

**Affirm Promo - SFRA version**

***Version 19.******1.0***

***SFRA 4.4.0***



**Table of Contents**

[1. Summary 3](#_Toc48741614)

[2. Component Overview 4](#_Toc48741615)

[2.1 Functional Overview 4](#_Toc48741616)

[2.2 Use Cases 4](#_Toc48741617)

[2.3 Limitations, Constraints 4](#_Toc48741618)

[2.4 Compatibility 4](#_Toc48741619)

[2.5 Privacy, Payment 4](#_Toc48741620)

[3. Implementation Guide 5](#_Toc48741621)

[3.1 Setup 5](#_Toc48741622)

[3.1.1 Deploying cartridge to a sandbox 5](#_Toc48741623)

[3.1.2 Sandbox setup 6](#_Toc48741624)

[3.2 Configuration 7](#_Toc48741625)

[3.2.1 Site Preferences configuration 7](#_Toc48741626)

[3.3 Custom Code (SFRA) 10](#_Toc48741627)

[Templates 10](#_Toc48741628)

[Client JavaScript 13](#_Toc48741629)

# Summary

This document describes how to implement the Affirm cartridge in Storefront Reference Architecture SiteGenesis site. This cartridge can be configured in the Business Manager and contains all elements necessary to perform a successful best practices implementation of Affirm.

# Component Overview

## Functional Overview

“Buy with Affirm” is a closed-end installment loan product offered by [Cross River Bank](http://www.crossriverbank.com/) through Affirm’s technology platform. “Buy with Affirm” allows you to buy goods or services offered by online merchants with whom Affirm partners (“Merchants”). If you agree to use “Buy with Affirm,” Cross River Bank will pay the Merchant on your behalf in exchange for your promise to repay the same amount plus a finance charge as determined by your creditworthiness.

## Use Cases

On the PDP, PLP and Cart Page, Customers can see Affirm promotions beneath the price displayed monthly payment.

## Limitations, Constraints

Affirm cartridge has the following limitation for purchasing products. List of goods that is excluded from cartridge support for buying:

Gift Certificates

## Compatibility

Cartridge is designed and developed for: Salesforce platform version 19.1 (Compatibility Mode 19.1).

## Privacy, Payment

Affirm stores and processes your information maintaining physical, electronic and procedural safeguards. We maintain physical security measures to guard against unauthorized access to systems and use safeguards such as firewalls and data encryption. We enforce physical access controls to our buildings, and we authorize access to personal information only for those employees who require it to fulfill the responsibilities of their jobs.

# Implementation Guide

## Setup

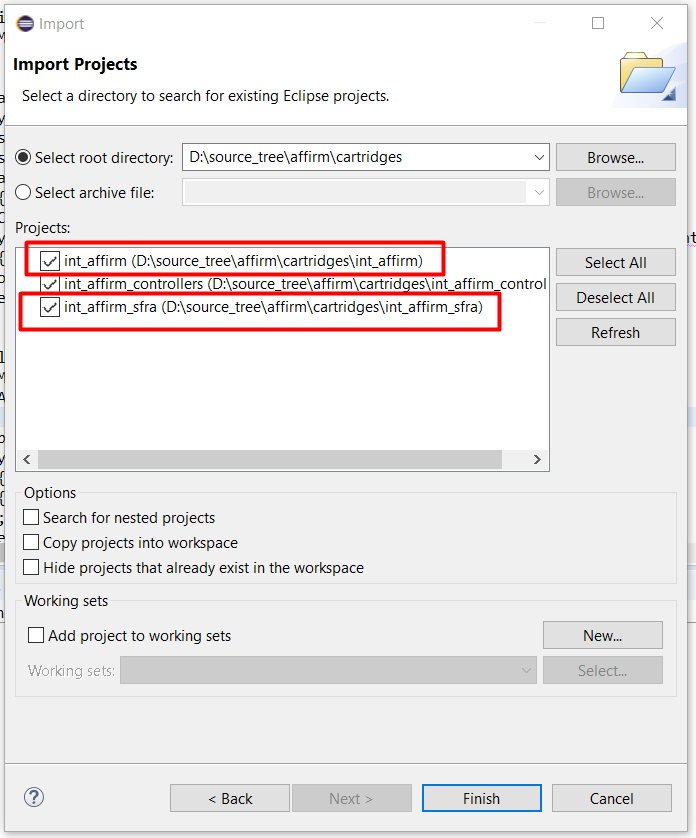
This section describes steps that should be completed before cartridge configuration in Business Manager.

