

**Affirm[[1]](#footnote-1)**

***Version 18.1.0***

****

**Table of Contents**

[1.](#_3znysh7) Summary 1-3

[2.](#_2et92p0) Component Overview 2-4

[2.1](#_tyjcwt) Functional Overview 2-4

[2.2](#_3dy6vkm) Use Cases 2-4

[2.3](#_4d34og8) Limitations, Constraints 2-4

[2.4](#_2s8eyo1) Compatibility 2-4

[2.5](#_17dp8vu) Privacy, Payment 2-4

[3.](#_3rdcrjn) Implementation Guide 3-5

[3.1](#_26in1rg) Setup 3-5

[3.1.1](#_lnxbz9) Deploying cartridge to a sandbox 3-5

[3.1.2](#_35nkun2) Sandbox setup 3-6

[3.2](#_1ksv4uv) Configuration 3-9

[3.2.1](#_44sinio) Site Preferences configuration 3-9

[3.2.2](#_2jxsxqh) Affirm Services configuration 3-12

[3.2.3](#_z337ya) Affirm Job configuration 3-12

[3.3](#_3j2qqm3) Custom Code (SFRA) 3-14

[3.5](#_4i7ojhp) External Interfaces 3-17

[3.6](#_2xcytpi) Testing 3-17

[4.](#_1ci93xb) Operations, Maintenance 4-18

[4.1](#_3whwml4) Data Storage 4-18

[4.2](#_2bn6wsx) Support 4-18

[5.](#_3as4poj) User Guide 5-19

[5.1](#_1pxezwc) Roles, Responsibilities 5-19

[5.2](#_2p2csry) Business Manager 5-19

[5.3](#_3o7alnk) Storefront Functionality 5-19

[6.](#_23ckvvd) Known Issues 6-20

[7.](#_ihv636) Release History 7-20

# Summary

*This document describes how to implement 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

Customers can use Affirm payment method to pay their purchases.

On the PDP, 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 18.5, Storefront Reference Architecture SiteGenesis and Compatibility mode 17.7

## 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

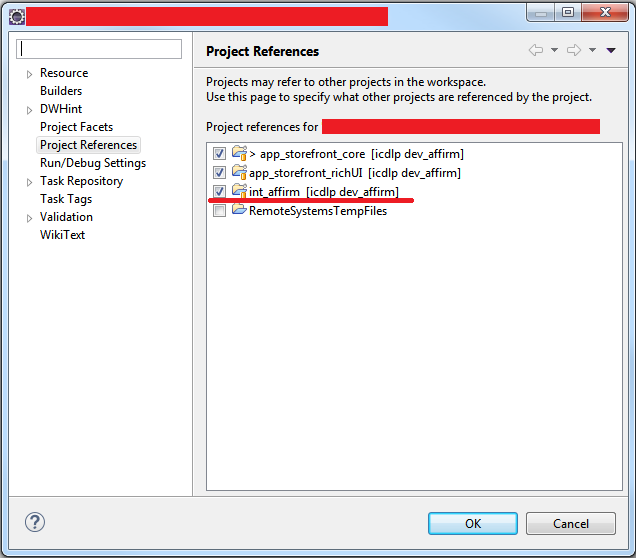
Section describes steps that should be completed before cartridge configuration in Business Manager.

### Deploying cartridge to a sandbox

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



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



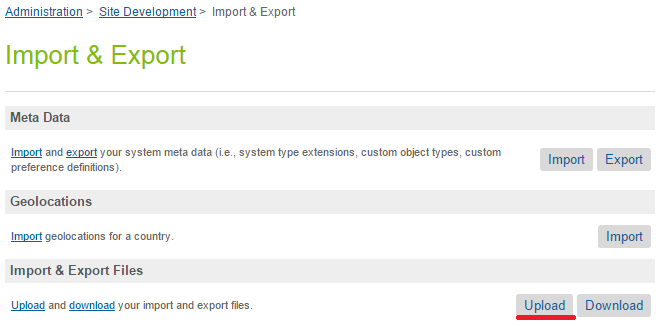
1. Wait until Studio completes workspace built and uploading source codes to a sandbox.

