

**Clyde LINK Integration**

****

***Version 22.1.0***

****

Table of Contents

[1. Overview 3](#_Toc43714234)

[2. Summary 3](#_Toc43714235)

[3. Component Overview 3](#_Toc43714236)

[**2.1 Limitations, Constraints** 3](#_Toc43714237)

[**2.2 Compatibility** 3](#_Toc43714238)

[**2.3 Privacy, Payment** 3](#_Toc43714239)

[3. Implementation Guide 4](#_Toc43714240)

[**3.1 Adding the cartridge in SFCC UX Studio** 4](#_Toc43714241)

[**3.2 Configuration** 4](#_Toc43714242)

[3.2.1 Importing Meta Data 4](#_Toc43714243)

[3.2.2 Setting Clyde Custom Site Preferences 5](#_Toc43714244)

[3.2.3 Jobs List 6](#_Toc43714245)

[3.2.4 Custom overriding changes for reference 7](#_Toc43714246)

[4 Testing 10](#_Toc43714247)

[5 Operations, Maintenance 13](#_Toc43714248)

[**5.1 Data Storage** 13](#_Toc43714249)

[**5.2 Support** 14](#_Toc43714250)

[6 User Guide 14](#_Toc43714251)

[**6.1 Roles, Responsibilities** 14](#_Toc43714252)

[**6.2 Business Manager** 14](#_Toc43714253)

[7 Known Issues 15](#_Toc43714254)

[8 Release History 15](#_Toc43714255)

# **Overview**

This document describes the technical integration of Clyde link cartridge.

Clyde’s platform is a robust tool for business. By offering Clyde’s product protection plans, B2C marketers ensure customer lifetime value, increase margins, and align with top-tier insurance companies instantly. Clyde created an unbelievable insurance experience for business and end-customers.

**1.1 Functional Overview**

1. Clyde provides warranty plans to the products that are purchased.  
 2. Customers can easily select the product protection of their choice – during their purchase.  
  
 The Clyde – Salesforce cartridge displays the contracts associated with the products.

 i. This cartridge displays the contracts of corresponding products on Product details page.

ii. This cartridge adds each contract as a ProductOption on the product line item and the contract ProductOption clearly describes which product it is associated with in the cart.

**1.2 Regionality**

The Clyde cartridge can be used to purchase products with Clyde warranty from United States regions.

# **Summary**

This implementation guide describes how to integrate Clyde (int\_clyde\_sfra) into SFCC reference application SFRA site v6.0.0.

This cartridge provides the ability to use Clyde contracts for a product during checkout of the SFCC(B2C Commerce) storefront.

To integrate you will need to modify RefArch storefront cartridge.

# **Component Overview**

This cartridge offers Clyde’s product protection plans.

## **2.1 Limitations, Constraints**

There is no known limitation and constraints.

## **2.2 Compatibility**

Compatibility mode: 19.10

## **2.3 Privacy, Payment**

No credit card details are stored within SFCC using this integration

# Implementation Guide

## 

## **3.1 Adding the cartridge in SFCC UX Studio**

To upload the cartridges into the SFCC server, you first need to add the cartridges in SFCC UX Studio. In order to do this, follow these instructions.

1. In UX Studio, select in the menu File > Import.
2. In the import dialog, select General > Existing Projects in the workspace and click on 'Next'.
3. Ensure 'Select archive file' is selected and select the compressed cartridge file by clicking on the 'Browse' button.
4. Click on 'Finish' button to import the cartridges.
5. Studio will now ask you if you want to link the cartridge to your active DigitalServer connection. Click on yes or manually link the cartridge to your server by checking the project under project references in the server connection properties.

**Activating the cartridges in Business Manager**

Before the Clyde functionality can become available to SFRA, the cartridges have to be added to the cartridge path of the Site in question. In order to do this, follow the following instructions:

1. Log into Business Manager

2. Navigate to Administration > Sites > Manage Sites.

3. Click on the target site name and Click on Settings tab.

4. In the Cartridges prepend the cartridge name “**clyde\_sfra\_changes:int\_clyde\_sfra:int\_clyde:bm\_clyde**” separated by a colon(:) and click on Apply button.

