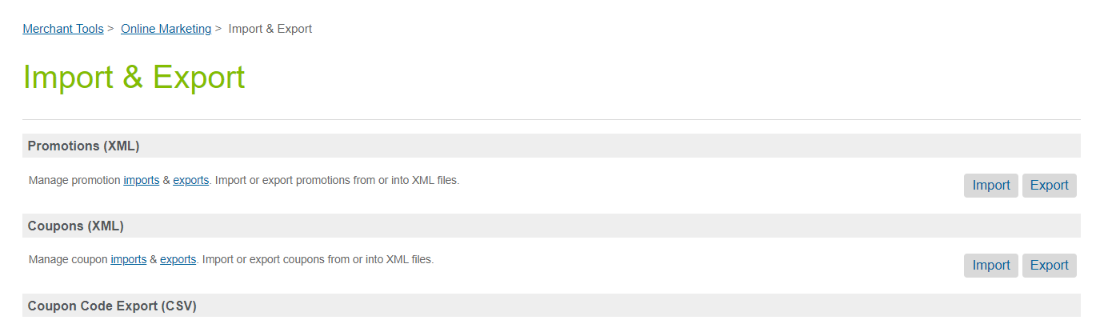


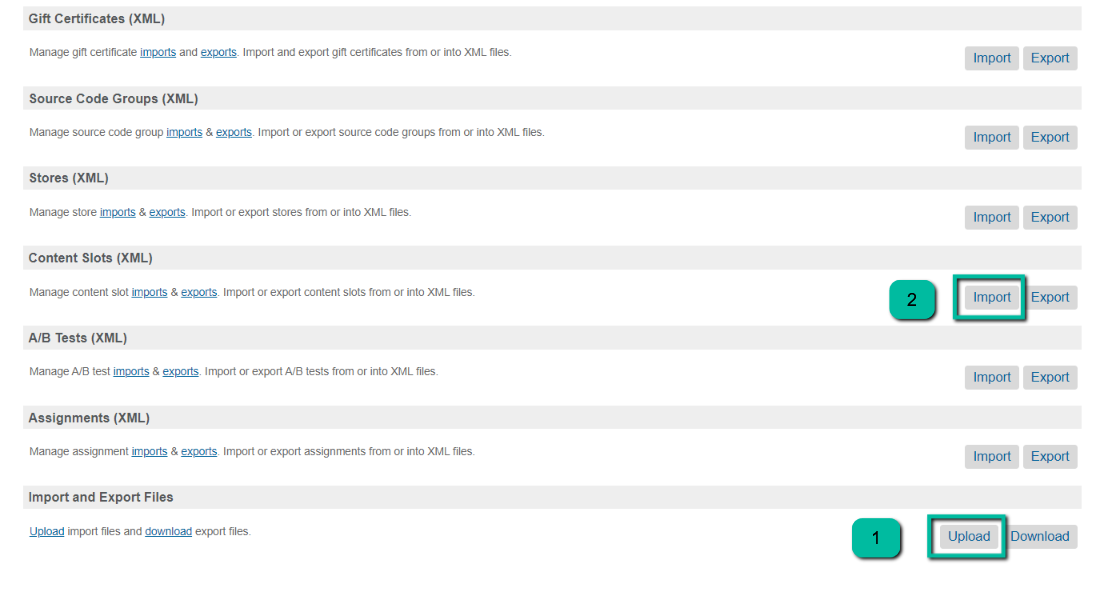
* + 1. Import **payment-methods.xml** file in in MERGE mode:





* + 1. Import **slots.xml** file in in MERGE mode:





* + 1. Enable ESW Retailer Pricing feed job. ESW recommends this job to be scheduled to run every day around midnight.
    2. Run ESW Retailer Auto configurator job. This job will configure ESW supported countries and currencies. This job is dependent upon the pricing feed job mentioned in step.6 so need to schedule accordingly. ESW recommends to schedule this job bi-weekly/monthly or to be run manually after country expansion. Retailer still need to add the following manually:
       - Non supported countries.
       - Price model, otherwise default price model will be used. (default price model is dynamic/calculated)
       - Base currency code, otherwise default base currency code will be used.

Graphical user interface, application, Teams

Description automatically generated

Cartridge Setup

SiteGenesis

To install the cartridge for the first time, perform the following steps:

1. Clone the repository to a local machine.
2. Import the two cartridges **int\_eshopworld\_controllers** and **int\_eshopworld\_core** into your workspace and link them to the server connection.
3. Import the site data by navigating to Metadata > Metadata and create a zip file for the site data folder, for example, Metadata.zip. After creating the file, import the Metadata.zip file into your instance.
4. Assign the **int\_eshopworld\_controllers** and **int\_eshopworld\_core** cartridges to the site’s cartridge path. The cartridge path sequence should look like this: **int\_eshopworld\_controllers:int\_eshopworld\_core:app\_storefront\_base:**
5. Assign the **int\_eshopworld\_controllers** and **int\_eshopworld\_core cartridges** to the business manager organization, for example, **int\_eshopworld\_controllers:int\_eshopworld\_core:bm\_custom\_plugin**.
6. To customize the code, refer to “**eShopWorld\_Cartridge\_Code\_Customization\_Guide\_SG.docx**” from documentation folder.

Storefront Reference Architecture

To install the cartridge for the first time, perform the following steps:

1. Clone the repository to a local machine.
2. Import the two cartridges **int\_eshopworld\_sfra** and **int\_eshopworld\_core** to your workspace and link them to the server connection.
3. Import the site data by navigating to Metadata > Sitesdata and create a zip file for the site data folder, for example, Metadata.zip. After creating the file, import the Metadata.zip file into your instance.
4. Assign the **int\_eshopworld\_sfra** and **int\_eshopworld\_core cartridges** to the site’s cartridge path. The cartridge path sequence should look like this:  
   **int\_eshopworld\_sfra:int\_eshopworld\_core:app\_storefront\_base:**
5. Assign the **int\_eshopworld\_sfra** and **int\_eshopworld\_core** cartridges to the Business Manager organization, for example, **int\_eshopworld\_sfra:int\_eshopworld\_core:bm\_custom\_plugin**.

Note: **int\_eshopworld\_sfra** is an overlay cartridge that overrides the retailer cartridge. All customizations are included in this cartridge; therefore, you do not need to make any customizations.

1. To customize the code, refer to “**eShopWorld\_Cartridge\_Code\_Customization\_Guide\_SFRA.docx**” from documentation folder.

Headless (OCAPI)

To install the cartridge for the first time:

1. Clone the repository to a local machine.
2. Import the two cartridges **int\_eshopworld\_headless** and **int\_eshopworld\_core** to your workspace and link them to the server connection.
3. Import the site data by navigating to **Metadata** > **Sitesdata** and create a zip file for the site data folder, for example, sitesdata.zip. After creating the file, import the sitesdata.zip file into your instance.
4. Assign the **int\_eshopworld\_headless** and **int\_eshopworld\_core** cartridges to the site’s cartridge path. The cartridge path sequence should look like this: **int\_eshopworld\_headless:int\_eshopworld\_core:app\_someother\_cartridge**
5. To add or update the OCAPI configuration, refer to “**eShopWorld\_Cartridge\_Configuration\_Guide\_Headless (OCAPI).docx**” from documentation folder.

SFCC Reference Information

ESW APIs in the Cartridge

Authentication API

An Authentication API call is required before a call is made to any of the ESW APIs, for example, Pricing Advisor API, Checkout API, or Package API.

You can get an authentication token for your API calls using the OAuth approach. Click [here](https://devportal.eshopworld.com/reference/api-security)to view the authentication requirements.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Service Name** | **Description** | **Interface** | **SLA** | **Notes** |
| esw.auth | HTTPS Interface in JSON format | REST JSON | Timeout: 30 sec  Circuit Breaker: 20 calls  Circuit Breaker Interval: 50 sec | The client ID and client secret must be a part of the service configuration. |

Basic Authentication during Order Confirmation

The cartridge uses basic authentication when it notifies SFCC about the order confirmation.

For Sandbox instances, you must pass the  OrderConfirmationUri\_TestOnly and  OrderConfirmationBase64EncodedAuth\_TestOnly parameters. These parameters are available undermetadataItems in the retailerCheckoutExperience object.

* The OrderConfirmationUri\_TestOnly parameter holds the URL for the Notify request.
* If basic authentication is enabled, the OrderConfirmationBase64EncodedAuth\_TestOnly parameter holds the encoded base64 string of the username and password.
* If basic authentication is not enabled, the OrderConfirmationBase64EncodedAuth\_TestOnly parameter contains ‘1’ as the fixed value.

For example:

{

"retailerCheckoutExperience": {

"backToCartUrl":"https://dev1.demandware.net/on/demandware.store/Sites-RefArch-Site/en\_US/Cart-Show",

"continueShoppingUrl":"https://dev1.demandware.net/on/demandware.store/Sites-RefArch-Site/en\_US/Home-Show",

"metadataItems": [

{

"Name":"OrderConfirmationUri\_TestOnly",

"Value":"https://dev1.demandware.net/on/demandware.store/Sites-RefArch-Site/en\_US/EShopWorld-Notify",

},

{

"Name":"OrderConfirmationBase64EncodedAuth\_TestOnly",

"Value":"gasxdasFSAFASGAfasfas==",

}

]

}

For the Production instance, you must save the username and password details provided by eShopWorld under the [ESW General Configuration](#ESWGeneralConfiguration) custom preference group.