### Deploying cartridge to a sandbox

1. Download cartridge source code.
2. Import the Affirm cartridge to a workspace in Salesforce UX Studio.



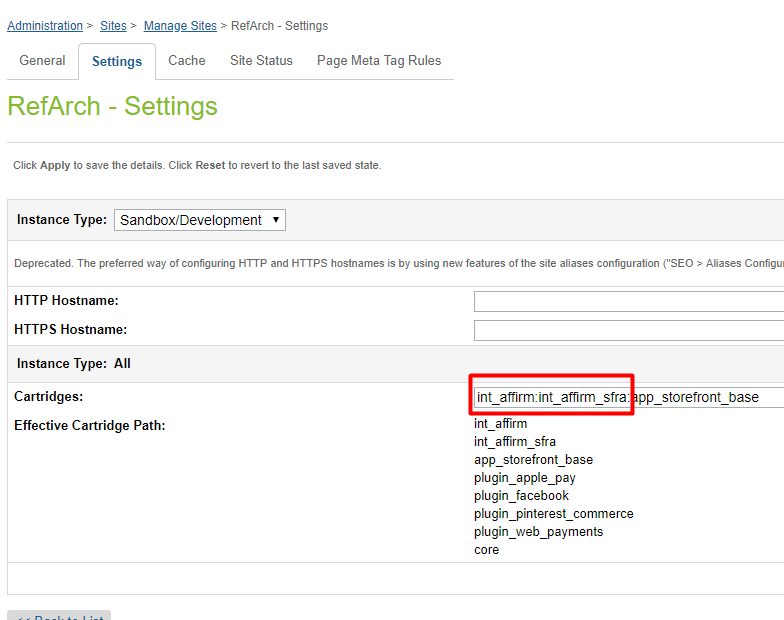
1. Add Affirm cartridges to Project Reference of Server Connection.



1. Wait until Studio completes workspace build and uploading source codes to a sandbox.

### Sandbox setup

1. Go to Business Manager -> Site -> Manage Sites. Select the correct site, then select Settings tab. In the cartridge path at the beginning write the following: int\_affirm\_sfra:int\_affirm:



1. Download cartridge source code.
2. Find metadata/affirm/meta/system-objecttype-extensions.xml
3. Go to Administration > Site Development > Import & Export > Upload Import Files. Click the Upload button and select system-objecttype-extensions.xml  
   A screenshot of a cell phone

   Description automatically generated
4. Go to Meta Data > Import and import the uploaded file.

A screenshot of a cell phone

Description automatically generated

1. Verify that Site Preferences group was created with ID - AFFIRM\_PROMO and name – Affirm.

## Configuration

This section describes configuration of the sandbox.

### Site Preferences configuration

1. Go to Merchant Tools > Site Preferences > Custom Preferences> Affirm
2. Enable attribute - Affirm Online Status. This attribute defines status (enable/disable) of the Affirm integration.
3. Select a value from the dropdown Affirm Mode. This attribute defines in which mode the cartridge will work. Allowable values are “Sandbox” and “Production”.
4. Add site preference attribute - Affirm Public Key with provided public key from Affirm.
5. Add site preference attribute - Affirm Private Key with provided private key from Affirm.
6. Enable attribute - Cart Promo Message. This attribute defines status (enable/disable) of Promotional messages on Cart Page.
7. Enable attribute - Product Promo Message. This attribute defines status (enable/disable) of Promotional messages on Product Detail Page.
8. Enable attribute - PLP Promo Message. This attribute defines status (enable/disable) of Promotional messages on Product Listing Page.
9. Add site preference value attribute - Default financing program. This attribute defines which financing program will be used by default (if any other financing programs have not been applied).
10. Add site preference value - Affirm Financing programs mapping. It allows mapping financing programs to promo-id.

