

**Feedonomics Integration Guide**

**Version: 19.1.0**

Contents

[**1.** **Summary** 3](#_Toc13742295)

[**2.** **Component Overview** 3](#_Toc13742296)

[2.1. Functional Overview 3](#_Toc13742297)

[2.2. Export specifics 3](#_Toc13742298)

[2.2.1. FeedonomicsProductExport Feed 3](#_Toc13742299)

[2.2.2. FeedonomicsProductInventoryExport Feed 4](#_Toc13742300)

[2.3. Limitations & Constraints 4](#_Toc13742301)

[2.4. Compatibility 4](#_Toc13742302)

[**3.** **Implementation Guide** 4](#_Toc13742303)

[3.1. Setup 4](#_Toc13742304)

[3.1.1. Setting Workspace 4](#_Toc13742305)

[3.1.2. Install Site Meta-Data 5](#_Toc13742306)

[3.1.3. Cartridge Path 5](#_Toc13742307)

[3.1.4. Service Definitions 5](#_Toc13742308)

[3.1.5. Jobs Configuration 6](#_Toc13742309)

[3.1.6. IMPEX Directory 7](#_Toc13742310)

[3.2. Configuration 8](#_Toc13742311)

[3.2.1. Configure Jobs 8](#_Toc13742312)

[3.2.1.1. FeedonomicsProductExport 8](#_Toc13742313)

[3.2.1.2. FeedonomicsInventoryExport 12](#_Toc13742314)

[3.3. External Interfaces 14](#_Toc13742315)

[3.4. Availability 14](#_Toc13742316)

[3.5. Support 14](#_Toc13742317)

[**4.** **User Guide** 14](#_Toc13742318)

[4.1. Roles, Responsibilities 14](#_Toc13742319)

[4.2. Business Manager 14](#_Toc13742320)

[5. **Known Issues** 15](#_Toc13742321)

[6. **Release History** 15](#_Toc13742322)

##### 

#### **Summary**

Feedonomics combines best-in-class technology and service to list your products everywhere people shop online, including Google Shopping, Amazon, and Facebook. Feedonomics services many of the world’s most prolific advertising agencies and brands, including over 30% of the top 1,000 internet retailers.

The Feedonomics Salesforce Commerce Cloud LINK cartridge allows for the scheduling of a full product feed. This document describes how to integrate the cartridge int\_feedonomics with your Business Manager.

Please contact Feedonomics ([hello@feedonomics.com](mailto:hello@feedonomics.com)) to obtain FTP credentials to export your product feed to Feedonomics.

This cartridge includes support for Controllers, SFRA and Pipelines.

#### **Component Overview**

## Functional Overview

The cartridge includes the following features:

• Product Export JOB pushes product data for each relevant site to Feedonomics FTP.

• Product Export JOB provides the functionality to push the data from a multi-locale which can be easily configured through the localeID job parameter.

* Product Inventory Export JOB pushes the inventory and pricing information of the product assigned to the relevant site to Feedonomics.
* Feed Files are pushed to Feedonomics via FTP.
* Both the JOBs (Product Export JOB and Inventory Export JOB) are designed in a chunk-oriented manner to avoid timeouts when there are many products.

## Export specifics

## FeedonomicsProductExport Feed

A site-level Product export job (‘FeedonomicsProductExport) has been provided as part of this cartridge. This job will export all products assigned to the relevant site to Feedonomics.

## FeedonomicsProductInventoryExport Feed

A site-level Inventory export job (‘FeedonomicsProductInventoryExport’) has been provided as part of this cartridge. This job will export inventory allocation and pricing information for all the products assigned to the relevant site to Feedonomics.

## Limitations & Constraints

There are no limitations and constraints for this cartridge. There is also no risk of optimistic locking failure because there no transaction is associated with any JOB run.

## Compatibility

This cartridge is compatible with all the three frameworks of SFCC (pipelines, controllers and SFRA version 4.2.1).

#### **Implementation Guide**

## Setup

## Setting Workspace

This repository contains int\_feedonomics cartridge inside the top-level folder link\_feedonomics under the cartridges folder. Add the int\_feedonomics cartridge to the Business manager site cartridge path after uploading.