Pricing Advisor

The cartridge uses the data from Pricing Advisor to calculate prices. The following formula is used to calculate product prices as below:

**For Pav3:** (((Product base price x (1 + priceUpliftPercentage)) x (1 + dutyPercentage)) x (1 + taxPercentage)) x fxRate = Price in the shopper currency → apply rounding = Final product price.

**For Pav4:** (((Product base price \* (1 + priceUpliftPercentage/100)) \* (1 + dutyPercentage/100)) \* (1 + taxPercentage/100) \* (1 + feePercentage/100)) \* fx = Price in the shopper currency → apply rounding = Final product price.

If the country is calculated price enabled, the following price calculation logic applies:

* If the shopper pays in the base currency, only the country adjustments are applied to the price from the price book.
* If the shopper pays in a currency other than the base currency, the FX rate and country adjustment are applied to the price from the price book.

If the country is fixed price enabled, the following price calculation logic applies:

* If the shopper pays in the base currency, no calculation is done, and the price is taken from the price book.

For information on how the product prices are calculated in the shopper currency in different scenarios based on the currency exponent, rounding direction, and model, click [here](https://know.eshopworld.com/space/DT/256835679/Pricing+Advisor+API+v3.0+Sample+Integration). For the

* Pricing Advisor API v3.0, click [here](https://know.eshopworld.com/space/DT/255590684/Pricing+Advisor+API+v3.0).
* Pricing Advisor API v4.0, click [here](https://know.esw.com/space/TDR/1234633252/Pricing+Advisor+API+v4.0)

Pricing Advisor API

For the Pricing Advisor API request, an access token must be sent in every Authorizationbearer header attribute.

Note that an Authentication API call is required before a call is made to the Pricing Advisor API.

* grant\_type = "client\_credentials"
* scope = pricing.advisor.api.all

How does it work?

When the pricing feed job is running from SFCC BM, a backend call is made to the Authentication API. The API call includes the access token in the request. ESW validates the request as well as the token and sends the pricing advisor details like FX rates, country adjustment and rounding rules in the response, which is then stored in SFCC custom site preferences.

|  |  |  |  |
| --- | --- | --- | --- |
| **Service Name** | **Description** | **Interface** | **SLA** |
| EswPriceFeedV3Service | HTTPS Interface in JSON format | REST JSON | Timeout: 30 sec  Circuit Breaker: 20 calls  Circuit Breaker Interval: 50 sec |

Pricing Advisor API Request/Response Schema

**API Call**: REST HTTPS

**Swagger URL**: <https://pricing-advisor-api.sandbox.eshopworld.com/swagger/index.html?urls.primaryName=Pricing.Advisor.Api%20PricingAdvisor_v3>

**Request**: No request body parameters

**Response**

{

"fxRates": [

{

"fromRetailerCurrencyIso": "EUR",

"toShopperCurrencyIso": "USD",

"rate": "1.05"

}

],

"deliveryCountryAdjustments": [

{

"deliveryCountryIso": "GB",

"retailerAdjustments": {

"priceUpliftPercentage": 12.5

},

"estimatedRates": {

"dutyPercentage": 18,

"taxPercentage": 22

}

}

],

"deliveryCountryCurrencyDisplays": [

{

"deliveryCountryIso": "GB",

"currencyDisplays": [

{

"currencyIso": "USD",

"currencyExponent": 2,

"currencySymbol": "€",

"showTrailingZeros": true,

"thousandSeparator": ",",

"decimalSeparator": ".",

"configurationString": "[Number][ExponentSeparator][Exponent] [CurrencyISO]"

}

]

}

],

"deliveryCountryRoundingModels": [

{

"deliveryCountryIso": "GB",

"roundingModels": [

{

"currencyIso": "EUR",

"currencyExponent": 2,

"direction": "Up",

"model": "none.fixed50"

}

]

}

],

"lastUpdated": "1994-11-05T13:15:30Z",

"pricingSynchronizationId": "f5ac11b9-cbc6-4ba2-932f-7bf207b3ed61"

}

ESW Checkout

The ESW Checkout page is displayed when shoppers click **Checkout** on the retailer’s site. The following features are available:

* URLs: The **Continue Shopping** and **Back to Cart** URLs are provided on the ESW Checkout page. When shoppers click this link, they are redirected to the SFCC storefront.
* Prices: The ESW Checkout page displays the product unit prices and shipping cost.

Checkout API

For all Checkout API requests, an access token must be sent in everyAuthorization bearer header attribute.

Note that an Authentication API call is required before a call is made to the Checkout API.

* grant\_type = "client\_credentials"
* scope = checkout.preorder.api.all

****How does it work?****

When a shopper clicks **Checkout** on the retailer’s site, a backend call is made to the Authentication API. The API call includes the access token in the request. ESW validates the request as well as the token and sends the console URL in the response. SFCC then redirects the shopper to the eShopWorld Checkout page.

|  |  |  |  |
| --- | --- | --- | --- |
| **Service Name** | **Description** | **Interface** | **SLA** |
| EswCheckoutV3Service | HTTPS Interface in JSON format | REST JSON | Timeout: 30 sec  Circuit Breaker: 20 calls  Circuit Breaker Interval: 50 sec |

Sending information

* **Order Number**: The orders are in the **Created** state before the checkout call is made. The generated order number must be sent in retailerCartId attribute in the request.
* **Shipping and Billing Address**: For guest shoppers, dummy values are sent to create shipping and billing addresses. For registered shoppers, the default saved address is sent in shipping and billing address attributes.

When a shopper returns to the Cart page from the ESW Checkout page, the order fails, and the cart opens for modifications. When the shopper clicks **Checkout**on the Cart page, the order is created again, and a new order number is generated.

Checkout API Request/Response Schema

**API Call**: REST HTTPS

**Swagger URL**: [https://checkout-api.sandbox.eshopworld.com/swagger/](https://checkout-api.sandbox.eshopworld.com/swagger/index.html?urls.primaryName=Checkout.Api%20PreOrder_v3)

**API Request**

{

"retailerCartId": "XX123456",

"contactDetails": [

{

"contactDetailsType": "IsDelivery",

"contactDetailsNickName": "NickName",

"addressId": "AddressId",

"address1": "1234 Line Ave S",

"address2": "Apt B1",

"address3": "New York state",

"city": "New York city",

"postalCode": "10001",

"region": "Region",

"country": "US",

"email": "email@test.com",

"firstName": "First Name",

"lastName": "Last name",

"telephone": "+14211112222",

"gender": "None",

"poBox": "",

"metadataItems": [

{

"name": "KeyName",

"value": "KeyValue"

}

]

}

],

"retailerPromoCodes": [

{

"promoCode": "XX123456",

"title": "Season discount",

"description": "Discounts for all goods due to winter season"

}

],

"lineItems": [

{

"quantity": 2,

"estimatedDeliveryDateFromRetailer": "2020-08-31T11:54:24.6310000+00:00",

"lineItemId": "2",

"product": {

"productCode": "123456",

"title": "Black dress",

"description": "Long black dress",

"imageUrl": "https://url.com/test.png",

"color": "black",

"size": "38",

"productUnitPriceInfo": {

"price": {

"currency": "AUD",

"amount": "100.00"

},

"discounts": [

{

"title": "Season discount",

"description": "Winter season discount",

"discount": {

"currency": "AUD",

"amount": "100.00"

},

"beforeDiscount": {

"currency": "AUD",

"amount": "100.00"

}

}

]

},

"metadataItems": [

{

"name": "KeyName",

"value": "KeyValue"

}

],

"isReturnProhibited": false,

"inStock": true

},

"cartGrouping": "Group1",

"metadataItems": [

{

"name": "KeyName",

"value": "KeyValue"

}

]

}

],

"cartDiscountPriceInfo": {

"price": {

"currency": "AUD",

"amount": "100.00"

},

"discounts": [

{

"title": "Season discount",

"description": "Winter season discount",

"discount": {

"currency": "AUD",

"amount": "100.00"

},

"beforeDiscount": {

"currency": "AUD",

"amount": "100.00"

}

}

]

},

"retailerCheckoutExperience": {

"backToCartUrl": "https://url.com/cart",

"continueShoppingUrl": "https://url.com",

"metadataItems": [

{

"name": "KeyName",

"value": "KeyValue"

}

]

},

"shopperCurrencyIso": "EUR",

"retailerCurrencyIso": "EUR",

"pricingSynchronizationId": "00000000-0000-0000-0000-000000000000",

"deliveryCountryIso": "AU",

"shopperCheckoutExperience": {

"useDeliveryContactDetailsForPaymentContactDetails": true,

"emailMarketingOptIn": true,

"registeredProfileId": "123456",

"shopperCultureLanguageIso": "en-IE",

"expressPaymentMethod": "PayPal",

"viewType": "string",

"sessionTimeout": 20,

"metadataItems": [

{

"name": "KeyName",

"value": "KeyValue"

}

]

},

"deliveryOptions": [

{

"deliveryOption": "Exp2",

"deliveryOptionOverridePriceInfo": {

"price": {

"currency": "AUD",

"amount": "100.00"

},

"discounts": [

{

"title": "Season discount",

"description": "Winter season discount",

"discount": {

"currency": "AUD",

"amount": "100.00"

},

"beforeDiscount": {

"currency": "AUD",

"amount": "100.00"

}

}

]

},

"metadataItems": [

{

"name": "KeyName",

"value": "KeyValue"

}

]

}

],

"retailerDeliveryOptions": [

{

"title": "Non-EshopWorld delivery option",

"description": "Fastest delivery method",

"estimatedDeliveryDateToShopper": "2020-08-31T11:54:24.6310000+00:00",

"retailerDeliveryOptionPriceInfo": {

"price": {

"currency": "AUD",

"amount": "100.00"

},

"discounts": [

{

"title": "Season discount",

"description": "Winter season discount",

"discount": {

"currency": "AUD",

"amount": "100.00"

},

"beforeDiscount": {

"currency": "AUD",

"amount": "100.00"

}

}

]

},

"metadataItems": [

{

"name": "KeyName",

"value": "KeyValue"

}

]

}

]

}

**API Response**

{

"preOrderCode": "00000000-0000-0000-0000-000000000000",

"orderNumber": "00000000-0000-0000-0000-000000000000",

"redirectUrl": "https://checkout\_url/00000000-0000-0000-0000-000000000000"

}

Advanced Shipping Notification (ASN)

The [eswRetailerPackageFeed](#eswRetailerPackageFeed)job synchronizes the package data and syncs the package information to the ESW package service. When the SFCC job runs, ESW Package API v4.0 is used to send the ASN to ESW.