5. Add “**bm\_clyde**” in Business Manager site. Refer to “**6.2 Business Manager”** section.

## **3.2 Configuration**

This chapter will describe how to configure the cartridge in Business Manager for SFRA.

### 3.2.1 Importing Meta Data

For the Clyde integration to work, the following object structures (metadata) needs to be imported and configured in the Business manager. Follow the below steps:

1. In the cartridge bundle find metadata/clyde-meta-import folder.
   * 1. Find the sites folder and inside it find RefArch folder.
     2. Rename this RefArch folder to the ID of your site. If you have multiple site to which these meta has to be imported, then copy and paste this RefArch folder to replicate for other sites. Rename those folders with ID of corresponding sites.
2. In the cartridge bundle, inside metadata folder compress clyde-meta-import folder to generate clyde-meta-import.zip file.
3. Go to Business Manager Menu > Administration > Site Development > Site Import & Export
4. Under Import: Upload Archive: Ensure that the radio button with label Local is enabled (Else click on the radio button to enable it)
5. Click on Choose File input field, select the clyde-meta-import.zip file from open dialog box and click on upload button
6. After finishing the upload, from the Archives list click the radio button corresponding to clyde-meta-import.zip and click on Import button
7. Click on OK button of the confirmation box asking, “Are you sure that you want to import the selected archive?”

### 3.2.2 Setting Clyde Custom Site Preferences

In Business Manager, navigate to the target Site > Site Preferences > Custom Preferences.

A custom site preference group with the ID Clyde is available. Please select it and edit the attributes accordingly.

**GroupID** – **Clyde**: Set below fields

* **API Key**: Clyde API key.
* **Signing Secret**: Clyde API signing secret.
* **Send only Clyde contract Orders**: Send only orders which has Clyde contract as one of the product line item. Set the value in this field to enable/disable. If enabled orders with Clyde contract are sent to Clyde else if disabled all the orders are sent to Clyde.
* **Clyde Widget Environment**: Clyde environment to be mentioned for widget display.
* **Clyde Widget Type**: Clyde widget view type on the page.
* **Clyde Widget Display**: Set the value in this field to enable/disable. If enabled Clyde contract widget will be displayed on the product detail page else it will not be displayed.
* **Send only Clyde contract orders:** Send only orders that have a Clyde contract associated with one or more product line items.
* **Clyde Widget Display on Cart:** A Boolean site preference which is used to enable or disable the Clyde widgets on cart.
* **Skip GeoIP Check:** A Boolean site preference that if enabled, the widget will skip the geofencing check.

**GroupID** – **Clyde Job Configuration**: Set below fields

* **Clyde Send Full Product Last Sync Time**: The last time the full product job ran.
* **Clyde Send New Product Last Sync Time**: The last time the new product job ran.
* **Clyde Send Delta Product Last Sync Time**: The last time the delta product job ran.
* **Clyde Send Order Last Sync Time**: Set last time in which Send Orderjob run.
* **Clyde Cancel Order Last Sync Time**: Set last time in which Cancel Orderrun.

**GroupID** – **Clyde Contract SKU Configuration**: Set below fields

* **Clyde Master Catalog ID:** The ID of the master catalog that holds your store’s product data. Needed for ClydeContractImport job.
* **Clyde Contract Tax Class ID:** Tax class ID for Clyde contract options.

### 3.2.3 Jobs List

Go to **Administration > Operations > Jobs**

1. **ClydeCreateOrder:**

Job for exporting orders from SFCC to Clyde dashboard.

This job must be assigned to any one of the available sites. This is because orders are created for the sites selected in the BM.

**Job step**:

job-workflow-step-create-order: This step will export created orders from SFCC to Clyde.

**Job parameter**:  
 Starting Order Number: If provided then the job will start sending the orders starting from this order number.

1. **ClydeCancelOrder:**

Job for exporting orders from SFCC to Clyde dashboard.

This job must be assigned to any one of the available sites. This is because orders are created for the sites selected in the BM.

**Job Step**:

job-workflow-step-cancel-order: This step will export orders from SFCC to Clyde for cancellation.

**Job parameter**:  
 Starting Order Number: If provided then the job will start sending the orders starting from this order number.

1. **ClydeProductFullExport:**

Job for exporting all products from SFCC to Clyde dashboard.

This job must be assigned to any one of the available sites. This is because products are contained by the catalogs and these catalogs are assigned to the site selected in the BM.