### Sandbox setup

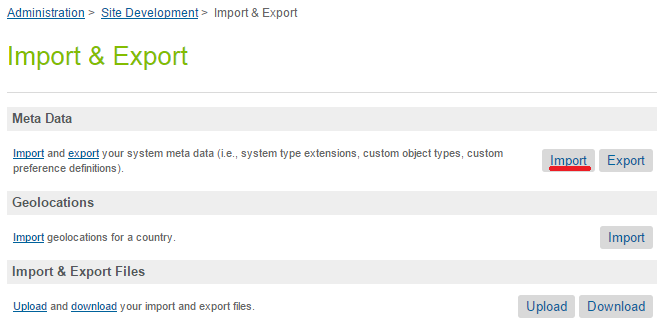
1. Go to Business Manager -> Site -> Manage Sites. Select correct site, then select Settings tab. In cartridge path at the beginning write the following:

int\_affirm\_overlay:int\_affirm\_sfra:

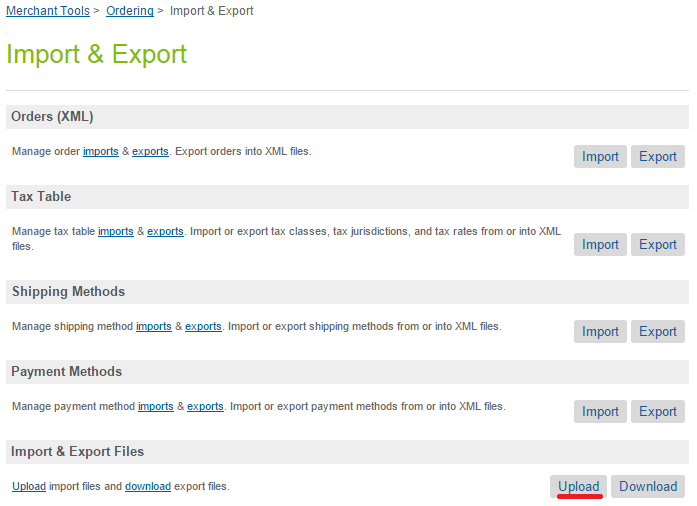
1. Download cartridge source code.
2. Find affirm\_metadata.xml in folded metadata.
3. Go to Administration > Site Development > Import & Export. Click Upload button and select affirm\_meta.xml.



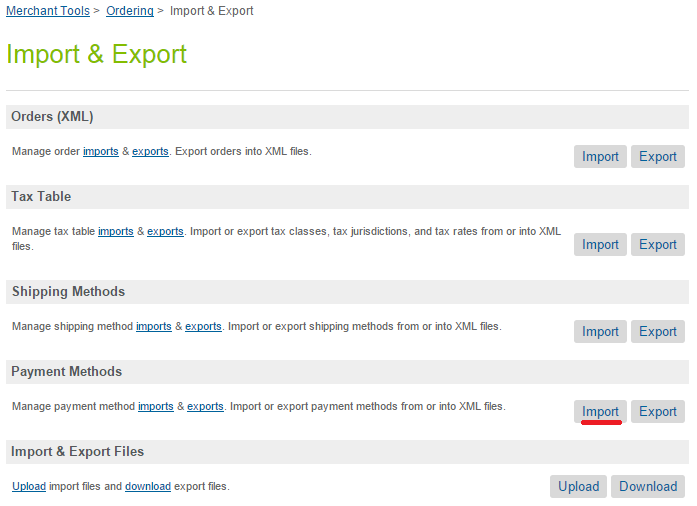
1. Click Import button and select affirm\_meta.xml.



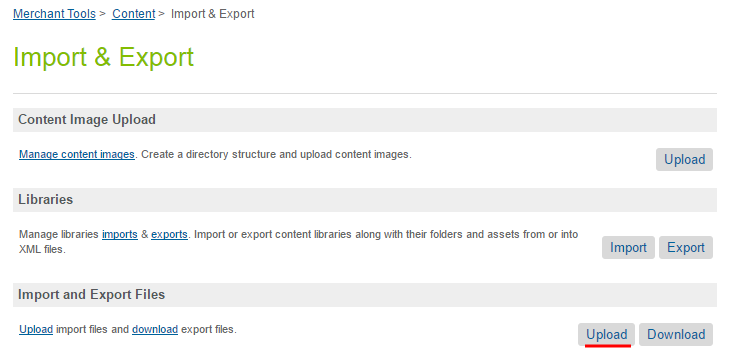
1. Go to Merchant Tools > Ordering > Import & Export. Click Upload button and select affirm\_payment.xml.