Package API

For the Package API request, an access token must be sent in every Authorizationbearer header attribute.

Note that similar to the other APIs described on this page, an Authentication API call is required before a call is made to the ESW Package API.

* grant\_type = "client\_credentials"
* scope = logistics.package.api.all

How does it work?

When the [eswRetailerPackageFeed](#eswRetailerPackageFeed)job is running from the SFCC Business Manager, a backend call is made to the Authentication API. The API call includes the access token in the request. eShopWorld validates the request and the token and sends the ESW Package Reference in the response. The ESW Package Reference is then stored in the SFCC order custom attribute.

|  |  |  |  |
| --- | --- | --- | --- |
| **Service Name** | **Description** | **Interface** | **SLA** |
| EswPackageV4Service | HTTPS Interface in JSON format | REST JSON | Timeout: 30 sec  Circuit Breaker: 20 calls  Circuit Breaker Interval: 50 sec |

Package API Request/Response Schema

**API Call**: REST HTTPS

**Swagger URL**: <https://package-api.sandbox.eshopworld.com/swagger/index.html?urls.primaryName=V4>

**API Request**

{

"brandCode": "GOC",

"orderReference": "ABC1234567890",

"packageReference": "AAA1234567890",

"orderType": "CHECKOUT",

"weight": {

"weight": 12.34,

"weightUnit": "KG"

},

"shippingDocumentationRequested": false,

"returnDocumentationRequested": false,

"shippingStatus": "Complete",

"parentOrderReference": "ABC1234567890",

"carrierId": 123,

"carrierReference": "ABC1234567890",

"distributionCentre": "USDC1",

"hubCode": "AMS",

"additionalImportInformation": "Additional Import Information",

"isBackOrder": false,

"dangerousGoods": false,

"consignee": {

"firstName": "First Name",

"lastName": "Last Name",

"address1": "Address 1",

"address2": "Address 2",

"address3": "Address 3",

"city": "City",

"postalCode": "Post Code",

"poBox": "PO Box 123",

"region": "US",

"country": "US",

"gender": "Female",

"email": "test@test.com",

"telephone": "0123456789",

"unit": "Unit 456"

},

"shippingInfo": {

"amount": 123.45,

"currency": "USD"

},

"dimensions": {

"dimHeight": "12.34",

"dimLength": "12.34",

"dimWidth": "12.34",

"dimWeight": "12.34",

"dimMeasurementUnit": "INCH"

},

"goodsDescription": "Goods Description",

"serviceLevel": "POST",

"packageItems": [

{

"productCode": "ABC1234567890",

"lineItemId": 1,

"quantity": 1,

"productDescription": "Product Description",

"productCustomsDescription": "US",

"countryOfOrigin": "US",

"weight": {

"weight": 12.34,

"weightUnit": "KG"

},

"unitPrice": {

"amount": 123.45,

"currency": "USD"

},

"hsCode": "ABC123456",

"fta": false,

"dangerousGoods": false,

"serialNumber": "ABC123456",

"warrantyId": "ABC123456"

}

],

"additionalCarrierData": {

"additionalCarrierData1": "Additional Carrier Data 1",

"additionalCarrierData2": "Additional Carrier Data 2",

"additionalCarrierData3": "Additional Carrier Data 3",

"additionalCarrierData4": "Additional Carrier Data 4",

"additionalCarrierData5": "Additional Carrier Data 5"

},

"palletId": "ABC123456",

"metadata": {}

}

**API Response**

{

"outcome": 1,

"statusMessage": "Success",

"package": {

"brandCode": "GOC",

"orderReference": "ABC123456789",

"packageReference": "AAA123456789",

"eShopPackageReference": 0,

"parentOrderReference": "AAA123456789",

"orderType": "CHECKOUT",

"creationTime": "2020-06-29T12:57:23.0000000",

"carrierId": 123456,

"carrierName": "DHL Express",

"carrierReference": "DHL1234567",

"weight": {

"weight": 12.34,

"weightUnit": "KG"

},

"shippingStatus": "Complete",

"shippingInfo": {

"amount": 123.45,

"currency": "USD"

},

"goodsDescription": "Test Description",

"serviceLevel": "POST",

"dangerousGoods": false,

"dimensions": {

"dimHeight": "12.34",

"dimLength": "12.34",

"dimWidth": "12.34",

"dimWeight": "12.34",

"dimMeasurementUnit": "INCH"

},

"holdReleaseStatus": "Release",

"palletId": "ABC1234567",

"distributionCentre": "USDC1",

"hubCode": "AMS",

"trackingUrl": "http://<example>/trackingId?1234567</example>",

"additionalCarrierData": {

"additionalCarrierData1": "Additional Carrier Data 1",

"additionalCarrierData2": "Additional Carrier Data 2",

"additionalCarrierData3": "Additional Carrier Data 3",

"additionalCarrierData4": "Additional Carrier Data 4",

"additionalCarrierData5": "Additional Carrier Data 5"

},

"additionalImportInformation": "Additional Import Information",

"consignee": {

"firstName": "First Name",

"lastName": "Last Name",

"address1": "Address 1",

"address2": "Address 2",

"address3": "Address 3",

"city": "City",

"postalCode": "Post Code",

"poBox": "PO Box 123",

"region": "US",

"country": "US",

"gender": "Female",

"email": "test@test.com",

"telephone": "0123456789",

"unit": "Unit 456"

},

"packageItems": [

{

"productCode": "ABC1234567890",

"lineItemId": 1,

"quantity": 1,

"productDescription": "Product Description",

"productCustomsDescription": "US",

"countryOfOrigin": "US",

"weight": {

"weight": 12.34,

"weightUnit": "KG"

},

"unitPrice": {

"amount": 123.45,

"currency": "USD"

},

"hsCode": "ABC123456",

"fta": false,

"dangerousGoods": false,

"serialNumber": "ABC123456",

"warrantyId": "ABC123456"

}

],

"shippingDocumentation": [

{

"Document": "JVBERi0xLjMNCjEgMCBvYmoNCjw8DQovVHlwM...",

"type": "SHIPPING\_LABEL",

"format": "Pdf",

"link": "http://examplelink",

"documentStatus": "Created"

}

],

"returnsDocumentation": [

{

"returnTrackingNumber": "AA12345678",

"returnShipmentNo": "12345678-1234-1234-1234-123456789012",

"returnTrackingUrl": "http://tracking-test.someurl.com?SendingId=1123123123123123..",

"Document": "JVBERi0xLjMNCjEgMCBvYmoNCjw8DQovVHlwM...",

"type": "SHIPPING\_LABEL",

"format": "Pdf",

"link": "http://examplelink",

"documentStatus": "Created"

}

],

"metadata": {}

}

}

New Countries or Markets Expansion

The ESW-SFCC cartridge allows you to manage shipping countries from within the cartridge, including adding new shipping countries and displaying them at the storefront.

For the newly added countries, you can also configure shopper currencies. This means when a shopper selects the newly added country from the **Country**selector, the shopper currency that you have configured for that country is displayed by default.

Enabling a new country at the storefront is a multi-step process that involves the following:

1. New ESW supported country needs to be configured on the ESW tenant side. This means ESW integration will configure the supported countries in the ESW Database for the tenant.
2. Running the **Retailer Pricing Feed** job. This will fetch the FX rates and country adjustments for the newly added country into SFCC custom objects.
3. Running the **ESW Retailer Auto Configurator** Job. This will create custom objects for the newly added country (ESW Countries) and its currency (ESW Currencies) if its custom object instance does not exists already.
4. Optionally, Configuring the base currency (if base currency is different from site’s default currency) and pricing model manually. By default the pricing model is set to dynamic/calculated. Pricing model for a country can be changed by toggling the ‘is Fixed Price Model’ Flag in the newly added country’s custom object. For more information visit: **eShopWorld\_Pricing\_Integration.docx**

Graphical user interface, text, application, email

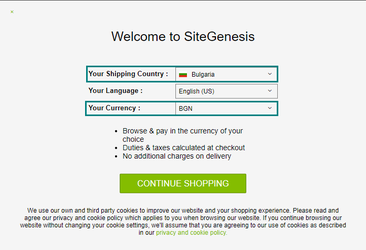
Description automatically generated

1. Optionally, verifying the country and currency information on the storefront and other webpages. The newly added country will also be available on welcome matt.

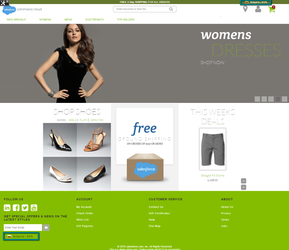
**Note:** Any non-supported country will have to be added manually by the retailer.

### **Verifying the new country-currency selection**

After the new country is enabled, open the storefront and select the newly added country from the dropdown. The default currency mapped to this country is selected automatically.