1. Clone this repository. (The name of the top-level folder is link\_feedonomics).
2. If you are using Eclipse import int\_feedonomics project by navigating to the root folder i.e. link\_feedonomics and upload using UX studio.
3. If you are not using Eclipse then navigate to the link\_feedonomics folder where package.json is present.
4. In the link\_feedonomics folder, run **npm install** to all the package dependency required.
5. Update the file dw.json under the link\_feedonomics folder with contents:

{

"hostname":"your-sandbox-hostname.demandware.net",  
 "username": "yourlogin",

"password": "yourpwd",

"code-version": "version\_to\_upload\_to",

}

1. In the link\_feedonomics folder, enter the following command:**npm run uploadCartridge** , this will upload the cartridge to the BM server.
2. Use npm run lint to check any linting error.

## Install Site Meta-Data

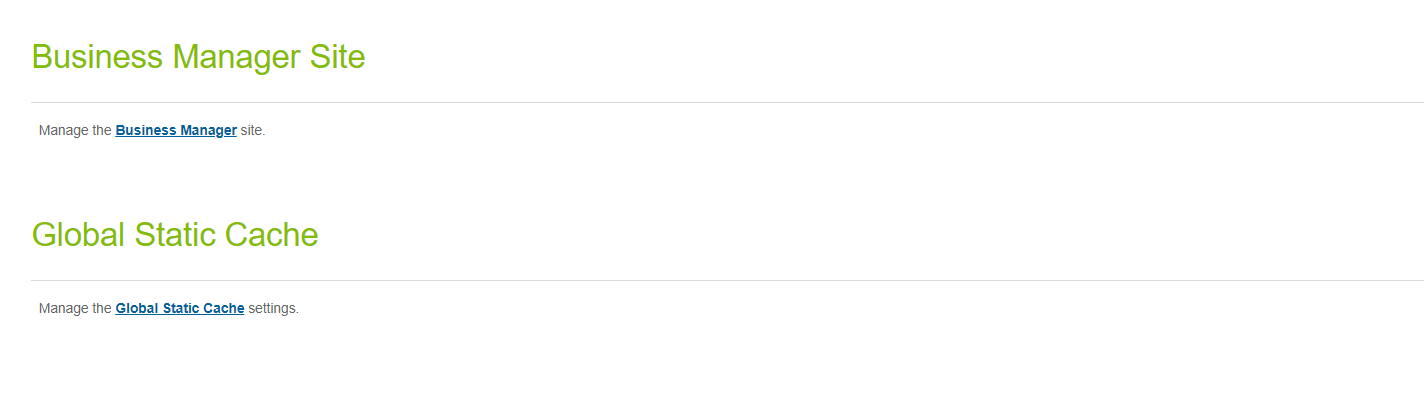
The site Meta data consists of JOB definition, services and site preference. These must be installed to support the cartridge code. To do this, perform the following:

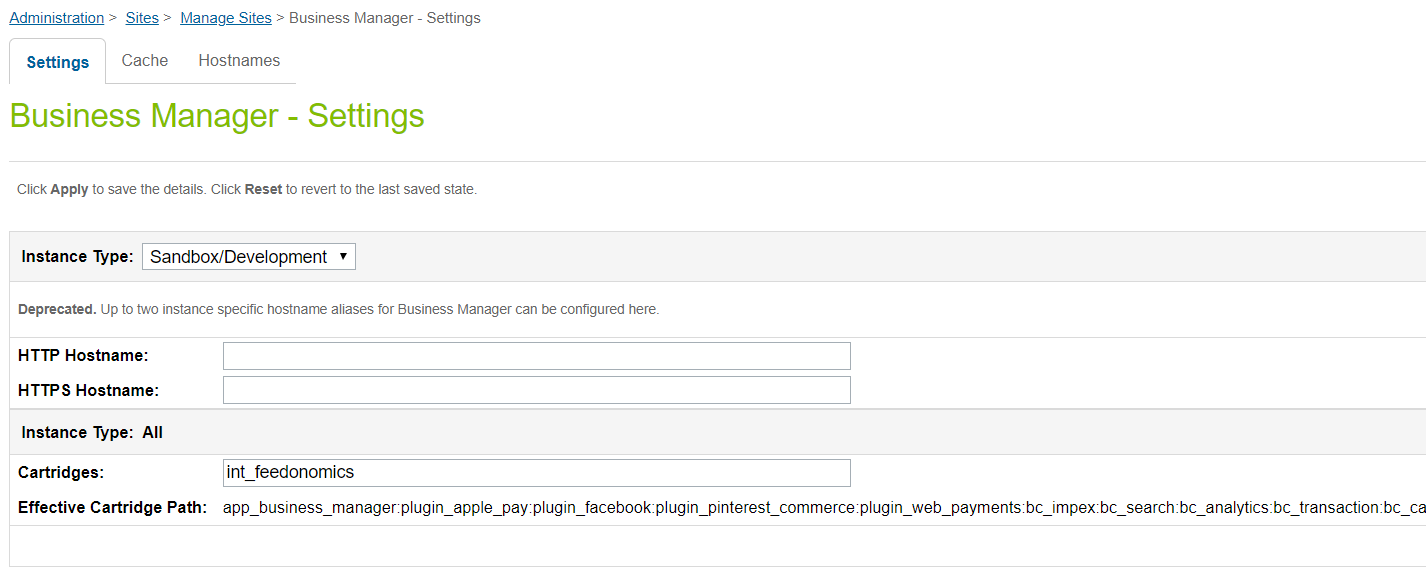
1. ZIP up the **metadata** directory
2. Upload the ZIP file via Administration > Site Development > Site Import and Export.
3. Select the file and click **Import.**