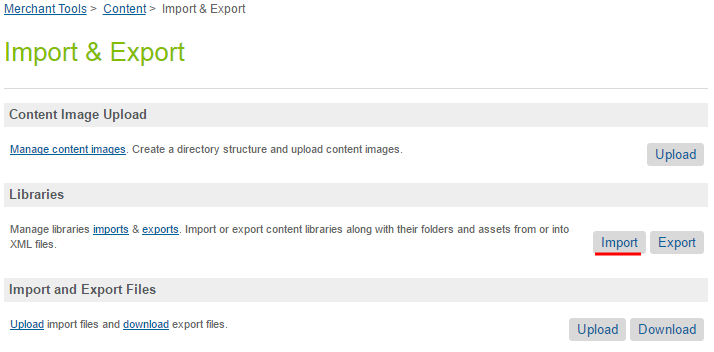
1. Click Import button and select affirm\_payment.xml.



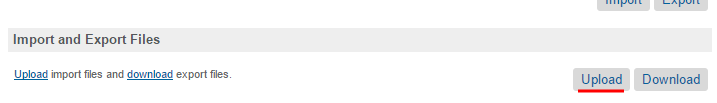
1. Go to Merchant Tools > Content > Import & Export. Click Upload button and select affirm\_asset.xml.



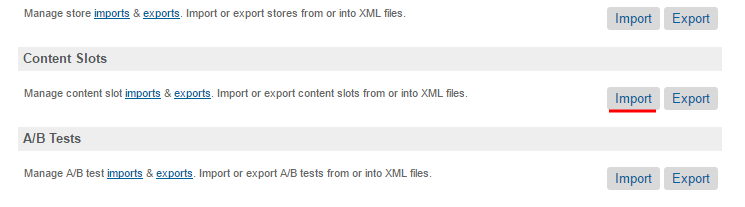
1. Click Import button and select affirm\_asset.xml.



1. Go to Merchant Tools > Online Marketing > Import & Export. Click Upload button and select affirm\_slots.xml.



1. Click Import button and select affirm\_slots.xml.



1. Go to [Merchant Tools](https://astound17.alliance-prtnr-eu01.dw.demandware.net/on/demandware.store/Sites-Site/default/ViewApplication-SelectSite?MenuGroupID=ChannelMenu&ChannelID=cdDHIiaag164kaaadmbwQoeF7W&SelectedSiteID=cdDHIiaag164kaaadmbwQoeF7W) > [Ordering](https://astound17.alliance-prtnr-eu01.dw.demandware.net/on/demandware.store/Sites-Site/default/SiteNavigationBar-ShowMenuitemOverview?CurrentMenuItemId=orders) > Payment Processors. Click NEW button and create payment processor with ID - AFFIRM\_PAYMENT.
2. Verify that Site Preferences group was created with ID - AFFIRM\_PAYMENT and name – Affirm.
3. Verify that Payment Method was created with ID - Affirm and name – Affirm. Enable Affirm payment method.
4. You can add image or text for your payment method by updating content asset “affirm-payment-method”.

## Configuration

This section describes configuration of the sandbox.

