Extend Integration for Salesforce Commerce Cloud SFRA

Welcome to Extend! We provide an easy way for any merchant to sell extended warranties - generating new revenue, increasing purchase conversion, and dramatically improving the customer experience.

This guide will walk you through the process of getting live with Extend protection plans, so you can start selling and earning revenue as quickly and easily as possible. The process is pretty simple - and even more simple with a connected SFCC reference application SFRA v4.4.0. Read on to learn more!

Table of Contents

Component Overview

Cartridge Features

This integration cartridge makes it easy to enable the following features:

- Automated product catalog export to Extend
- Ability to enable/disable the main functionalities for Extend warranties in Business Manager
- Display simple Extend protection plan offers in the PDP & Cart
- Display Extend interstitial modal protection plan offers in PDP & Cart
- Automatically generate Extend protection plan contracts upon order placement whenever warranties are sold

By enabling extend in your SFRA instance, your customers can:

- Attach an Extend warranty to a catalog product, in PDP, Offer Modal or Cart and subsequently checkout products
- View details about the warranty products
- Remove warranties from the Cart (either by removing the main product or just the warranty product)

Compatibility

This cartridge has been developed and tested against Commerce Cloud Digital 19.10 and integrated on top of SFRA 4.4.0.

Privacy Policy on Payments

This cartridge implements a custom object for temporary storing order data needed for Extend warranty contract generation. This custom object (ExtendContractsQueue) was implemented based on a queue pattern that gets consumed by a job configured to execute every 5 minutes. This data is not exposed publicly and used only in a server-to-server communication setup.

Extend does not handle payments from the end-user. The merchant is responsible for charging the end-user for the warranty plans purchased, using the existing tenders that are currently supported on the specific environment. The merchant is responsible for generating Extend contracts only after charging the end-user for the warranty plans purchased. Please visit https://extend.com/privacy for more information.

Implementation Guide

The SFRA version of this integration follows the extensibility and modularity design principles behind SFRA. It's been designed to live on top of the storefront-base-architecture components (app_storefront_base & modules). This means it's required that this cartridge is included in an environment that follows the SFRA recommendations.

Business Manager Configuration

Upload cartridge & include in the cartridge path

1. Upload the **int_extend_sfra** cartridge to the sandbox.
2. Go to Business Manager → Administration → Site → Manage Sites. Select desired site and go to Settings. At the beginning of the Cartridge Path add "int_extend_sfra:" in order to include the cartridge in the current site's configuration.

Import metadata

1. Go to Business Manager → Administration → Site Development → Import & Export
2. Upload the following files from ./metadata/meta/ folder
    a. system-objecttype-extensions.xml
    b. custom-objecttype-definitions.xml
3. Go to Business Manager → Administration → Site Development → Import & Export → Metadata → Import. You should see both previous files as available for import. Proceed to importing both.

Import custom jobs

1. Go to Business Manager → Administration → Operations → Import & Export
2. Upload the following file: ./metadata/jobs.xml
3. Go to Business Manager → Administration → Operations → Import & Export → Jobs → Import. You should see the previous file as available for import. Proceed to importing it.

Import service

1. Go to Business Manager → Administration → Operations → Import & Export
2. Upload the following file: ./metadata/services.xml
3. Go to Business Manager → Administration → Operations → Import & Export → Services → Import. You should see the previous file as available for import. Proceed to importing it.
4. Note: by default, the service URL is configured to point to the demo instance of the Extend API. Prior to going live with the integration, this will need to be changed and tested against the live production API.

Import Extend catalog

1. Select the site you're integrating Extend with
2. Go to Business Manager → Merchant Tools → Products and Catalogs → Import & Export
3. Upload the following file: ./metadata/catalog.xml. This will import 3 dummy option products, which are used in order to create dynamic product line items in cart. We recommend keeping this as a separate master catalog, for visibility purposes.