**Job Step**:

job-workflow-step-bulk-product: This step will export all the products of the catalog from SFCC to Clyde.

1. **ClydeProductDeltaExport:**

Job for exporting last modified and the newly created products from SFCC to Clyde dashboard.

This job must be assigned to any one of the available sites. This is because products are contained by the catalogs and these catalogs are assigned to the site selected in the BM.

**Job Step**:

Reindex: This step will reindex the products

job-workflow-step-delta-product: This step will export last updated and the newly created products of the catalog from SFCC to Clyde.

1. **ClydeContractImport:**

Job for importing Clyde contracts from Clyde dashboard to SFCC as an option product.

This job must be assigned to any one of the available sites. This is because products are contained by the catalogs and these catalogs are assigned to the site selected in the BM. You also need to add your storefront catalog id inside customer preference for this job.

**Job Step**:

ClydeContractImport: This step will export all option products related to each product in BM according to its assign catalog.

ImportCatalog: This step will export all option products related to each product in BM according to its assign catalog.

*Note: Please keep the import mode as merge otherwise it will replace our existing catalog*

### 3.2.4 Custom overriding changes for reference

Following are the custom code changes in overriding files for reference:



* File: {custom storefront cartridge}/cartridge/client/default/js/product/base.js after line number 652 add the below line.

**form = getSelectedClydeContract(form);**

****

* File: {custom storefront cartridge}/ cartridge/client/default/js/product/detail.js at line number 331 5 to 14 and at line number 66 add the below lines.

var base = require('./base');

var clydeWidget = require('../getClydeWidget.js');

/\*\*

\* @description Get clyde widget on variant change

\* @param {Object} response - response object

\*/

Function getClydeVariantChange(response) {

clydeWidget.getClydeVariantChange(response.data.product.id);

}

At line number 66  
 getClydeVariantChange(response);





**Overriding templates**

Following files are overriding templates on top your storefront custom cartridge. Actual custom changes in those files are mentioned below, so you can reuse your customized storefront templates with following custom changes in it.



**Template 1**: **clyde\_sfra\_changes/cartridge/templates/default/product/productDetails.isml**

<script type="text/javascript"><isinclude template="resources/appresources"/></script>

<isif condition = "${dw.system.Site.current.preferences.custom.clydeWidgetDisplay}">

<div id="clyde-cta" >

</div>

</isif>

**Template 2: clyde\_sfra\_changes/cartridge/templates/default/product/components/options.isml**

<isset name="product" value="${typeof product === 'undefined' || empty(product) ? pdict.product : product}" scope="page" />

<isloop items="${product.options}" var="option" status="loopStatus">

<isif condition="${!(option.id == "clydeWarranty")}">

<div class="row product-option" data-option-id="${option.id}">

<div class="col-8">

<isinclude template="product/components/option" />

</div>

</div>

</isif>

</isloop>



**Template 3:**  **clyde\_sfra\_changes/cartridge/templates/default/common/scripts.isml**

<isif condition="${dw.system.Site.current.preferences.custom.clydeWidgetsEnabled}">

<script src="${dw.system.Site.current.preferences.custom.clydeWidgetJs}"></script>

<script src="${URLUtils.staticURL('/js/customClydeWidget.js')}" ></script>

</isif>



**Template 4: clyde\_sfra\_changes/cartridge/templates/default/cart/productCard/cartProductCard.isml**

<isloop items="${lineItem.options}" var="option">

<isif condition="${!!option.optionId && !(option.optionId == 'clydeWarranty')}" />

<div class="lineItem-options-values" data-option-id="${option.optionId}" data-value-id="${option.selectedValueId}">

<p class="line-item-attributes">${option.displayName}</p>

</div>

</isif>

</isloop>

**Line no 59 to 61**

<div class="hidden-md-down">

<isinclude template="components/deleteClydeProductButton" />

</div>



**Template 5: clyde\_sfra\_changes/cartridge/templates/default/cart/productCard/cartProductCardProductPrice.isml**

**Template 6: clyde\_sfra\_changes/cartridge/templates/default/cart/cart.isml**

<isif condition="${dw.system.Site.current.preferences.custom.clydeWidgetDisplayCart}">