## Cartridge Path

The **int\_feedonomics** cartridge must be added to the Business Manager cartridge path only as it doesn’t have any storefront related functionality. To do this go to Administration > Sites > Manage Sites.

Click on Business Manager Sites





## Service Definitions

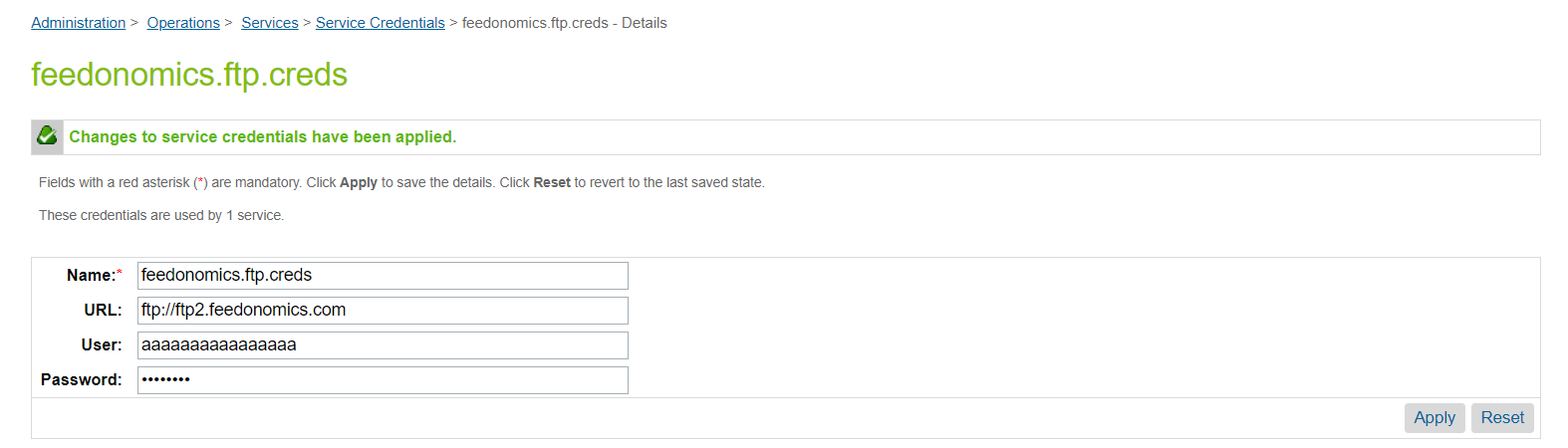
The new Services framework has been used to enable the feed transfer to Feedonomics using an FTP service client.

Service definitions are installed as part of the site import, but will require the following configuration:

Go to Business Manager, navigate from Administration > Operations > Services

Click on credentials Tab and click **feedonomics.ftp.creds** link.

1. Credentials URL : FTP Host
2. User / Password: Username and Password for FTP connection.



## Jobs Configuration

All scheduled jobs provided as part of the cartridge must be configured as required. As implementation requirements will vary, this configuration has been omitted from the cartridge meta-data. Specifically, the following must be configured for each Job:

1. Enabled/Disabled
   * All jobs are disabled by default
2. Schedule
   * Job schedule and interval to be defined
3. Notification Settings
   * Email notification to be provided in the event of job failure
   * This should be set to a current and active email address which is monitored so that issues can be identified and resolved.

Product Export Feed is currently configured for default locale only. If the requirement is for multi-locale then create a new job component inside the job and pass the locale id. Please refer to configure job section for detailed steps.