Configuration

Catalog setup

The 3 Extend products that need configuration are:

- EXTEND-12
- EXTEND-24
- EXTEND-36

In order to make the Extend products merchandisable, you need to:

- Assign them to your storefront catalog
- Allocate an inventory record in the inventory file assigned to the current site
  - Configure this entry as perpetual, since these are **intangible goods** and **need to always be in-stock**
- These do not need any description, pricing or any other regular product attributes to be configured

Custom preferences

1. Go to Business Manager → Merchant Tools → Site Preferences → Custom Preferences → Extend
2. Configure the provided Extend credentials in the following fields
   a. Access Token: this is used for authenticating web service calls to [Extend's API](#)
   b. Store ID: this is used for the client-side Javascript SDK to identify the current store
3. Configure the image breakpoint for using when exporting the product feed:
   a. The Image View Type is by default configured to large
   b. Configure this to specify the breakpoint for the largest image size you have available across your products
4. Extend Javascript SDK URL
   a. By default, this points to the demo JS SDK. Make sure that by the end of the integration process this is switched to the live production version
5. Use the provided feature switches in order to enable / disable Extend features
   a. Enable Extend- Enables/disables Extend globally
   b. Enable Up-sell in Cart - Enables/disables the up-sell button and modal in Cart
   c. Enable in PDP - Enables/disables Extend in PDP
   d. Enable Up-sell in PDP - Enables/disables the interstitial modal in PDP

Jobs

Extend Products Export

This job uses Extend's API to traverse the entire storefront catalog and export each Simple and Variant Product that is, at the time of the job run, merchandisable (online and in stock):

1. Go to Business Manager → Operations → Jobs. Select the Extend Products Export job.
2. Go to Job Steps and select Scope. Configure this to run in the scope of the site you're implementing Extend for.
3. Schedule this to run once a day, at a time that's most suitable for your recurring job setup, ideally after the catalog has been updated.
4. Make sure you're running this once and notify Extend after the successful run. Warranty products or plans are dynamically generated based on this stream of products.

Extend Contracts Creation

This job uses Extend's API to consume contracts queue, stored in the ExtendContractsQueue custom object. As Extend plans are ordered, an instance of this object is created. Thereafter, this job will send a contract generation request for each queue element – subsequently, it will remove the request from the queue.

Custom Code

Custom changes are required to the calculate.js hook. First, open your calculate.js hook script (*/cartridge/scripts/cart/calculate.js) and find the function `calculateProductPrices(basket) {...}`.

Change the "handle non-catalog products" if statement to be:

```
if (!productLineItem.catalogProduct ||
productLineItem.custom.parentLineItemUUID)
```

```
 96    /**
 97     * @function calculateProductPrices
 98     *
 99     * Calculates product prices based on line item quantities. Set calculates prices
100     * on the product line items.  This updates the basket and returns nothing
101     *
102     * @param {object} basket The basket containing the elements to be computed
103     */
104    function calculateProductPrices(basket) {
105        // get total quantities for all products contained in the basket
106        var productQuantities = basket.getProductQuantities();
107        var productQuantitiesIt = productQuantities.keySet().iterator();
108
109        // get product prices for the accumulated product quantities
110        var productPrices = new HashMap();
111
112        while (productQuantitiesIt.hasNext()) {
113            var prod = productQuantitiesIt.next();
114            var quantity = productQuantities.get(prod);
115            productPrices.put(prod, prod.priceModel.getPrice(quantity));
116        }
117
118        // iterate all product line items of the basket and set prices
119        var productLineItems = basket.getAllProductLineItems().iterator();
120        while (productLineItems.hasNext()) {
121            var productLineItem = productLineItems.next();
122
123            // handle non-catalog products
124            if (!productLineItem.catalogProduct || productLineItem.custom.parentLineItemUUID) {
125                productLineItem.setPriceValue(productLineItem.basePrice.valueOrNull);
126                continue;
127            }
128
129            var product = productLineItem.product;
130
131            // handle option line items
132            if (productLineItem.optionProductLineItem) {
133                // for bonus option line items, we do not update the price
134                // the price is set to 0.0 by the promotion engine
135                if (!productLineItem.bonusProductLineItem) {
```

Testing

Enabling Extend

- Set the following configuration options to 'Yes'
    - o Enable Extend
    - o Enable in PDP
    - o Enable Up-sell in PDP
    - o Enable Up-sell in Cart
- Ensure the product catalog has been exported to Extend and your Extend representative enabled warranty plans in the system