<script src="${URLUtils.staticURL('/js/clydeCartWidget.js')}" ></script>

</isif>

<isif condition="${dw.system.Site.current.preferences.custom.clydeWidgetDisplayCart && !clydeOption}">

<isset name="clydeProductId" value="${lineItem.id}" scope="page"/>

<isset name="clydeProductLineItemUUID" value="${bonusproductlineitem + "uuid-" + lineItem.UUID}" scope="page" />

<isinclude template="clyde/widgets/clydeCartWidgets" />

</isif>

**Line no 96**

<isinclude template="cart/cartRemoveClydeProductModal"/>

**Template 7: clyde\_sfra\_changes/cartridge/templates/default/checkout/productCard/productCard.isml**

<div class="item-options">

<isloop items="${lineItem.options}" var="option">

<isif condition="${!empty(option.displayName) && !(option.optionId === 'clydeWarranty')}">

<div class="line-item-option">${option.displayName}</div>

</isif>

</isloop>

</div>

<isloop items="${lineItem.options}" var="option">

<isif condition="${!empty(option.displayName) && option.optionId === 'clydeWarranty'}">

<div class="product-line-item-details d-flex flex-row">

<div class="item-image">

<img class="product-image" src="${option.imageURL}" alt="${option.displayName}" title="${option.displayName}">

</div>

<div class="item-attributes">

<p class="line-item-attributes">${option.lineItemText}</p>

<p class="line-item-attributes clyde-option-price" data-option-id="${option.optionId}" data-value-id="${option.selectedValueId}" data-uuid="${lineItem.UUID}">${option.price}</p>