You can also verify the country and currency information on the ESW storefront components (Header, Footer, Country selector widgets) and check prices on the Product Display Page (PDP), Product Listing Page (PLP), and Cart Page.



* Depending on the pricing model for the newly added countries, ensure that the other configuration like override pricebook, override shipping method, ESW FX rates, BM currencies, and so on are correct as per the business use case. For information on how to configure these, see[ESW Pricing Configuration](#ESWPricingConfiguration).
* Ensure that the newly added country is also configured on the ESW Checkout tenant side so that shoppers are redirected successfully to the appropriate ESW Checkout page.

Headless (OCAPI) Attributes Mapping with Cart

This section describes how you can use the OCAPI response attributes to display or represent different prices on the Cart page.

ESW Headless (OCAPI) Attributes

The following table contains different pricing-related Cart (Basket) level and ProductLineItem level custom attributes from OCAPI and their description.

Product Line Item

|  |  |
| --- | --- |
| **Custom Attribute** | **Description** |
| eswUnitPrice | Item price before the discount is applied. |
| eswBasePrice | Item price after the discount is applied. This is the final or sale price. |
| eswPrice | Total item price multiplied by the quantity. |
| eswPriceAfterOrderDiscount | Total item price after the order discount. |

Cart (Basket)

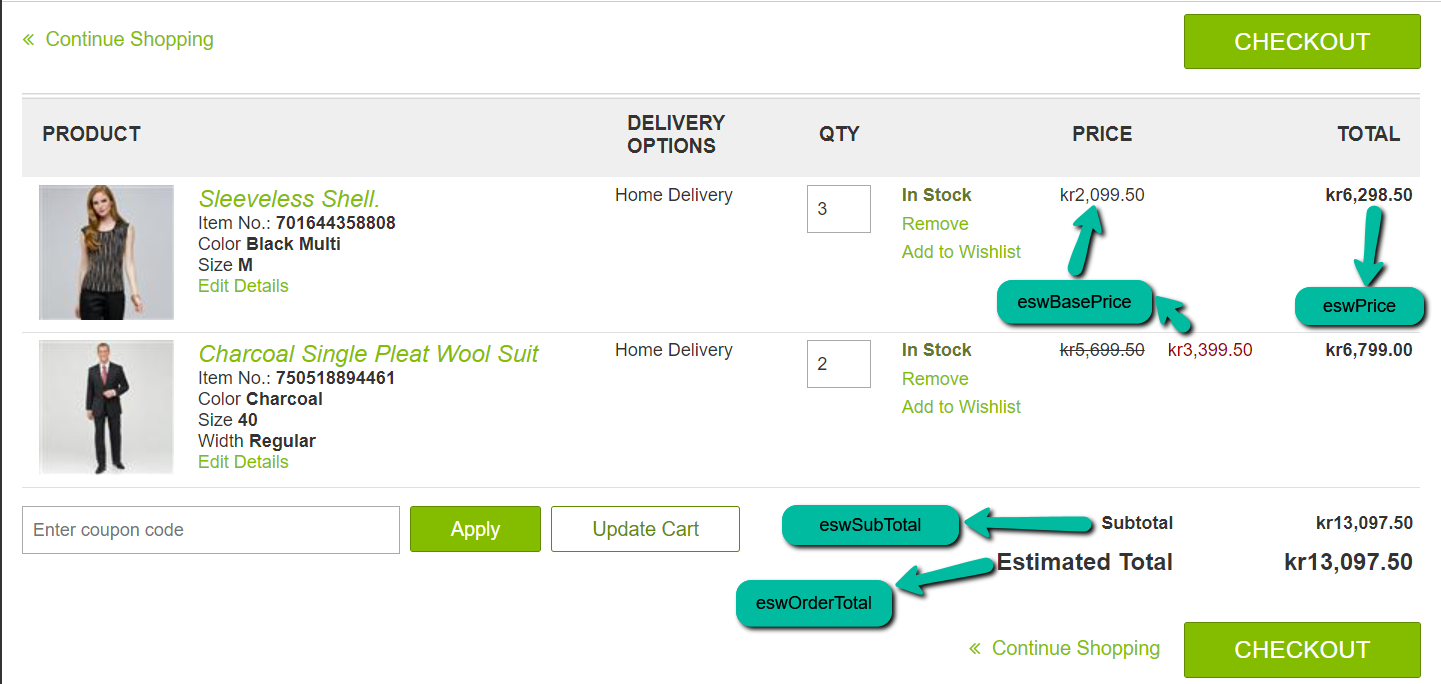
|  |  |
| --- | --- |
| **Custom Attribute** | **Description** |
| eswSubTotal | Cart subtotal or a total of all product line items. |
| eswOrderDiscount | Total order discount applicable to the cart.  Note that this custom attribute does not include a shipping discount. |
| eswOrderTotal | The estimated order total. |
| eswShopperCurrency | Three-letter ISO code of the shopper's currency. |

Mapping Examples

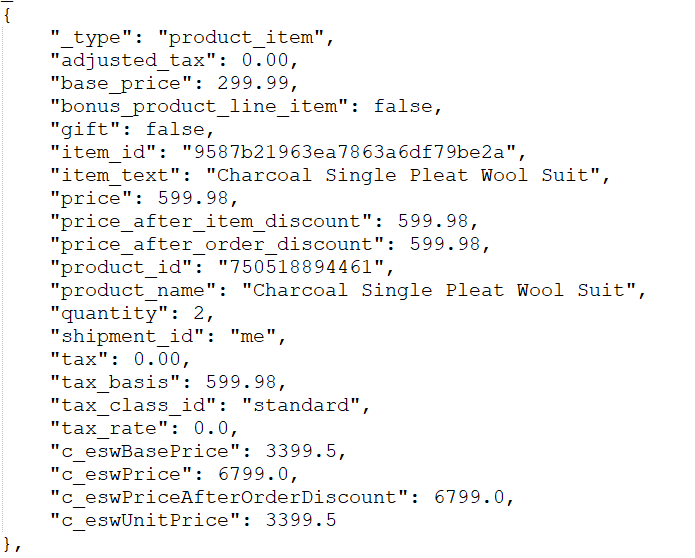
This section contains examples of how the attributes are mapped on the Storefront, Postman, and ESW Checkout.

