int_klaviyo

Table of Contents

Summary

This documentation describes how to implement Klaviyo's cartridge and OCAPI integration into your Salesforce Commerce Cloud application. The Klaviyo integration features a self-contained cartridge that handles passing events in real-time from your website to Klaviyo, as well as integrations to OCAPI endpoints for product catalog and order data.

The cartridge includes preferences to be configured in the Business Manager and once enabled, contains all elements necessary to perform a successful best practices implementation of Klaviyo.

The OCAPI integration utilizes both "order_search" and "product_search" endpoints to sync catalog and order data.

Events tracked through cartridge (real-time):

- Active on Site
- Added to Cart
- Checkout Started
- Order Confirmation
- Site Search
- Viewed Category
- Viewed Product

Events tracked through OCAPI (periodic sync):

- Placed Order
- Ordered Product

Component Overview

Functional Overview

The Klaviyo cartridge provides integration for the following functionality:

- Integration of Klaviyo JavaScript library (referenced here: https://apidocs.klaviyo.com/reference/javascript-client-library)
- Integration of Klaviyo Server Side library (referenced here: https://apidocs.klaviyo.com/reference/track-identify) - Cartridge includes a controller to call the Klaviyo Track Server Side API. This is to fire order related events such as Order Confirmation to Klaviyo.
- Klaviyo JavaScript snippet integration and setup – Klaviyo tags can be fired in any page with the help of a modular ISML template and controller integration.
- Klaviyo Subscribe API (optional) – The cartridge also includes a controller to call the Klaviyo Lists API to check for the presence of and subscribe a user to a Klaviyo List. This can be used for functionality such as Newsletters and other forms.

Prerequisites

Before performing the steps in this document, ensure that you and your organization have satisfied the following criteria:

- You have reached out to Klaviyo and have set up a Klaviyo Account (where you can find your Klaviyo Organization ID and API Key, more details below).
- You have downloaded and unzipped the Klaviyo cartridge ZIP file from Salesforce Commerce Cloud Marketplace or your Klaviyo Consultant. The cartridge ZIP file contains the Klaviyo cartridge package residing in the folder int_klaviyo, the preferences configuration XML file, the services configuration XML file, and custom attribute definition file.

Compatibility, Reliability, Locales

The cartridge is designed for Salesforce Commerce Cloud API version 18.8+ for both Site Genesis (SG) and Storefront Reference Architecture (SFRA), but if you are on older versions please contact us for confirmation, as it should be supported.

If the Klaviyo service is down, events passed by the cartridge will not be recorded, logs will be created for "Order Confirmation" and "Added to Cart" events that use the KlaviyoTrackService, and there will be no performance impact on the Commerce Cloud website.

Klaviyo can support Localization in the form of multi-lingual data storage and ingest on a per-account basis. For example, if you wish to store marketing data in 5 languages and various currencies, you would require 5 Klaviyo accounts with the data for each locale syncing to each Klaviyo account. Klaviyo's administration interface is only available in English.
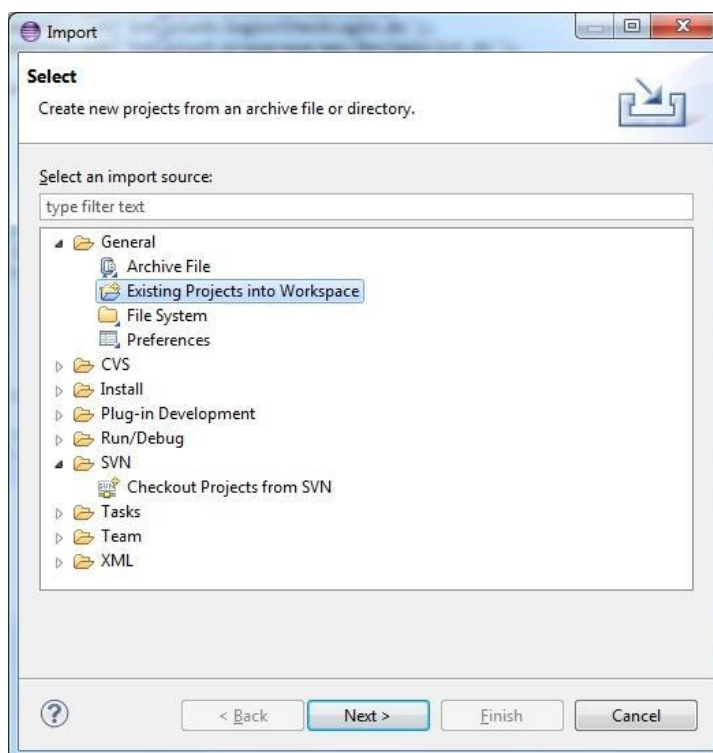
Privacy, Payment