</div>

</div>

</isif>

</isloop>

**Template 8: clyde\_sfra\_changes/cartridge/templates/default/checkout/productCard/productShippingCard.isml**

<div class="item-options">

<isloop items="${lineItem.options}" var="option">

<isif condition="${!empty(option.displayName)}">

<div class="line-item-option">${option.displayName}</div>

</isif>

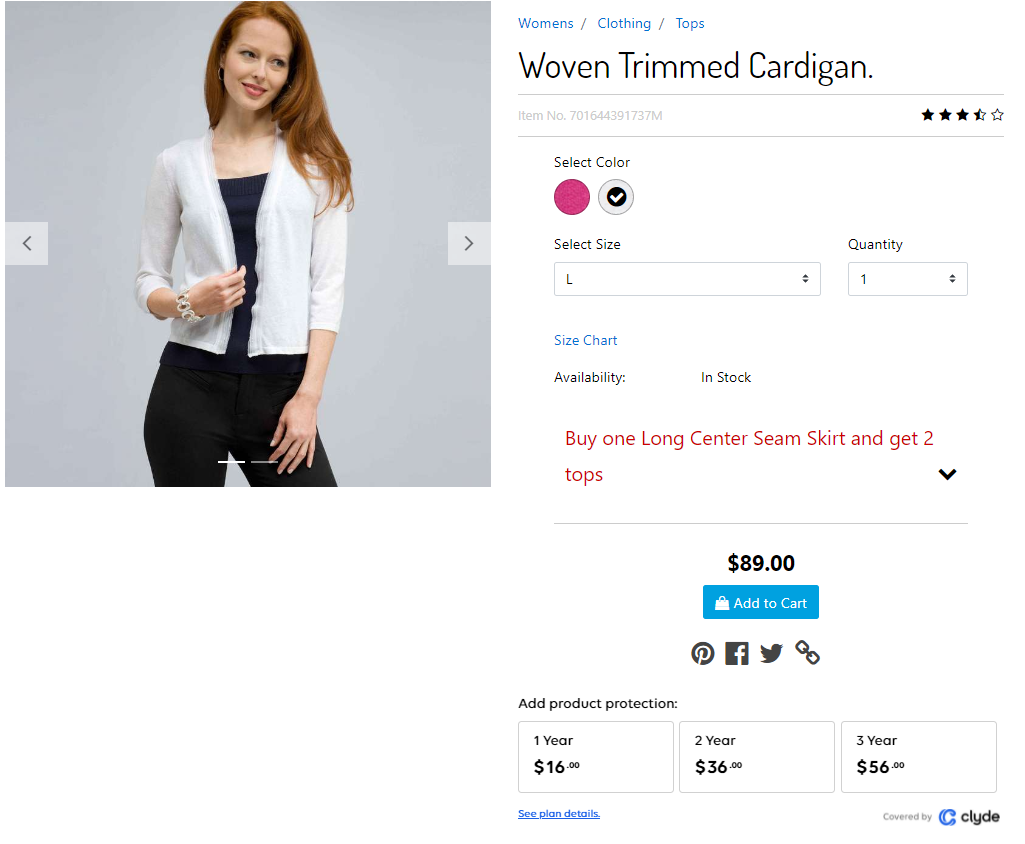
</isloop>

</div>

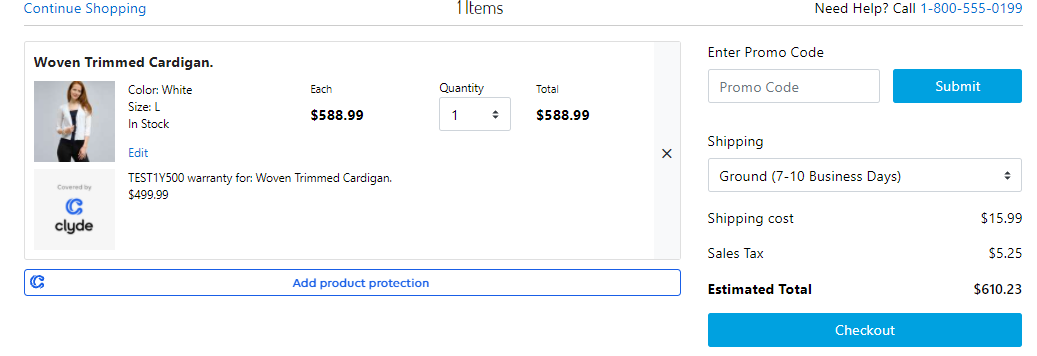
# Testing

In Business Manager, navigate to the target Site > Site Preferences > Custom Preferences. A custom site preference group with the ID Clyde is available. Please select it and add the respective data (obtained from Clyde).