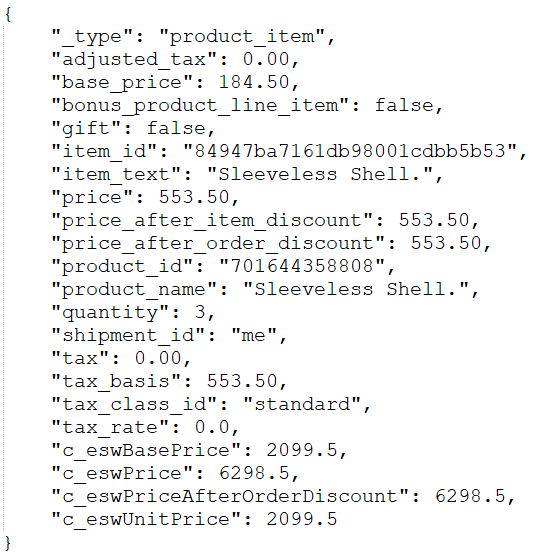
Cart (Basket) Page without Promotion

***SFCC Storefront (Cart Page)***

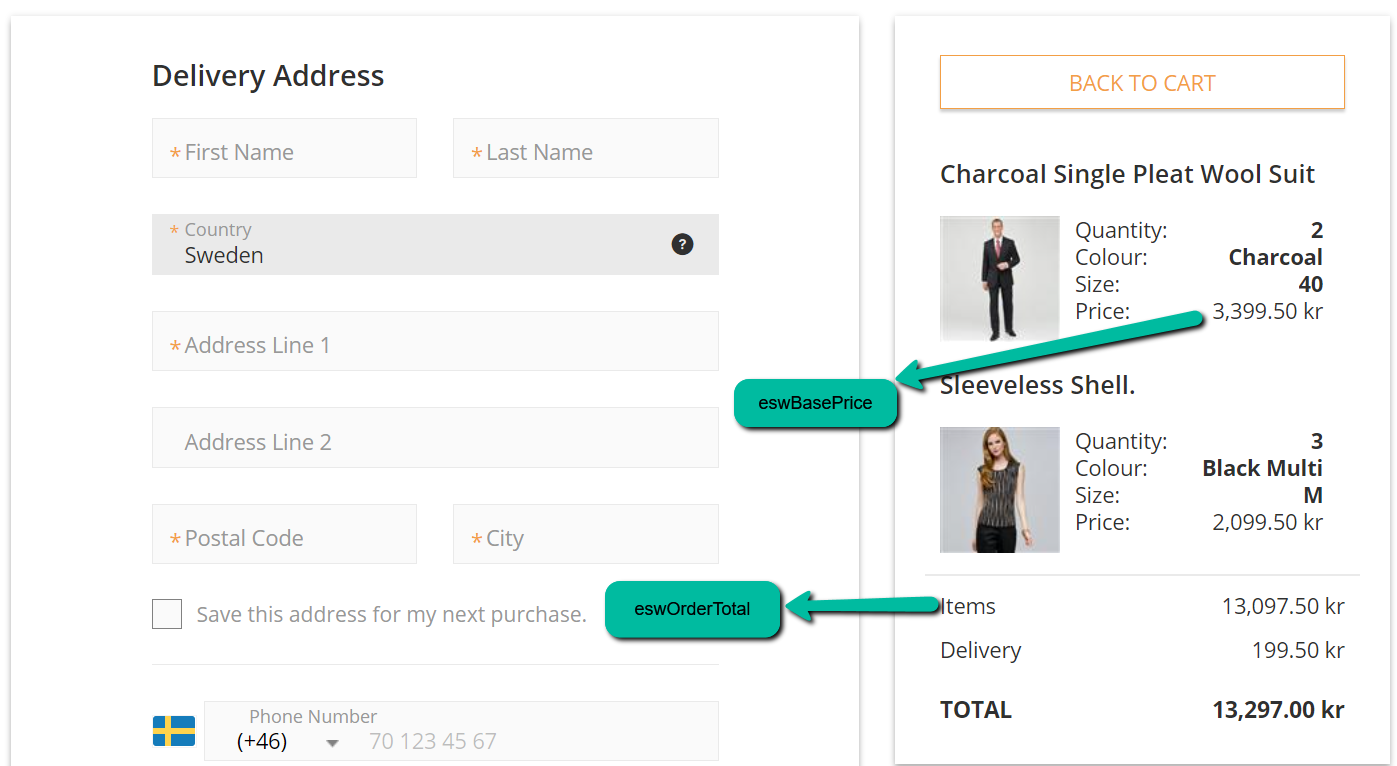


***Postman (OCAPI Response)***





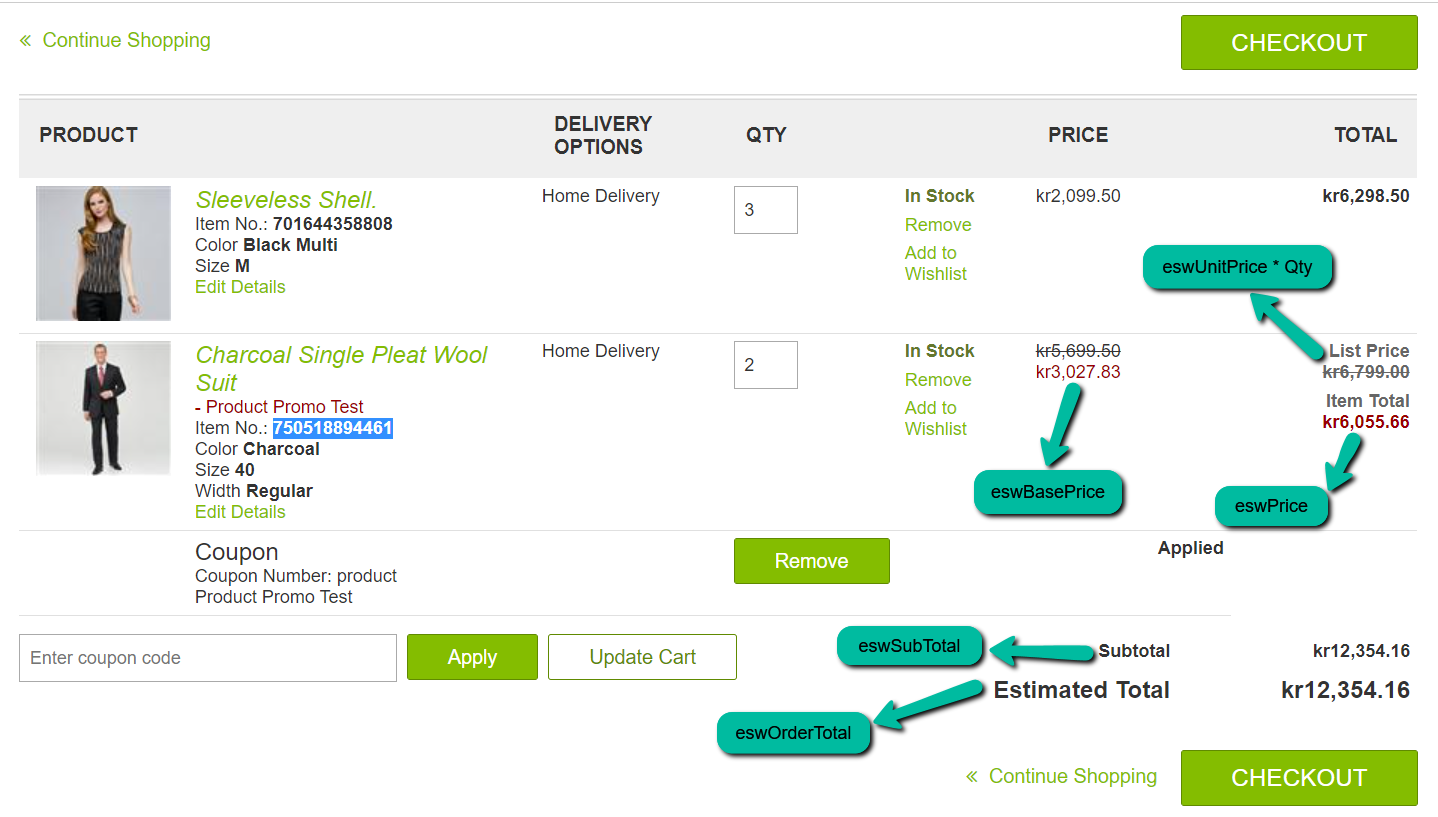
***ESW Checkout***



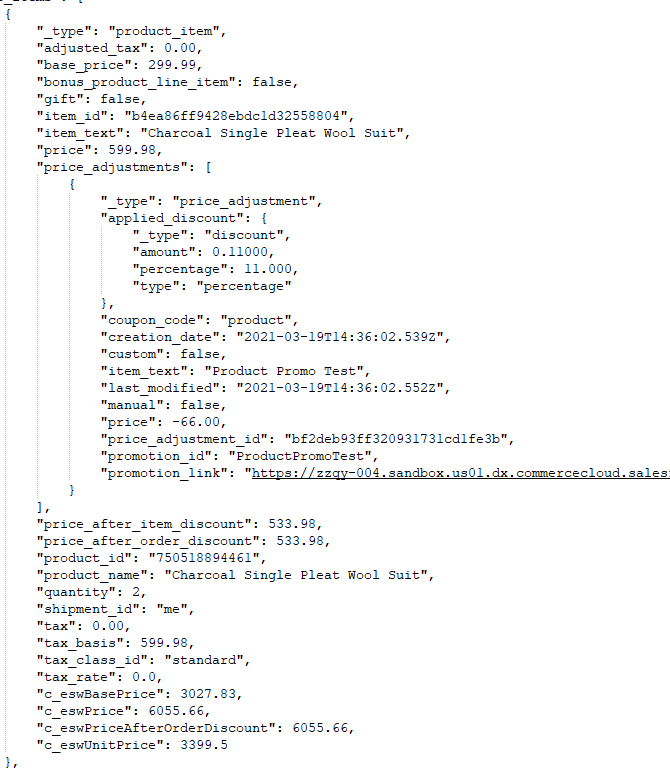
Cart (Basket) Page with Promotion

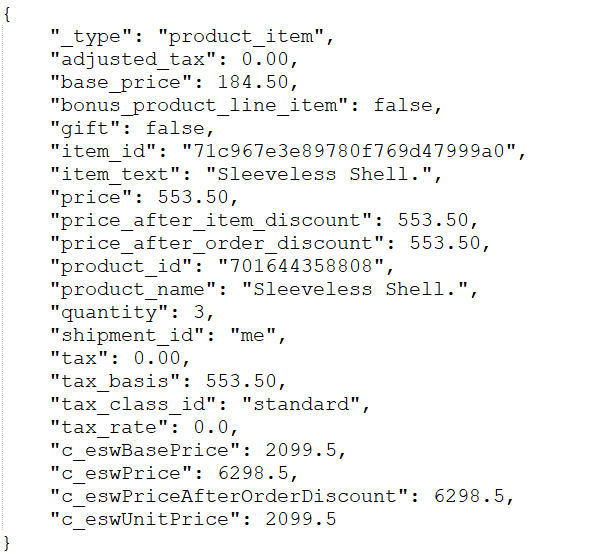
##### **Product Level Promotion (11% off)**