Please see the Klaviyo Privacy Policy here - https://www.klaviyo.com/privacy

Contact your account manager for pricing.

Cartridge Implementation Guide

Setup

- Open Salesforce Commerce Cloud Studio.

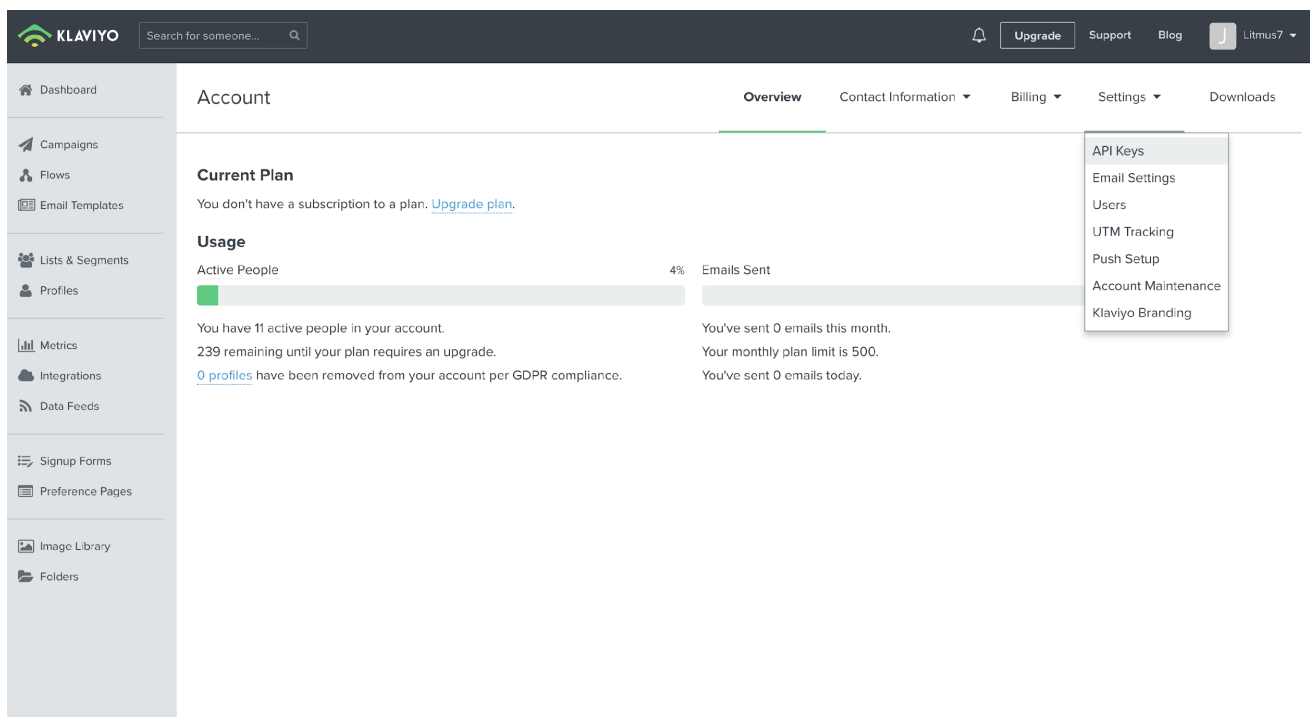- Import downloaded cartridge. Select on import screen **int_klaviyo**