### Site Preferences configuration

1. Go to Merchant Tools > Site Preferences > [Custom Site Preferences](https://astound17.alliance-prtnr-eu01.dw.demandware.net/on/demandware.store/Sites-Site/default/CustomPreferences-View?PreferenceType=SITE) > Affirm
2. Enable attribute - Affirm Online Status. This attribute defines status (enable/disable) of Affirm integration.
3. Select a value from dropdown Affirm Mode. This attribute defines in which mode cartridge will work. Allowable values are “Sandbox” and “Production”.
4. Add site preference attribute - Affirm Finance Key with provided finance key from Affirm.
5. Add site preference attribute - Affirm Public Key with provided public key from Affirm.
6. Add site preference attribute - Affirm Private Key with provided private key from Affirm.
7. Select a value from Affirm Payment Action dropdown. This attribute defines which Affirm payment actions will be used in the checkout process.
8. Enable attribute - Cart Promo Message. This attribute defines status (enable/disable) of Promotional messages on Cart Page.
9. Enable attribute - Product Promo Message. This attribute defines status (enable/disable) of Promotional messages on Product Detail Page.
10. Enable attribute - PLP Promo Message. This attribute defines status (enable/disable) of Promotional messages on Product Listing Page.
11. Disable Affirm Payment Method – This attributes disables Affirm as a payment method if Affirm checkout is enabled using another payment gateway. Promotional Messaging is still enabled for cart, product and PLP pages.
12. 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 promotional message.
13. 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).
14. Add site preference value - Affirm Financing programs mapping. It allows to map financing programs to the 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. Please use the following format: {customerGroup}|{financingProgram},  
   Example: Registered|registeredFinProgram



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



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



1. Add site preference attribute - Payment Min Total with minimal cart total when Affirm payment method will be available.
2. Add site preference attribute - Payment Max Total with maximal cart total when Affirm payment method will be available.
3. Checkout Flow – Enable Redirect or Modal checkout flow
4. Enable attribute - Affirm Virtual Card Integration.
5. Add site preference attribute - VCN Payment Instrument with last part of hook name (see External Interfaces section of this document).

### Affirm Services configuration

1. Go to Administration > Operations > Import & Export.
2. Click Upload button and select affirm\_services.xml



1. Click Import button and select affirm\_services.xml.

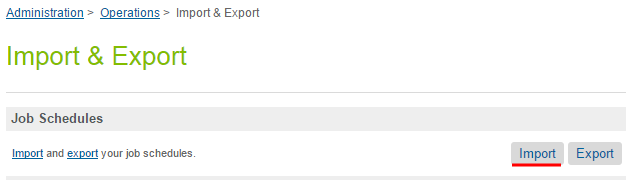


### Affirm Job configuration

1. Go to Administration > Operations > Import & Export.
2. Click Upload button and select affirm\_jobs\_sfra.xml.



1. Click Import button and run import of affirm\_jobs\_sfra.xml.



1. Go to Administration > Operations > Job Schedules
2. Verify that each imported job was created.



1. Change execution scope of every job to your site.

## Custom Code (SFRA)

This section describes changes that should be made to a merchant custom cartridge. All of this functionality is implemented in int\_affirm\_sfra and int\_affirm\_overlay cartridges for default Storefront Reference Architecture so you can find exact code there

1. Find and open the below template /comon/htmlHead.isml. Find the following code.



Paste code after:

<isinclude template=*"affirm/affirmheader\_mf"* />

1. Find and open template –/common/scripts.isml. Paste code at the top of the file:

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

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

1. Find and open template –/cart/cart.isml. Paste following code in section where you want to see Affirm promo message section. If it should be below cart then you need to paste code after <isinclude template="cart/cartApproachingDiscount" /> line:

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

<isinclude template="util/affirmmodule\_mf"/>

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

1. Paste following code in place where Affirm promo message should be shown on product details page. You need to modify setDetails.isml, productDetails.isml, bundleDetails.isml in product folder:

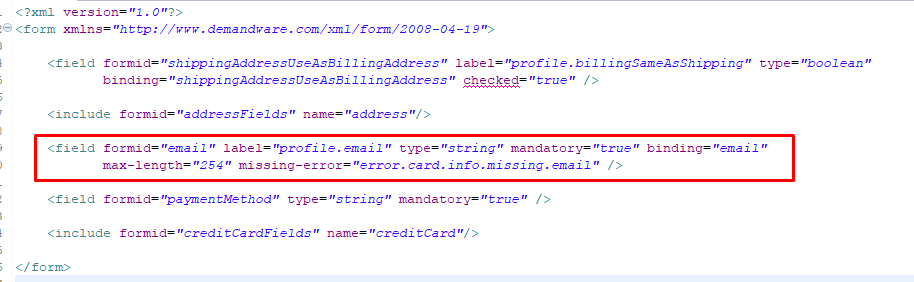
<isinclude template="util/affirmmodule\_mf"/>