***SFCC Storefront (Cart Page)***

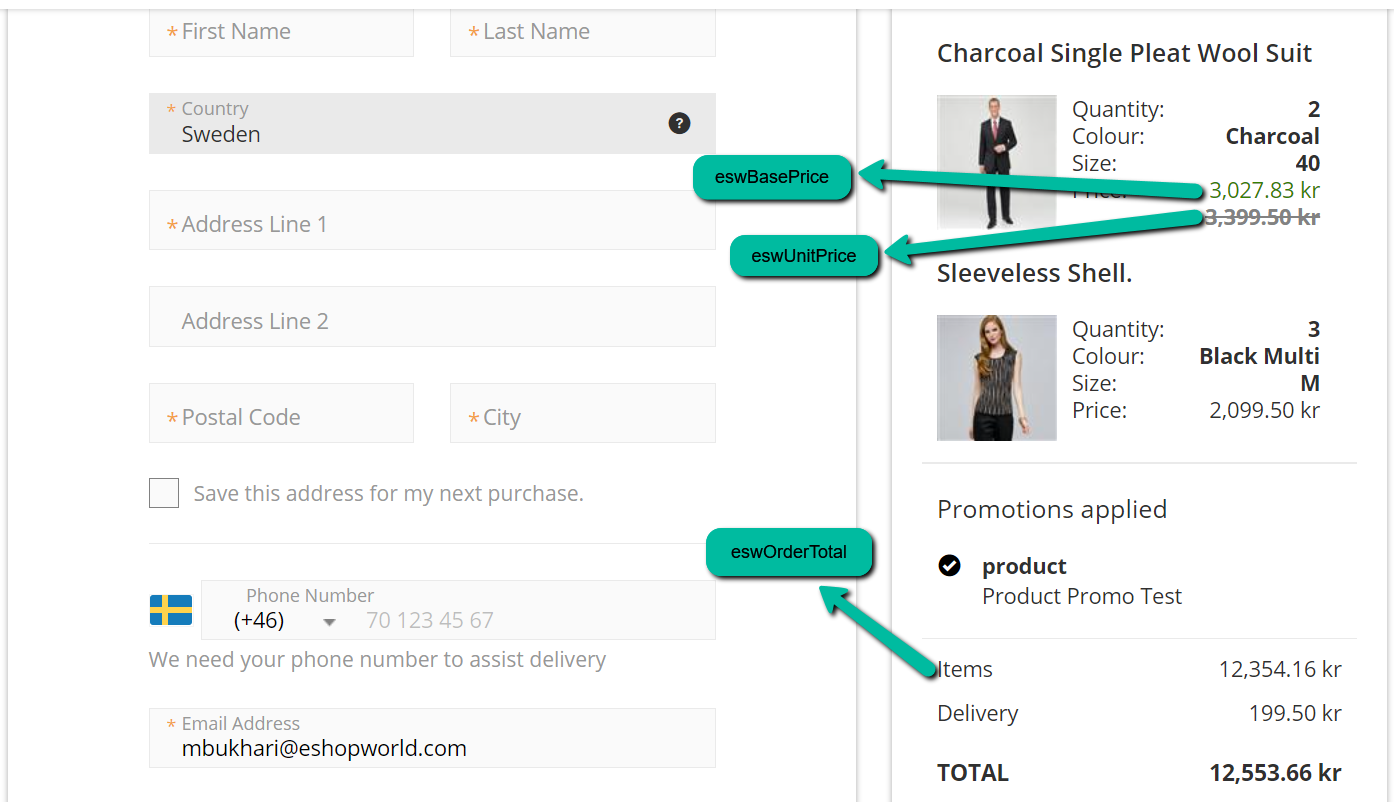


***Postman (OCAPI Response)***



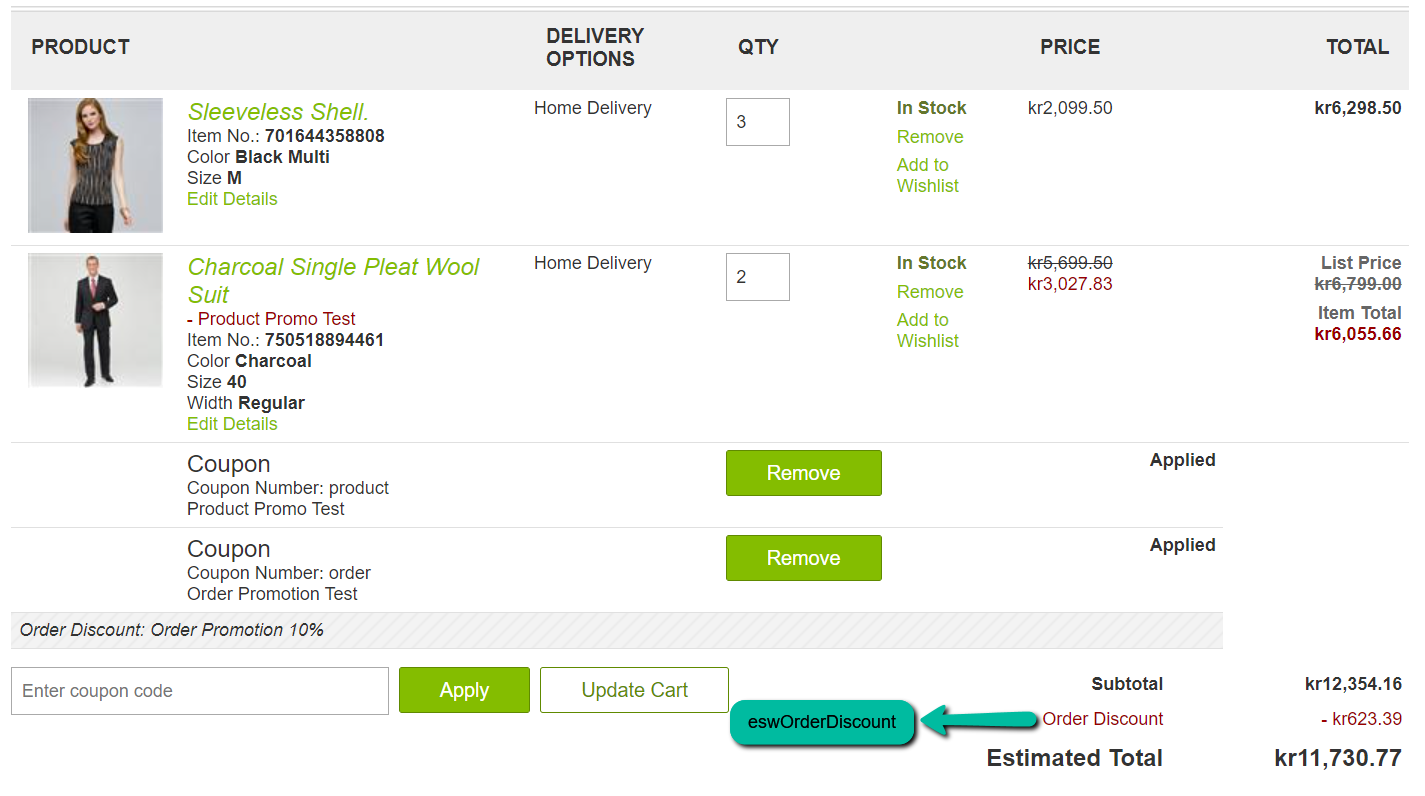


***ESW Checkout***

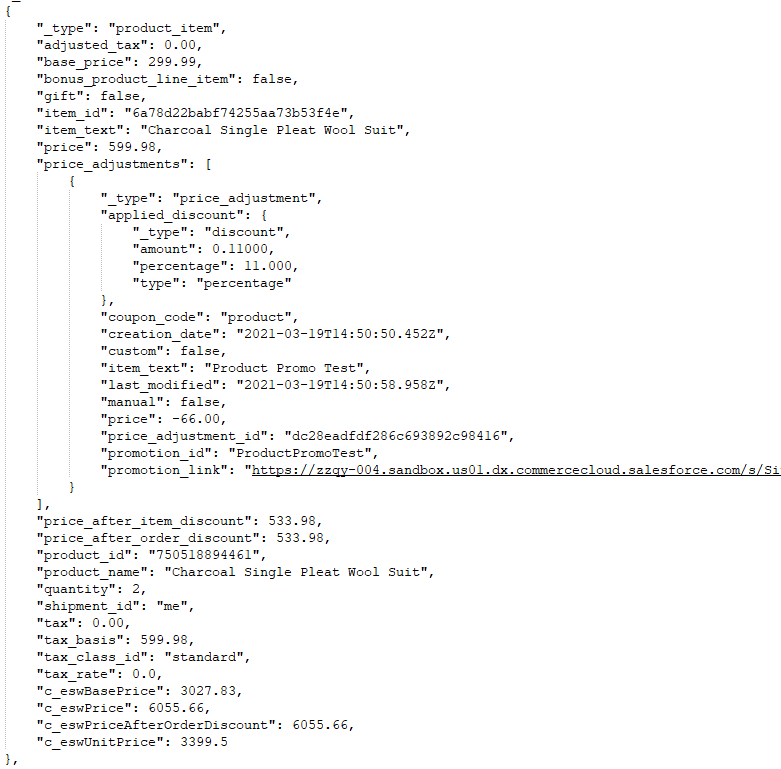


##### **Product & Order Promotion**

***SFCC Storefront (Cart Page)***

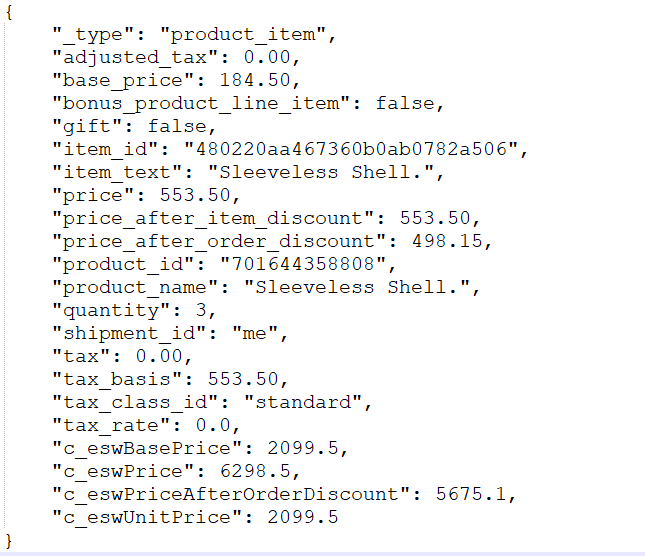


***Postman (OCAPI Response) – ProductLineItem & Basket level***

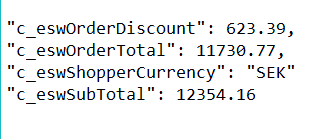


*Postman (OCAPI Response) – ProductLineItem & Basket level*

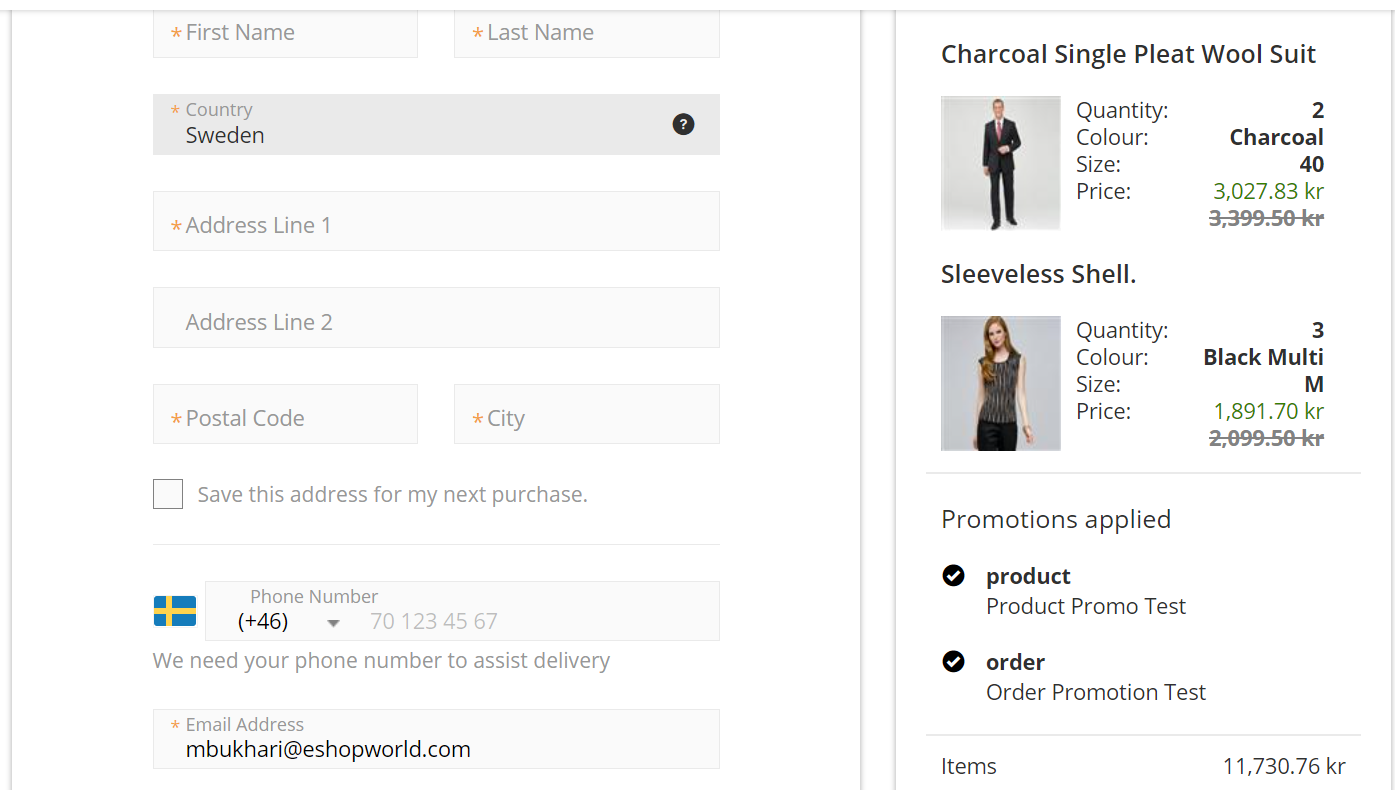
**Postman (OCAPI Response) – ProductLineItem & Basket level**



***Postman (OCAPI Response) – Order level***



***ESW Checkout***



On the ESW Checkout, the order discount amount is prorated across the 'qualified products'. In the last example (Cart (Basket) Page with Promotion), the product with the name 'Sleeveless Shell' is a 'qualified product. Therefore, the prorated price is displayed on the ESW checkout for this product. The item’s price after order discount is stored in the eswPriceAfterOrderDiscount ProductLineItem level custom attribute.

How to Upgrade?

The ESW-SFCC cartridge is updated at regular intervals to include new enhancements and integration support for ESW services. For the ESW-SFCC cartridge, two upgrade paths are available: Full Version Upgrade and Patches.

Full Version Upgrade

ESW releases a new version of the ESW-SFCC cartridge on a quarterly basis. You can upgrade the cartridge to the latest version by replacing the installed version with the latest version.

**IMPORTANT**: A full version upgrade must be carried out carefully as it directly impacts the configuration. If not performed correctly, the upgrade might override your customizations. For SiteGenesis, there is a higher risk because all template files are shared and there is no modularity, as is the case with SFRA.

To minimize any integration risk, ESW recommends that you refer to our latest [product release notes](https://know.esw.com/space/TDR/1234639152/SFCC+Release+Notes) and use the document as a base to conduct a gap analysis between the new version and the installed version. Remember to include code customizations in your analysis.

To perform a full version upgrade, perform the following steps:

1. Import the updated metadata of the site from **link\_eshopworld\Metadata**.
2. Update the codebase by performing one of the following steps depending on the SFCC architecture that you use:

* For SiteGenesis, import the **link\_eshopworld\cartridges\ int\_eshopworld\_controllers** cartridge from the **link\_eshopworld cartridges** directory.
  + For SFRA, import the **link\_eshopworld\cartridges\ int\_eshopworld\_sfra** cartridge from the **int\_eshopworld\_sfra** directory.
  + For Headless OCAPI, import the **link\_eshopworld\cartridges\ int\_eshopworld\_headless** cartridge from the cartridges provided.
  + For PWA, import the **link\_eshopworld\cartridges\ int\_eshopworld\_pwa** cartridge from the cartridges provided.

1. Ensure that the services and site preference configurations in the Business Manager are configured fully as per the documentation.
2. Review the site-level configuration, as provided in our [online documentation](https://know.esw.com/space/TDR/1234637137/Merchant+Tools).
3. Review the administration-level configuration, as provided in our [online documentation](https://know.esw.com/space/TDR/1234637280/Administration).

Recommended end-to-end approach

The following steps outline the end-to-end approach that ESW recommends for full version upgrades:

1. Set up a clean SFCC sandbox and install all third-party cartridges.
2. Remove any customization that you might have done to extend the functionality of the ESW-SFCC cartridge.
3. Install the new version of the ESW-SFCC cartridge using the steps outlined in the **Full Version Upgrade** (above) section.
4. Review the gap analysis to determine if any customization is still required. If yes, apply the customization in line with the new version.
5. Test the implementation to ensure that everything is working as expected.

**Patch Upgrades**

Patch upgrades require applying patches of code to meet a retailer’s specific business needs. Patch upgrades are usually less disruptive compared to full version upgrades.

ESW recommends that you apply patch upgrades only in cases where there is a specific requirement and it is not feasible for you to wait until the next version is released. Note that ESW prepares all patches and shares them directly with the retailer.

Each patch includes the following files:

* **Readme.docx**: This file includes the steps that must be performed to apply the patch with the line numbers and file paths. You must perform the steps exactly as outlined in this file.
* **Reference code**: These files contain the marked code files that require the upgrade. The changes must be applied as described in the **Readme.docx** file. Do not copy and paste the contents of the reference code files.

**Additional Information**

Each time a new version of the ESW cartridge is rolled out, the ESW online documentation is updated with the list of new features and changes. To stay up-to-date with the latest release information, we recommend that you bookmark this page: [Release Notes](https://know.esw.com/space/TDR/1234639152/SFCC+Release+Notes).

To get access to the latest version of the cartridge, contact the ESW-SFCC team, who will provide you the latest code and relevant documentation.

ESW submits the cartridge to Salesforce for certification every 6 months. This is primarily due to the overall time involved in the certification process.

Go-Live Readiness

This section describes how to prepare to go-live with ESW:

1. Check that the ESW plugin/cartridge is fully up-to-date and the latest code files are replicated from the Staging environment to Production.

2. Check that all ESW related site preferences and the other site data is replicated from the Staging environment to Production.

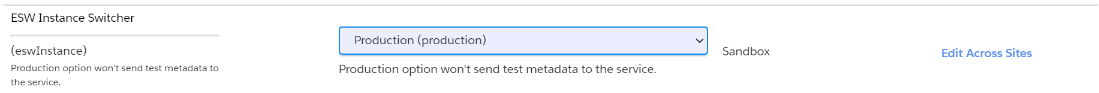
* To check ESW Custom Site Preference, navigate to ***Merchant Tools > Site Preferences > Custom Site Preference Groups.***
* To check Order - Attribute Definitions, navigate to ***Administration > Site Development > System Object Types > Order - Attribute Definitions.***
* To check Product Line Item - Attribute Definitions, navigate to ***Administration > Site Development > System Object Types > Product Line Item - Attribute Definitions***.
* To check Jobs, navigate to  ***Administration > Operations > Jobs***.
* To check Services, navigate to ***Administration > Operations > Services***.

3. Check that all shipping and payment methods are replicated from the Staging environment to Production, especially ESW shipping and payment methods.

* To check Shipping Methods, navigate to ***Merchant Tools > Ordering > Shipping Methods***.
* To check Payment Methods, navigate to ***Merchant Tools > Ordering > Payment Methods***. Ensure that the *ESW\_Payment* payment method is present

4. Ensure that Tax settings are configured in SFCC BM *(Merchant Tools > Ordering > Taxation Settings - Tax Rates*) to sync the tax totals in SFCC and ESW.