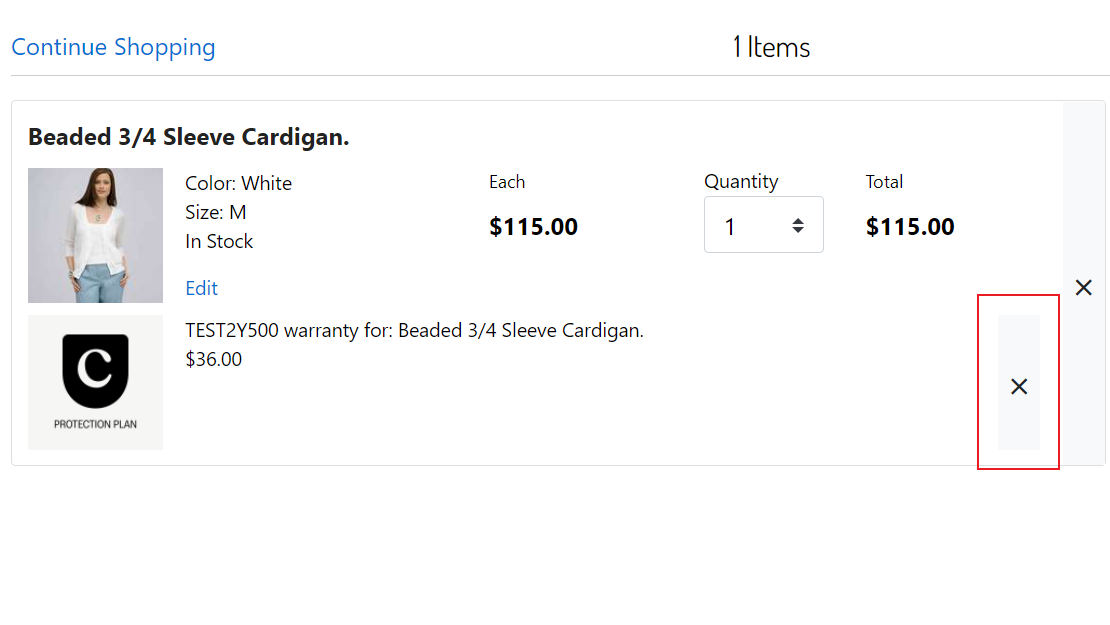
**Products Details Page:**



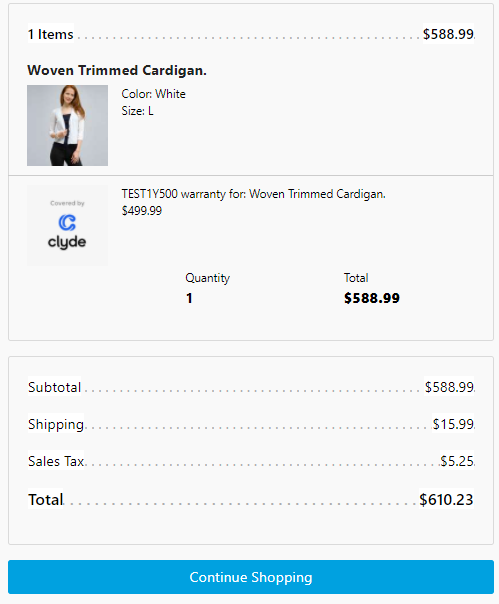
**Cart Page :**



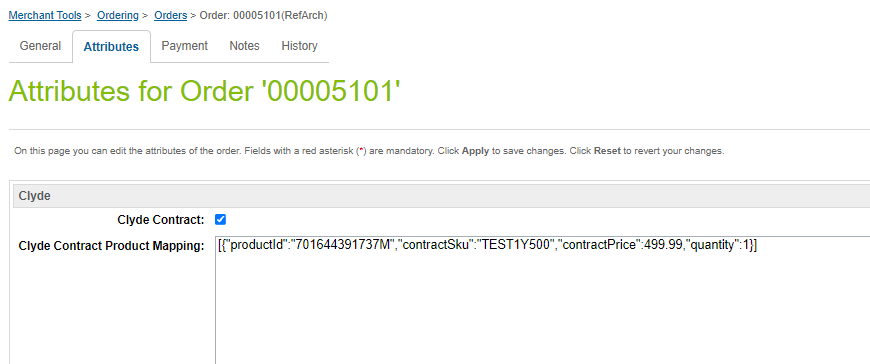
**Remove Clyde Product from Cart:**



**Checkout Page:**



**Order verification (Business Manager):**

After successful placement of order search in Business Manager with the created order number.  
GoTo [Merchant Tools](https://integocodbaclyde-tech-prtnr-na01-dw.demandware.net/on/demandware.store/Sites-Site/default/ViewApplication-SelectSite?MenuGroupID=ChannelMenu&ChannelID=b25a97823877ab1263d50a3bb1&SelectedSiteID=b25a97823877ab1263d50a3bb1&csrf_token=mLNZ83KJmPDgW23ecNLiVB_HwAj3clCbqOy1LDz8h83pHauiK5eua8bar-7ecfrWTv7YO56WSHgEiXjULHXLfpGoAVWdCpAKjqMDDJpaTT9t_Xhg7LHhhyrDmN3uFSgJNw_bqhRE2OSQBfQekyH8YhYaNX_bDm0ndvl4c69yAeVwlvOjbQc=) >  [Ordering](https://integocodbaclyde-tech-prtnr-na01-dw.demandware.net/on/demandware.store/Sites-Site/default/SiteNavigationBar-ShowMenuitemOverview?CurrentMenuItemId=orders&csrf_token=mLNZ83KJmPDgW23ecNLiVB_HwAj3clCbqOy1LDz8h83pHauiK5eua8bar-7ecfrWTv7YO56WSHgEiXjULHXLfpGoAVWdCpAKjqMDDJpaTT9t_Xhg7LHhhyrDmN3uFSgJNw_bqhRE2OSQBfQekyH8YhYaNX_bDm0ndvl4c69yAeVwlvOjbQc=) >  Orders. Search for the order number in the search box and click on the order and check for the Product and Clyde contract mapping under Attributes tab of Order object as show in the screenshot below.  
  


**Clyde Service:** When the service is unavailable, Clyde API will provide the industry wide and expected HTTP 500 error response. And this case will not break the page, however user will not able to see Clyde contracts on the product detail page.

when service is unavailable for Clyde then connect to Clyde account manager.