<isaffirmpromo

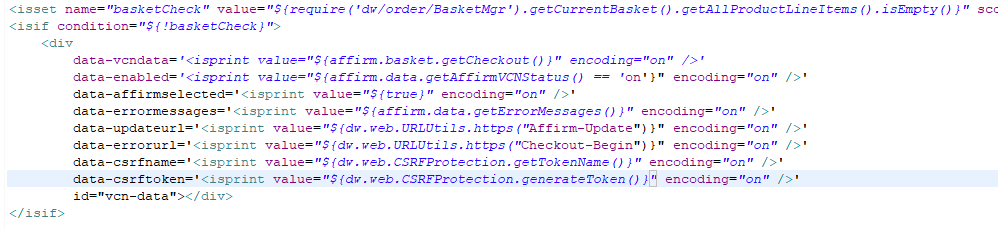
context="pdp" fpname="${require('int\_affirm\_sfra/cartridge/scripts/utils/affirmUtils').getFPNameForPDP(product)}" >

1. Paste following code in place where Affirm promo message should be shown on product listing page. You need to modify product/productTile.isml  
   <isinclude template="util/affirmmodule\_mf"/>

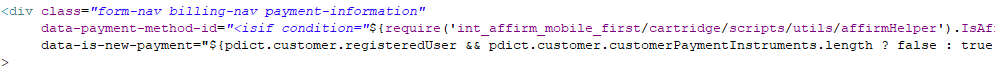
<isaffirmpromo context="plp" fpname="${require('int\_affirm\_sfra/cartridge/scripts/utils/affirmUtils').getFPNameForPLP(product)}" price="${price}"/>

1. Since credit card is hardcoded in Storefront Reference Architecture you need manually modify templates for enable Affirm payment. Following steps was performed in overlay cartridge:  
   Forms folder:  
   billing.xml - added email field in billing level  
   

creditCard.xml - removed email field  
Templates folder:  
checkout/billing/billing.isml - added email field HTML  

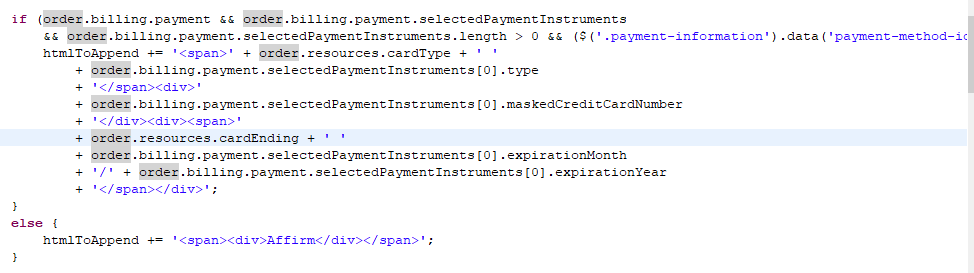

checkout/billing/billingSummary.isml - added div with vcn data  


checkout/billing/creditCardForm.isml - removed email field from credit card HTML

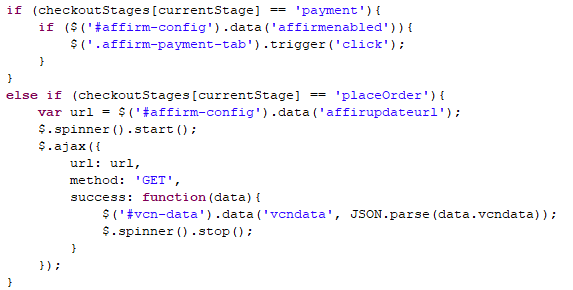
checkout/billing/paymentOption.isml - added condition for switching payment methods if Affirm is not applicable  
  
Controllers:

CheckoutServices.js was overridden for handling both Affirm and Credit card payments