PDP Offers

Go to any simple or variant product that has an Extend warranty enabled. Select a variation and you will be presented with the Extend offer and widget to select a warranty plan.
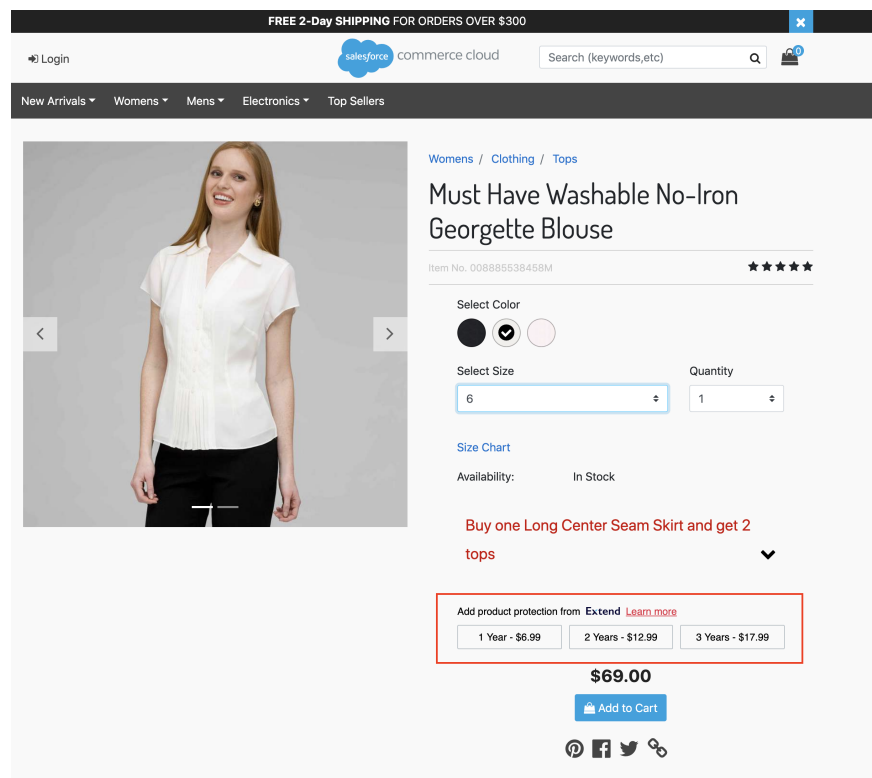
Adding a Warranty to the Cart

Select an Extend warranty plan and click Add to Cart. You will notice 2 products were added to the cart – the SFCC catalog product along with the Extend warranty plan.