- Link cartridge to the sandbox: Select the Salesforce Commerce Cloud connection and select Properties. Select Project References and check in **int_klaviyo**
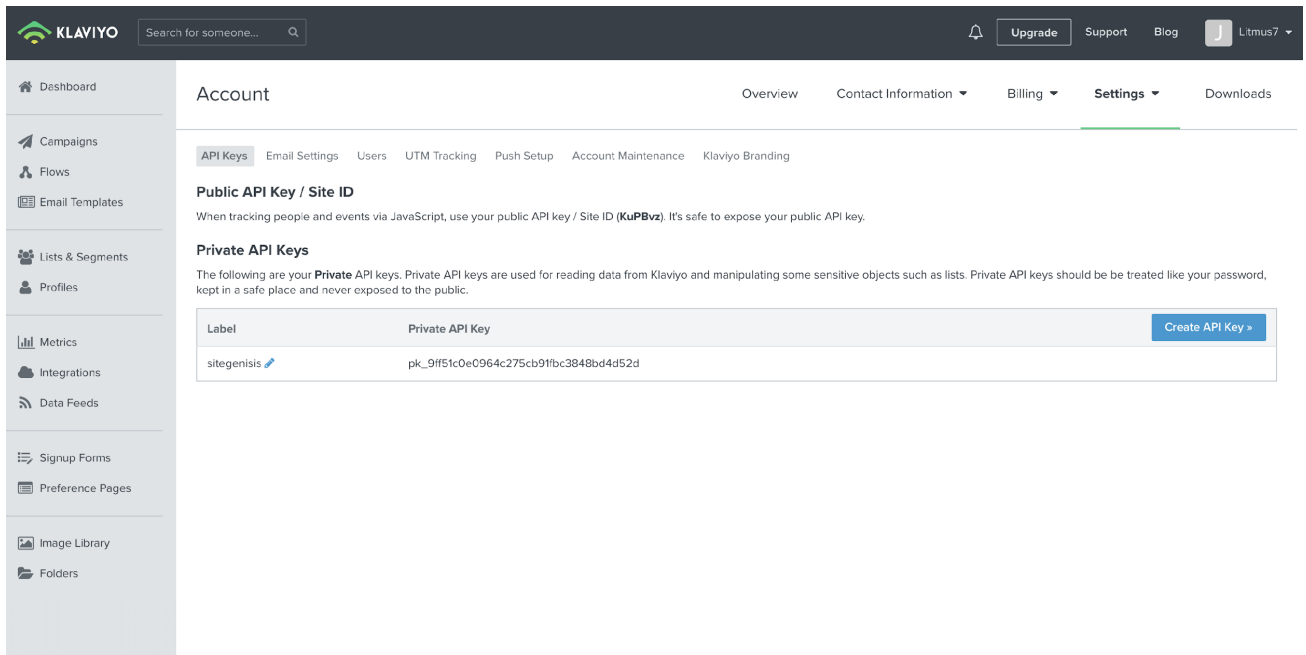
Klaviyo Configuration

Once you sign up for your Klaviyo account, you can get your Account ID and API key for setup from your Klaviyo account by following the below steps:

Navigate to the account section by selecting the drop down on the top right. Once in the account page, select the API Keys option under settings to reveal the Account ID and API key.