# Operations, Maintenance

## **5.1 Data Storage**

Some additional custom attributes and custom objects are stored against system objects. The following table lists these attributes and their uses, other than Site Preferences. (which are listed under section 3.3.2).

|  |  |  |  |
| --- | --- | --- | --- |
| **Object** | **Attribute** | **Type** | **Description** |
| Basket | Clyde Contract Product List  (clydeContractProductList) | Text | ProductContractList |
| Order | Clyde Contract  (isContainClydeContract) | Boolean | Flags the order if it has contracts or not |
| Order | Clyde Contract Product Mapping  (clydeContractProductMapping) | Text | Product Contract Mapping |
| Site Preferences | Clyde Widget Display | Bolean | Set the value in this field to enable/disable. If enabled Clyde contract widget will be displayed on the product detail page else it will not be displayed. |
| Site Preferences | Clyde Widget Display On Cart | Bolean | It is bolean type site preference which is used to enable or disable the clyde widgets on cart. |
| Site  Preferences | Clyde Widget JS | String | Its render the default url of clyde widgets js |
| Site Preferences | Clyde Send full Product Last Sync Time. | Datae+Time | Set last time in When full product job run |
| Site Preferences | Clyde Send New Product Last Sync Time. | Datae+Time | Set last time in When Send New Product job run |
| Site Preferences | Clyde Send Delta Product Last Sync Time. | Datae+Time | Set last time in When Send Delta Product job run |
| Site Preferences | Clyde Send Order Last Sync Time. | Date+Time | Set last time in When Send Order job run |
| Site Preferences | Clyde Cancel Order Last Sync Time. | Datae+Time | Set last time in When Cancel Order job run |
| Site Preferences | Clyde Master Catalog ID | String | Its set Catalog ID for jobs |
| Site Preferences | Clyde Contract Tax Class ID | String | Its set Tax class id of clyde contracts |
| Site Preferences | Clyde Img URL | String | Its set default image of clyde widgets. |

## **5.2 Support**

# User Guide

## **6.1 Roles, Responsibilities**

Integration of this cartridge will typically be done by a SFCC (B2C Commerce) developer.

Clyde will provide access keys for be used with the API.

## **6.2 Business Manager**

For the Clyde integration to work for the jobs, the following needs to be configured in the sites of the Business manager.

1. Upload the cartridge bm\_clyde into server.
2. Go to B2C Commerce > **Administration > Manage Sites > Settings > Cartridges** input field append cartridge name “bm\_clyde‟.
3. Click on Apply button.
4. Repeat steps 2 and 3 for all sites including Business Manager site
5. Jobs list mentioned in 3.2.3.
6. **Administration > Operations > Jobs**.
7. To assign sites to a job, go to Job Steps tab in the Job, under “Scope:” select “Specific Sites” from the drop down and check the buttons beside sites names and click on “Assign” button.
8. To run Jobs manually click on “Run” button.
9. After successful job run check Clyde dashboard for the exported orders and products from SFCC to Clyde.
10. Repeat the above steps(6 to 8) for other jobs and check Clyde dashboard for Orders and Products data.

**Steps to Schedule a Job:**

Go to **Administration > Operations > Jobs**  
 1. Go to any one of the job, under Schedule and History  
 a. Check the box for Enabled.  
 b. Under “**Trigger**” dropdown select “**Recurring Interval**”  
 c. Fill the “From” and “To” fields with dates.

d. Under “**Interval**” dropdown select Minutes and Amount as required as

5(Minutes). Check below screenshot.

A screenshot of a cell phone

Description automatically generated

**Note**: For sandbox testing the service URL is <https://api.sb.joinclyde.com/> which is pointing to sandbox environment. You need to get it updated before going live with <https://api.stg.joinclyde.com/> live URL. You can change the URL from

Administration > Operations > Services > clyde.credentials

Please see more details here: <https://documentation.b2c.commercecloud.salesforce.com/DOC2/topic/com.demandware.dochelp/Jobs/CreatingJobs.html>

# Known Issues

No known issues at the time

# Release History

|  |  |  |
| --- | --- | --- |
| Version | Date | Changes |
| 20.1 | 16-Jun-2020 | Initial release |
|  |  |  |
| 22.1 | 06-January-2022 | Second Release |