Interstitial Modal

Click Add to Cart on the PDP without selecting any Extend warranty plan. You will be presented with a modal recommending the associated Extend plans for this current catalog product. Selecting "Add protection plan" will proceed to add to cart the Extend plan, alongside the catalog product. Selecting "No, thanks" will only add the catalog product to the cart.
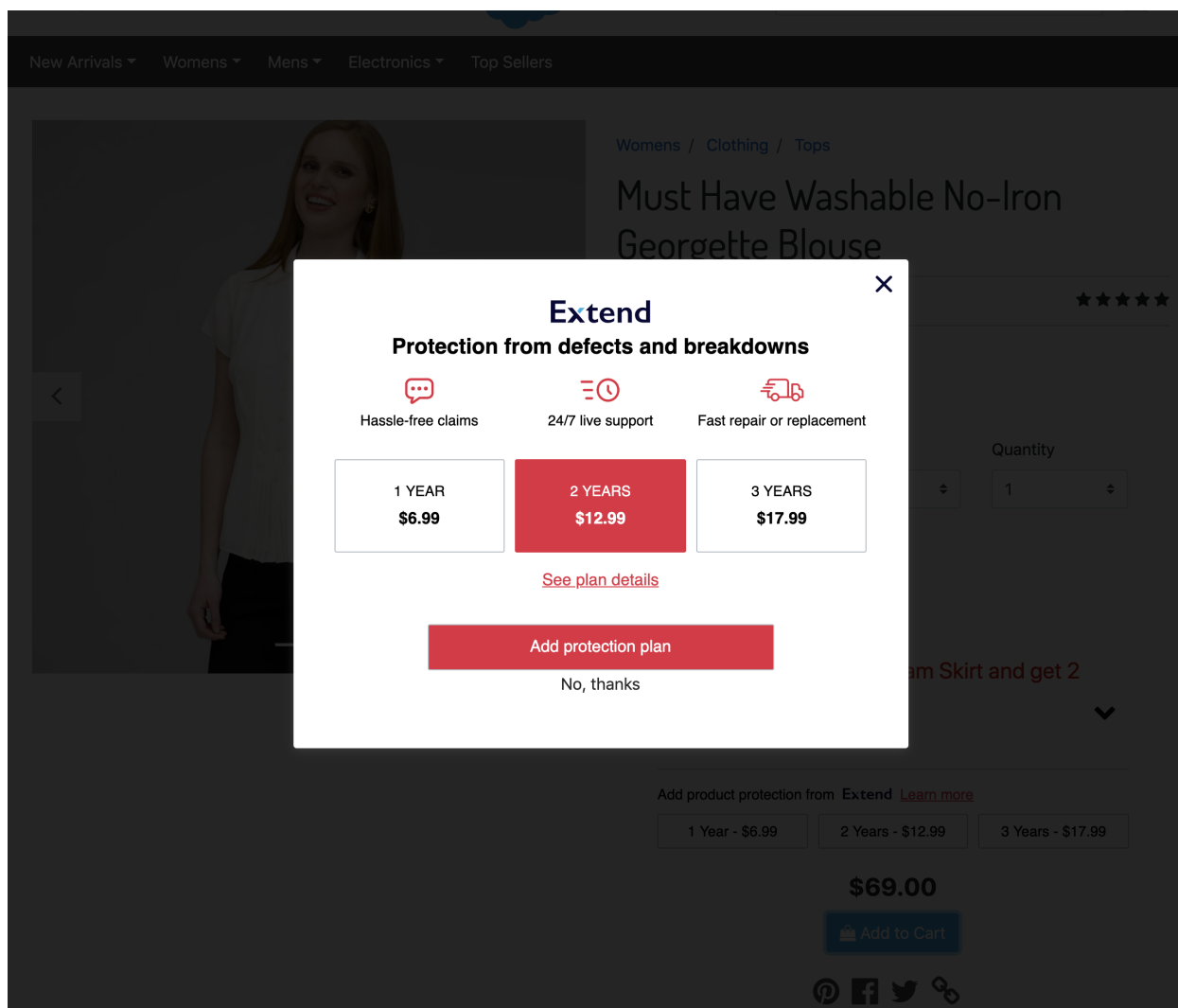
Cart Offers

Extend Plans in Cart

Confirm an Extend plan has been added to the cart.

Up-sell button for eligible products in cart

Add to cart an eligible for warranty product, without selecting any Extend plan for it. Go to the cart page. You will notice a button that recommends adding a protection plan.

Interstitial Modal in Cart

Click on the up-sell button. You will be presented with a modal recommending the associated Extend plans for this current catalog product. Selecting "Add protection plan" will proceed to add to cart the Extend plan. Selecting "No, thanks" will close the modal.

Business Manager

ExtendContractsQueue custom object

After placing an order containing Extend warranty plans, a new instance of this custom object is created. This holds all the data needed for generating an Extend contract.

Merchant Tools > Custom Objects > Custom Objects > 8f6c7fc79e60c41637321dab6a - General

General

Manage '8f6c7fc79e60c41637321dab6a' (ExtendContractsQueue)

Fields with a red asterisk (*) are mandatory. You can view and edit the name and description in other languages, if required. Click **Apply** to save the details.

contractData

Site Currency:	USD
Extend Warranty Plan:	{"purchasePrice":1364,"planId":"10001-misc-elec-base
SFCC Product:	{"referenceId":"008885538465M","purchasePrice":724
Last Modified:*	12/18/2019 : 9:50 pm
	MM/dd/yyyy h:mm a
Creation Date:*	12/18/2019 : 9:50 pm
	MM/dd/yyyy h:mm a
Order Customer:	{"phone":"4157477799","email":"kris@extend.com","nan
Log:	
Line Item UUID:*	8f6c7fc79e60c41637321dab6a
Order Total:	9,238.00 (Number)
Order Number:	00000907

Extend Contracts Creation Job

This job will process the ExtendContractsQueue object instances and will generate a contract for each instance. After successfully executing the request, the objects will be deleted.

Questions? Feedback?

At Extend, customer experience is our North Star, and that goes for both our merchant customers (you!) and your store's customers as they buy your products and our protection plans. So, we want to do anything we can to make your Extend integration successful and to make your customers' experiences outstanding.

If you have any questions, please don't hesitate to reach out to us any time or contact integrations@extend.com for help. And if you have any feedback about our documentation, integration process, or our APIs and SDK, we are all ears!