Business Manager Setup

- Once the cartridge is imported, it needs to be added to the list of cartridges in your site. Go to Business Manager: Administration -> Sites -> Manage Sites. Select your site, then select **Settings** tab. In cartridge path at the end, add **int_klaviyo.**

- Import the zipped services file that come with the cartridge (int_klaviyo\metadata\klaviyo_services_and_preferences.zip). This houses the service definitions for the Klaviyo track and lists server side API, and the custom preferences for Klaviyo. Go to Administration->Operations->Import and Export->Services. Upload and import the klaviyo-services.xml file. This will create one service (two others are optional for the Subscription Service and can be provided upon request):

  o KlaviyoTrackService

- After the metadata zip file import, create the Klaviyo preferences. This creates site level custom attributes for storing the Klaviyo account ID, image size, and API Key (optional). To import, navigate to Merchant Tools > Administration >Site Development->Import and Export. Under the meta data section, upload and import the custom_atribute_klaviyo.xml file. This creates the following attributes in Site Preferences:

  o Klaviyo Account – This is mandatory and used to authenticate to the Javascript API.

  o Klaviyo API Key – The private API key for the server side API.

  o Klaviyo Enabled – To Enable the Klaviyo Service

  o Image type for Klaviyo – The image size you use for products ('hi-res', 'large', etc.)

- Setup OCAPI access for Klaviyo. This access allows Klaviyo to periodically sync catalog and order data into your Klaviyo account.

  o Navigate to https://account.demandware.com/dw/account/APIAdmin and add an API client for Klaviyo. The API Client ID and password will be required to generate the bearer token for OCAPI.

  o Once the API client is added, navigate to Administration > Site Development > Open Commerce API Settings in the SFCC Business Manager.

  o Add the following snippets, replacing the API version and Client ID. We support API versions 19.5 and above as well as 18.8. Replace CLIENT_ID with the API Client ID generated from the API client setup in the previous step (this should look something like "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx").

  o Add the following JSON under the type Shop and context Global (Organization Wide), replacing SHOP_API_VERSION with your OCAPI Shop API version, then click Save.

```
{
  "_v": "SHOP_API_VERSION",
  "clients": [
   {
    "client_id": "CLIENT_ID",
    "resources": [
     {
      "resource_id": "/order_search",
      "methods": [
       "post"
      ],
      "read_attributes": "(**)",
      "write_attributes": "(**)"
     }
    ]
   }
  ]
}
```

  ▪ Once added, the settings should appear similar to this:

Open Commerce API Settings

This page allows you to make client application-specific configurations of Open Commerce API resources, i.e. manage resource access privileges, attribute read/write permissions, or configure client application-specific response headers. You can select the actual type of the API that you want to configure and specify whether your settings are site-specific or global (organization-wide). Please note that due to caching, changes may take up to three minutes to become effective.
You can browse the Open Commerce API here API Explorer.

Select Type: Shop ▾   Select Context: Global (organization-wide) ▾

```
{
  "_v":"19.10",
  "clients":[
    {
      "client_id":"xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
      "resources":[
        {
          "resource_id":"/order_search",
          "methods":["post"],
          "read_attributes":"(**)",
          "write_attributes":"(**)"
        }
      ]
    }
  ]
}
```

o Add the following JSON under the type data and context Global (Organization Wide), replacing DATA_API_VERSION with your OCAPI Data API version, then click Save.

```
{
  "_v": "DATA_API_VERSION",
  "clients": [
    {
      "client_id": "CLIENT_ID",
      "resources": [
        {
          "resource_id": "/product_search",
          "methods": [
            "post"
          ],
          "read_attributes": "(**)",
          "write_attributes": "(**)"
        },
        {
          "resource_id": "/sites",
          "methods": [
            "get"
          ],
          "read_attributes": "(**)",
          "write_attributes": "(**)"
        }
      ]
    }
  ]
}
```

▪ Once added, the settings should appear similar to this:

Administration > Site Development > Open Commerce API Settings

Open Commerce API Settings

This page allows you to make client application-specific configurations of Open Commerce API resources, i.e. manage resource access privileges, attribute read/write permissions, or configure client application-specific response headers. You can select the actual type of the API that you want to configure and specify whether your settings are site-specific or global (organization-wide). Please note that due to caching, changes may take up to three minutes to become effective. You can browse the Open Commerce API here API Explorer.