5. Ensure that the **ESW Instance Switcher** option under **ESW General Configuration** is set to **Production**. You can access this option by navigating to  ***Merchant Tools > Site Preferences > Custom Site Preference Groups.***



When set to **Production**, the **ESW Instance Switcher** option removes the preorder metadataitems test details as follows:

"MetadataItems": [

{

"Name": "OrderConfirmationUri\_TestOnly",

"Value": "https://development.example.com/on/demandware.store/Sites-sw-us-Site/en\_US/EShopWorld-Notify"

},

{

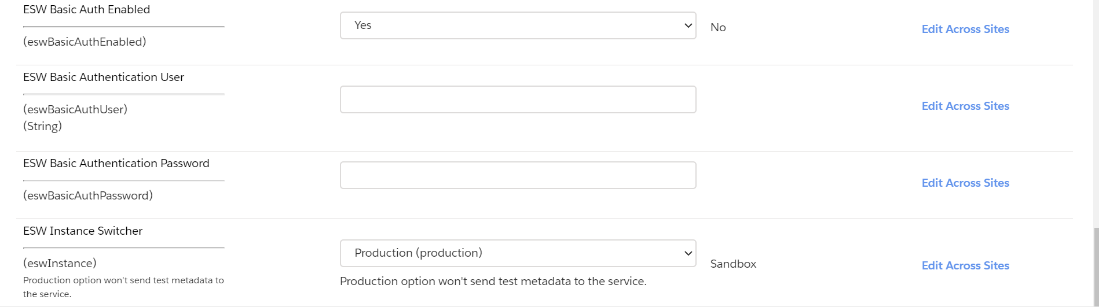
"Name": "OrderConfirmationBase64EncodedAuth\_TestOnly",

"Value": "xxxxxxxxxxxxxxxxx"

}

]

6. Enable basic authentication and enter the username and password. This username/password information is provided by the retailer to protect the order confirmation endpoint. You can enable basic authentication under [ESW General Configuration](#ESWGeneralConfiguration).



7. Configure or use the standard Order Confirmation endpoint in SFCC. The standard Order Confirmation endpoint looks like this: https://www.example.com/EShopWorld-Notify.

8. Share the order confirmation endpoint and basic authentication login details with the ESW Integration Engineer/Onboarding Team.

9. Ensure that the jobs are scheduled and pricing advisor data is present in the ESW\_PA Custom objects. While configuring jobs please make sure that the ESW Retailer Pricing Feed job is scheduled **before** ESW Retailer Auto configurator job. Also, the ESW Countries & ESW Currencies custom objects should be populated with countries and currencies respectively.

SFCC FAQs

This section contains questions and answers that you might find helpful when working with the ESW cartridge. Click on a question to view the answer.

How does pricing conversion work?

For pricing conversions at several places, such as product titles and product details page, the ESW cartridge provides the getMoneyObject(price, noAdjustment, formatted) function. This function converts the product prices as per the shopper’s currency and country selection. In this function:

1. Price can be a number or a money object.
2. noAdjustment can be a Boolean value. If set to True, no adjustment is applied.
3. Formatted can be a Boolean value. If set to True, it returns a string value with currency code, for example, $10.00. If set to False, a money object is returned with the currency and price.

This function is a helper function that is used wherever prices need to be displayed. To use this function, add the following code on a page. For example, if you want to add pricing conversion on producttile.isml, which is used to add price on product tiles, add the following code at the top of the page:

< var eswHelper = require('int\_eshopworld\_controllers/cartridge/scripts/helper/eswHelper').getEswHelper(); >

To display updated prices as per the currency and country selection, for example, on the Cart page and the product display page, modify the isml file and add the following code:

< var eswHelper = require('int\_eshopworld\_controllers/cartridge/scripts/helper/eswHelper').getEswHelper(); >

How can I add a header bar, footer bar, and welcome landing page?

* To add a header bar, add < isinclude url="${URLUtils.url('EShopWorld-GetEswHeader')}"/ > to your cartridge’s header.isml.
* To add a footer bar, add < isinclude url="${URLUtils.url('EShopWorld-GetEswFooter')}"/ > code to your cartridge’s footer.isml.
* To add a welcome landing page, add < isinclude url="${URLUtils.url('EShopWorld-GetEswLandingPage')}"/ > in either header.isml or footer.isml of the cartridge.

The landing page appearance depends on the cookie variable (esw.Landing.Played). The cookie is initially set to false. Once the landing page appears, the cookie value is set to true. The landing page will not be displayed until the cookie is cleared.

What happens during a failover?

If the service times out or if there is an error with the service, the shopper is redirected to the Cart page and an error is displayed on the screen. The shopper will not be able to go to the next page and continue the flow because the cartridge redirects to the hosted page. Therefore, if the service is down, shoppers cannot proceed further.

What cookies does ESW create?

The following cookies are created and managed by ESW:

* **esw.Landing.Played**: Set to true once the welcome landing page is shown.
* **esw.LanguageIsoCode**: Stores the locale selected on the site on first-time page load. If the shopper changes the locale, then this cookie is updated with the latest locale.
* **esw.location**: Stores the country code selected by the shopper.
* **esw.currency**: Stores the currency code selected by the shopper.
* **eswInternationalUser**: Set to true if the selected country is enabled in ESW.
* **esw.sessionid**: Stores the shopper’s session ID.
* **esw-shopper-access-token**: Stores the saved credit card information

These cookies work as expected based on the Google Chrome browser cookie feature update (Chrome version: 80.0.3987.132).

Why can’t I see the sales tax information on the Cart page?

SFRA provides sales tax information on the cart page. The information is hidden when ESW supported countries are selected. ESW takes sales tax into account while calculating the price and displays it on the storefront. The sales tax is included in the price of the product. Therefore, the sales tax information is only displayed for ESW non-supported countries. For ESW supported countries, the sales tax information on the cart page is not displayed.

How long does it take for an order to be marked as ‘Failed’?

An order is marked as Failed usually within 30 minutes. However, you can configure this by navigating to Merchant Tools > Site Preferences > Order.

Why can’t I see the Shipping Method dropdown on the Cart page?

SFRA provides a drop-down menu to select shipping methods on the cart page. As ESW overrides the shipping method and sends it through the preorder request to the API, the shipping methods can be changed or overridden, resulting in changed rates accordingly as per the override functionality. Therefore, the shipping method option on the cart page is only displayed for ESW non-supported countries. For ESW supported countries, the Shipping Method dropdown on the cart page will not be displayed.

Does SFCC reduce inventory when an order is created during the 'Cart to eSW' handover?

Yes. This is native SFCC behaviour to avoid overselling.

Why does an order show as ‘Created’ during ‘Cart to ESW' handover?

As required by Salesforce, an order with a ‘Created’ status is required for our cartridge to be certified.

When does the SFCC native checkout gets displayed to the shoppers?

For ESW non-supported countries, shoppers will go through the SFCC native checkout and will not be redirected to ESW Checkout.

Can only the fixed pricing model be supported without FX rates, uplifts, and rounding data being present in SFCC?

Yes, by verifying and ensuring the following:

* Verify that the flag ‘is Fixed Price Mode’ in every country’s custom object of type ESW Country is checked.
* Ensure that these currencies are allowed currencies and are added in Business Manager under **Merchant Tools > Site Preferences > Currencies.**
* Note that the **ESW FX Rates JSON, ESW Country Adjustment JSON**, and **ESW Rounding Rules** in the ESW\_PA\_DATA custom object can be empty.

Why do discounted prices require page refresh on the Cart page?

This is because of an internal framework issue. We recommend the following workaround provided by SFCC. These steps must be performed at your own discretion and should be removed when framework fix is available from SFCC.

**Workaround**: In SFRA, you can include the output of another controller, for example, < isinclude url=“${URLUtils.url(‘Cart-MiniCart’)}” / >. The isinclude is also called a 'remote include' and it allows you to create pages that have different caching in different areas. Click [here](https://documentation.b2c.commercecloud.salesforce.com/DOC1/topic/com.demandware.dochelp/DWAPI/scriptapi/html/api/class_dw_web_URLUtils.html?resultof=%22%55%52%4c%55%74%69%6c%73%22%20%22%75%72%6c%75%74%69%6c%22%20)to see the Element.

Why am I stuck on cart page and unable to redirect to ESW Checkout?

This mostly happens due to the ‘RetailerCartId is required’ message, which appears if SFCC is unable to create an order. The 'orderNo' is the 'RetailerCartId' in the request that is sent to ESW Checkout API. You can perfrom the following steps to ensure that order is being created in SFCC.

* Verify that the valid shipping methods are configured and assigned to the **ESW Override Shipping** custom preference.
* Verify that the tax classes are being applied to the order. This is required by SFCC to create the order.
* Verify that **ESW\_PAYMENT** is configured under **Business Manager > Merchant Tools > Ordering > Payment Methods.**

Why my SFCC order level system attribute values do not match with ESW order confirmation custom attribute?

If shipping cost is different in SFCC and ESW than retailer has an option to replicate ESW shipping methods in SFCC. Check following link for more information.

<https://know.eshopworld.com/space/AE/255721518/Shipping+Methods+Integration>

Also make sure tax default configuration is set to zero in SFCC BM (*Merchant Tools > Ordering > Taxation Settings - Tax Rates*).

## **Why don’t I see stored cards on the checkout?**

If you are not able to see the stored cards on the checkout, then:

* Ensure that the custom preference value “ESW Top Level Domain” is not empty, it contains the top-level domain. For example, if the storefront URL is [http://www.abc.xyz.com](http://www.abc.xyz.com/) , then the preference should contain .xyz.com
* The top-level domain should be the same as the storefront. If the domains are different, then follow the steps [here](https://know.eshopworld.com/space/AE/723714049/Stored+Payment+Cards+Integration#Options-For-Testing-In-Sandbox%2FStaging-Environment).
* Confirm with the CSM if the stored cards are enabled for your sandbox/production environment.