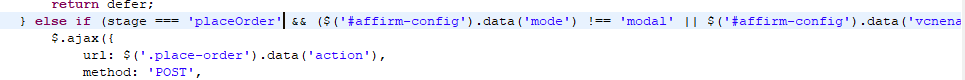
Client JS code:

js/checkout/billing.js - was added holder for updating payment information  


added checks if credit card fields are absent   

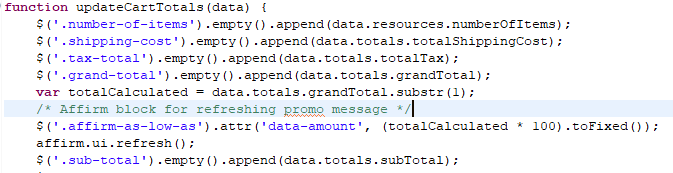

js/checkout/checkout.js  
added checking is affirm object is formed properly  


added insertion of Modal mode to checkout steps



js/cart/cart.js

added recalculation for Affirm Modal Promo price



1. Find and open template:

/components/modules.isml.

Paste code after the last line:

<isinclude template=*"util/affirmmodule\_mf"*/>

Don’t forget to rebuild your app.js after changing JS-files

## External Interfaces

Commutation between Affirm service and Salesforce Commerce Cloud Platform is based on Affirm REST API. All outside traffic from Salesforce Commerce Cloud instance is handled by HTTPS protocol.

In the Salesforce Commerce Cloud Platform each API call to Affirm is wrapped into Service that is handling monitoring and logging functionality.

Affirm integration documentation -<https://docs.affirm.com/Integrate_Affirm/Direct_API>

For as low as - <https://docs.affirm.com/Integrate_Affirm/Promotional_Messaging>

Virtual card integration assumes processing credit card using customer’s payment processors. Initially, it is BASIC\_CREDIT processor but if you need to integrate Affirm to site with other payment processors, do the next steps:

* Set site preference attribute “VCN Payment Instrument” with the last part of hook name, e.g. “CYBERSOURCE\_CREDIT\_CARD” (“BASIC\_CREDIT” by default);
* Register new hook handler for extension point “dw.int\_affirm.payment\_instrument.%PI%”, where %PI% is payment instrument value you’ve set in previous step.   
  For example “dw.int\_affirm.payment\_instrument.CYBERSOURCE\_CREDIT\_CARD”. By default it is “dw.int\_affirm.payment\_instrument.BASIC\_CREDIT”, defined in the file “int\_affirm\_sfra/cartridge/scripts/hooks.json”;
* Implement a method “add” of this hook handler. It will take basket as parameter. The method should return newly created payment instrument. Credit card info can be extracted from HttpParametersMap. Example of implementation you can find in “int\_affirm\_sfra/cartridge/scripts/payment/instrument/BASIC\_CREDIT.js”.

## Testing

Test scenarios and test cases are included in document Affirm Test Cases.docx

# Operations, Maintenance

## Data Storage

Affirm cartridge is extending Salesforce Commerce Cloud system objects to store related Affirm data form request. Following objects that were extended: Order, Product, Category, SitePreference.

## Support

In case of any availability issues, Affirm support can be reached via email at [merchanthelp@affirm.com](mailto:merchanthelp@affirm.com).

# User Guide

## Roles, Responsibilities

There are no recurring tasks that need to be fulfilled to set up the Affirm cartridge. As long as the configuration details are correct in your Site Preferences based on instructions in section 3.1 3.2 and 3.3 that is all that needs to be done.

## Business Manager

The cartridge doesn’t extend Business Manager.

## Storefront Functionality

Affirm cartridge is an implementation of Affirm payment method. Affirm is represented as a default payment method in Salesforce Commerce Cloud platform. Such implementation provides native support of Payment method.

# Known Issues

|  |  |
| --- | --- |
| **#** | **Description** |

# Release History

|  |  |  |
| --- | --- | --- |
| **Version** | **Date** | **Changes** |
| 15.1.0 | 07/03/2015 | Initial release |
| 17.1.0 | 02/01/2017 | * Controllers support * Enhancements and bug fixes * Added VCN support * Promotional Messaging support |
| 17.1.1 | 04/12/2017 | * Bug fixes for certification |
| 17.1.2 | 08/11/2017 | * Fix for Inconsistent Payment display at checkout after editing payment method |
| 17.1.3 | 09/18/2017 | * Ability to disable Affirm as a payment method |
| 18.1.0 | 05/27/2018 | * SFRA support |

1. e.g. <int\_affirm> [↑](#footnote-ref-1)