Select Type: Data   Select Context: Global (organization-wide)

{
  "_v":"19.10",
  "clients":[
    {
      "client_id":"xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
      "resources":[
        {
          "resource_id":"/product_search",
          "methods":["post"],
          "read_attributes":"(**)",
          "write_attributes":"(**)"
        },
        {
          "resource_id":"/sites",
          "methods":["get"],
          "read_attributes":"(**)",
          "write_attributes":"(**)"
        }
      ]
    }
  ]
}

Code isml Integration

- Add the following code to the end of your footer_UI.isml file:

  `<isinclude template="components/footer/klaviyo_footer"/>`

- Import Add the following code to your cart templates for adding and editing items in the cart:

```
<isif condition="${pdict.CurrentHttpParameterMap.cartAction == 'add' ||
pdict.CurrentHttpParameterMap.cartAction == 'update'}">
    <isinclude url="${URLUtils.url('Klaviyo-RenderKlaviyoAddToCart')}"/>
</isif>
```

Be careful to check imports in Klaviyo cartridges if you are overwriting core cartridges.

External Interfaces

All calls made from tracking and identification on the pages are done through the Javascript Track and Identify calls which can be referenced in
https://apidocs.klaviyo.com/reference/javascript-client-library.

For the server side calls pertaining to List Subscription and Track call for order confirmation, please refer https://apidocs.klaviyo.com/reference/track-identify and https://apidocs.klaviyo.com/reference/lists-segments.

Firewall Requirements

N/A

Testing

Klaviyo tracks every event except "Order Confirmation" by placing a data-structure in the DOM named "_learnq". There is a full list under "Storefront functionality" that shows what events to expect to see. You can see if events are being tracked by searching for the variable "klParsedData" in the Browser's DOM:

```
> klEventTag
< "{"data":{"Items":["Long Sleeve Button Out Turtle Neck","Swing Tank","Casual Spring Easy Jacket","Casual Spring Easy Jacket",
```

Login to Klaviyo and go to https://www.klaviyo.com/dashboard

To see the service in action, login to the storefront and cookie yourself by adding **"utm_email=test@test.com"** test the user activities such as viewed product, add to cart, active on site, search, order placed to see the events come through to Klaviyo.

Click on the Activity Feed to see all the activities and events tracked like Viewed Product, Add to Cart, Checkout Started , Subscribed to list etc..

Click on Details to see the event data

Operations

Data Storage

N/A

Availability

N/A

Support

For technical support or any issues related to Klaviyo, please contact your Klaviyo Contact. For specific questions related to the cartridge, please reach out via email to support@klaviyo.com

User Guide

Roles, Responsibilities

N/A

Business Manager

Please refer the Business Manager Section above for setup related information. There is no other Business Manager setup required other than the initial configuration stated in that section.

Storefront Functionality

The storefront integration will invoke the following events and tracking for the following functionality:

- o Added to Cart – Event fired when user adds/modifies his/her cart

- o Checkout Started – Event fired when user initiates the checkout process

- o Order Confirmation – Event fired when the user places his/her order, useful for real-time receipts

- o Site Search – Event fired when a user initiates a search.

- o Viewed Category – Event fired when a user views a category page.

- o Viewed Product – Event fired when a user views a product page. The information regarding the viewed product is passed along to Klaviyo here.

Known Issues

N/A

Release History

Version	Date	Changes
01/01/00	22nd March 2019	Initial release
01/02/00	28th January 2020	Updated for image links and language clarification
01/03/00	20th February 2020	Updated for Salesforce Commerce Cloud Certification process
21.10.9	14th October 2021	Removed references to the KlaviyoSubscriptionService and KlaviyoCheckSubscriptionService, added information and snippets about the OCAPI setup