Please use the following format: {financing program name}|{promo-id}

Example:



1. Add site preference value - Financing programs customer group. This allows detection of financing programs based on customer group.
2. Please use the following format: {customerGroup}|{financingProgram},
3. Example: Registered|registeredFinProgram



1. Add site preference value - Financing programs cart total range. This allows detection of financing programs based on cart total range.
2. Please use the following format: {minValue}|{maxValue}|{financingProgram}.
3. Example: 100|200|CartRange



1. Add site preference value - Financing programs date range. This allows detection of financing programs based on date range.
2. Please use the following format: {startDate}|{endDate}|{financingProgram}, where startDate and endDate format is "yyyy-MM-dd".
3. Example: 2017-01-01|2017-02-28|myWinter



1. Add site preference value - Affirm Min Total. It allows setting minimal price value for displaying promotional messages. Products with price lower than this value will not have a promotional message.
2. Add site preference attribute - Payment Min Total with minimal cart total when Affirm payment method is available.
3. Add site preference attribute - Payment Max Total with maximal cart total when Affirm payment method is available.

## Custom Code (SFRA)

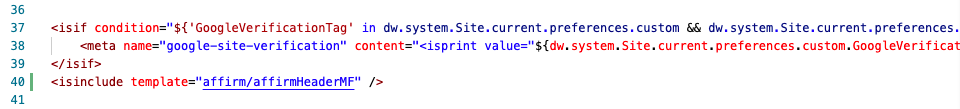
This section describes changes that should be made to a merchant’s custom cartridge. All of the modifications are stored in the mirrored cartridge structure in **int\_affirm\_sfra/\_cartridge** for default Storefront Reference Architecture so you can find the exact code there.

### Templates

1. **cartridge/templates/default/common/htmlHead.isml**

Include Affirm header script tag after Google Verification tag (or after any other relevant tags *and* before the htmlHead hook):

<isinclude template="affirm/affirmHeaderMF" />



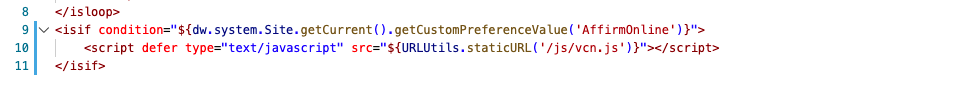
1. **cartridge/templates/default/common/scripts.isml**

Include vcn.js as static file at the bottom of the file:

<isif condition="${dw.system.Site.getCurrent().getCustomPreferenceValue('AffirmOnline')}">

<script defer type="text/javascript" src="${URLUtils.staticURL('/js/vcn.js')}"></script>

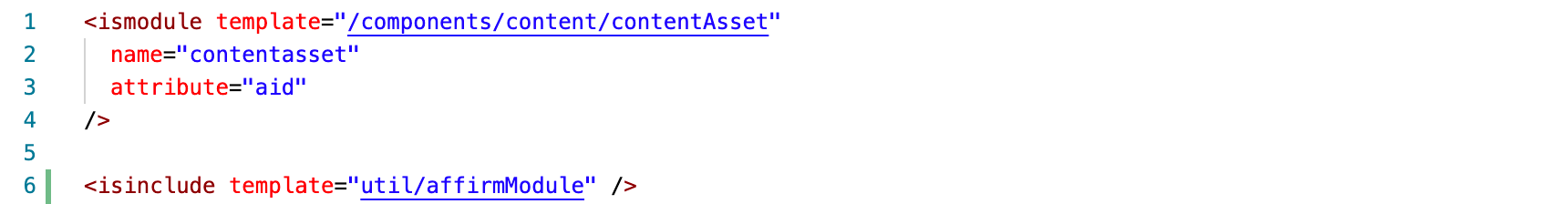
</isif>



1. **cartridge/templates/default/components/modules.isml**

Include affirmModule template after the last line:

<isinclude template="util/affirmModule" />



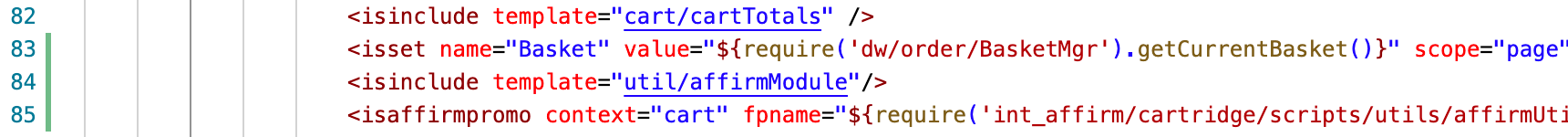
1. **cartridge/templates/default/cart/cart.isml**

Place promo message below where cart total is displayed:

<isset name="Basket" value="${require('dw/order/BasketMgr').getCurrentBasket()}" scope="page" />

<isinclude template="util/affirmModule"/>

<isaffirmpromo context="cart" fpname="${require('int\_affirm/cartridge/scripts/utils/affirmUtils').getFPNameByBasket(Basket)}" />



1. **cartridge/templates/default/product/**

There is a set of recurring snippets that need to be placed in order for the Affirm promo message to be shown on the various product related pages. The list of templates that correspond to the pages include setDetails, productDetails, bundleDetails, quickView and setQuickView in the product folder.

**setDetails.isml**

Place promo message where set price is displayed:

<isinclude template="util/affirmModule" />

<isaffirmpromo context="pdp" fpname="${require('\*/cartridge/scripts/utils/affirmUtils').getFPNameForPDP(product)}" >

**bundleDetails.isml**

Similar steps as the previous file:



**productDetails**

Similar steps as the previous file:



**quickView.isml & setQuickView.isml**

In order for Affirm promo and Express Checkout to appear on quick views, the following should be included to **/default/search/searchResults.isml**:

****

1. **product/productTile.isml**

Paste the following code where the Affirm promo message should be shown on the product listing page:

<isinclude template="util/affirmModule"/>

<isaffirmpromo context="plp" fpname="${require('int\_affirm/cartridge/scripts/utils/affirmUtils').getFPNameForPLP(product)}" price="${price}"/>



### Client JavaScript

Note: Client-side JavaScript will need to be compiled before deployment. Update package.json at the root to make sure the value of paths property is referencing the base cartridge. In addition, please make sure the [SGMF Scripts](https://www.npmjs.com/package/sgmf-scripts) package is installed and run the following command:

sgmf-scripts --compile js

More information on the build commands can be found [here](https://documentation.b2c.commercecloud.salesforce.com/DOC3/topic/com.demandware.dochelp/SFRA/BuildingSFRA.html).

1. **js/cart/cart.js**

Add recalculation for the Affirm Modal Promo price on re-rendering of order totals

var totalCalculated = data.totals.grandTotal.substr(1).replace(/,/g, '');

$('.affirm-as-low-as').attr('data-amount', (totalCalculated \* 100).toFixed());

var isWithinAffirmLimit = (parseFloat(totalCalculated) >= parseFloat(affirmLimits.min) && parseFloat(totalCalculated) <= parseFloat(affirmLimits.max))

if (isWithinAffirmLimit) {

$('#js-affirm-checkout-now').show();

} else {

$('#js-affirm-checkout-now').hide();

$('.affirm-as-low-as').attr('data-amount', NaN);

}

affirm.ui.refresh();

1. **js/search/search.js**

Update showMore method to include affirm.ui.refresh as a success callback as shown below:

showMore: function () {

// Show more products

$('.container').on('click', '.show-more button', function (e) {

e.stopPropagation();

var showMoreUrl = $(this).data('url');

e.preventDefault();

$.spinner().start();

$(this).trigger('search:showMore', e);

$.ajax({

url: showMoreUrl,

data: { selectedUrl: showMoreUrl },

method: 'GET',

success: function (response) {

$('.grid-footer').replaceWith(response);

updateSortOptions(response);

affirm.ui.refresh()

$.spinner().stop();

},

error: function () {

$.spinner().stop();

}

});

});

},

Note: Make sure to rebuild client JS after making modifications.