**\*\*\*** In case job is disabled for BM account then navigate to

**Administration > Organization > Roles > Administrator - Business Manager Modules** to enable Jobs and Job history

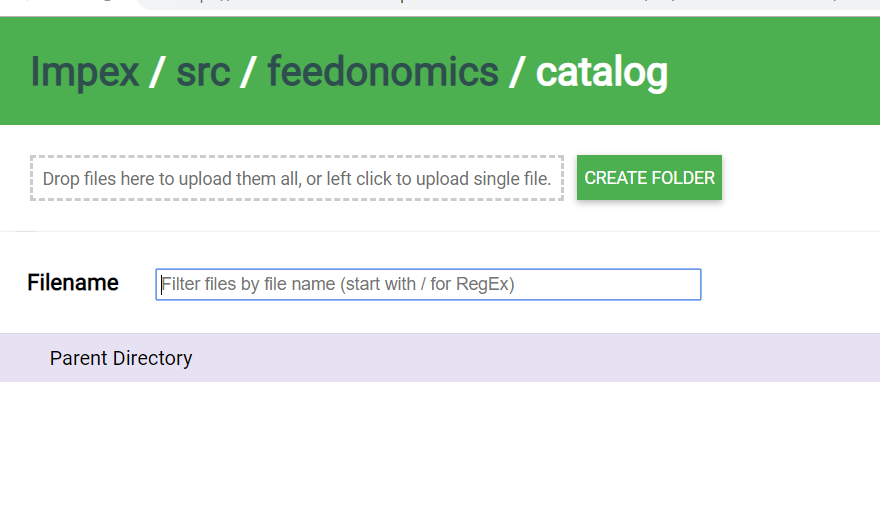
****

## IMPEX Directory

The IMPEX directory can be monitored by the respective JOBs itself but after the initial import of the metadata, default values of the site folders will be present and can be overridden in JOB components.

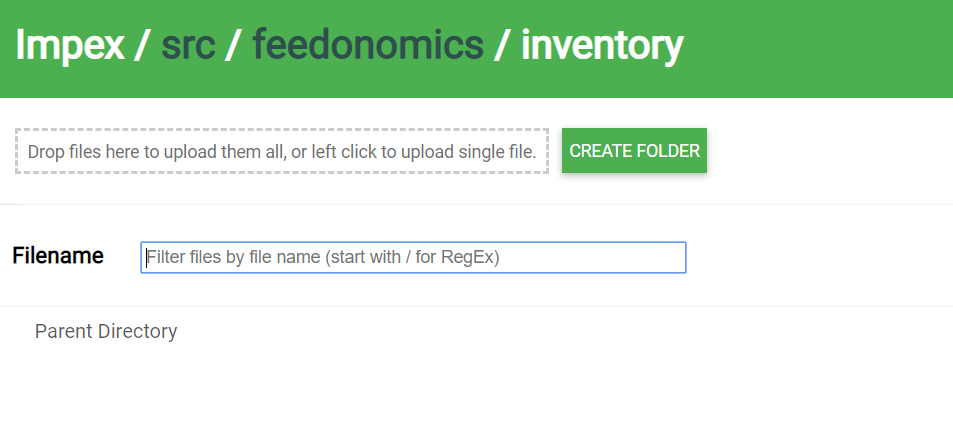
**FeedonomicsProductExport Feed**:

Target Folder: src/feedonomics/catalog



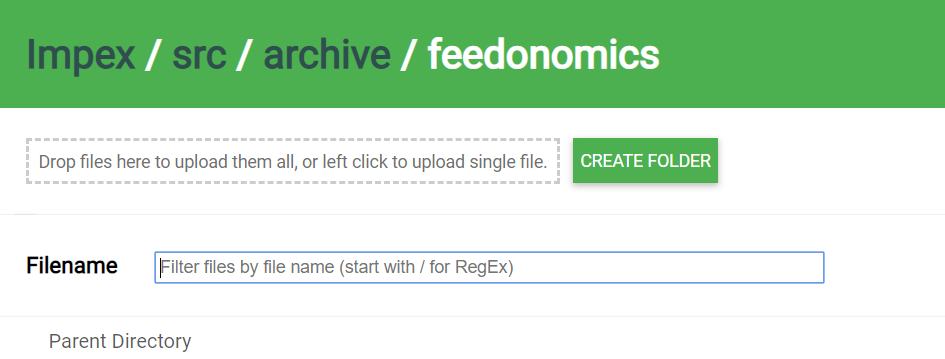
**FeedonomicsProductInventoryExport Feed**:

Target Folder: src/feedonomics/inventory



**FTP Upload:**

Archive Folder: src/archive/feedonomics



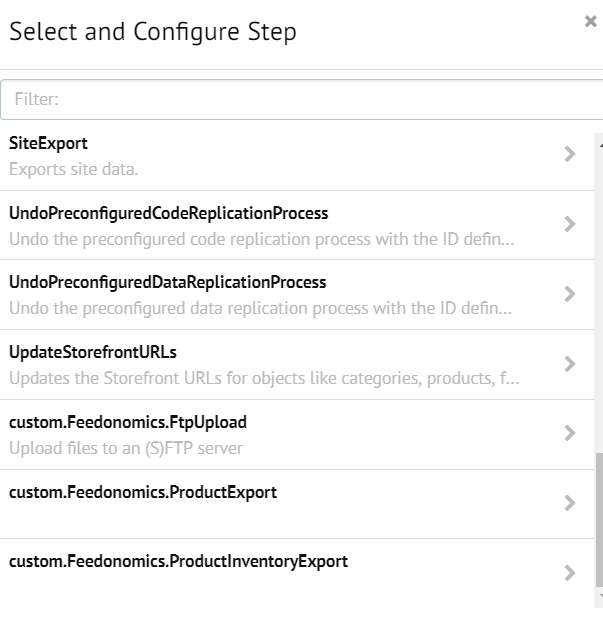
## Configuration

## Configure Jobs

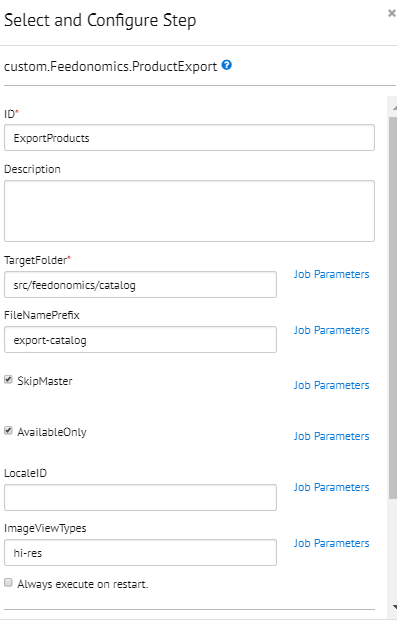
## FeedonomicsProductExport

Feedonomics Product Export job runs in site context which exports all the products assigned to the site. This job supports the multi-locale functionality which creates the specified locale file at the IMPEX and later uploads the same to the FTP location specified in the feedonomics.ftp service.

1. A custom feedonomics product export job component has been created as part of this job which is managed by code’s steptype.json.



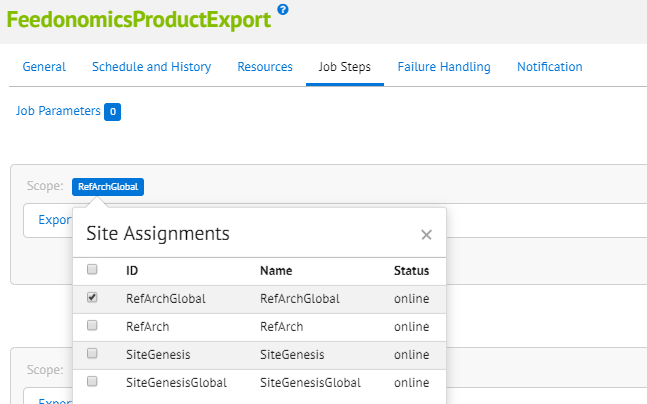
**custom.Feedonomics.ProductExport:** This custom job component creates a new CSV format file at IMPEX and writes all the product data to that.



**This component has six job parameters:**

1. **TargetFolder**:Location relative to the IMPEX directory where product export feed file will be written.
2. **FileNamePrefix**: File Name prefix of the product export feed file. If this file is set up for dummy site ID and locale is en\_US then file name will be export\_product\_dummy\_en\_US.csv
3. **SkipMaster:** This parameter will skip the master products and default value is set as true. Uncheck this parameter to include master products in the feed.
4. **LocaleID:** This parameter sets the request locale to the specified value in this field and if not set then it will be considered as default locale. Please specify the site locale id to create a locale specific file otherwise keep it empty for default locale.
5. **AvailableOnly:** This parameter sets the request for instock products only. Default value set as true, unchecking this parameter includes all products.
6. **ImageViewTypes:** This parameter is used to retrieve the product images for custom viewtypes defined separated by comma. E.g. hi-res,medium

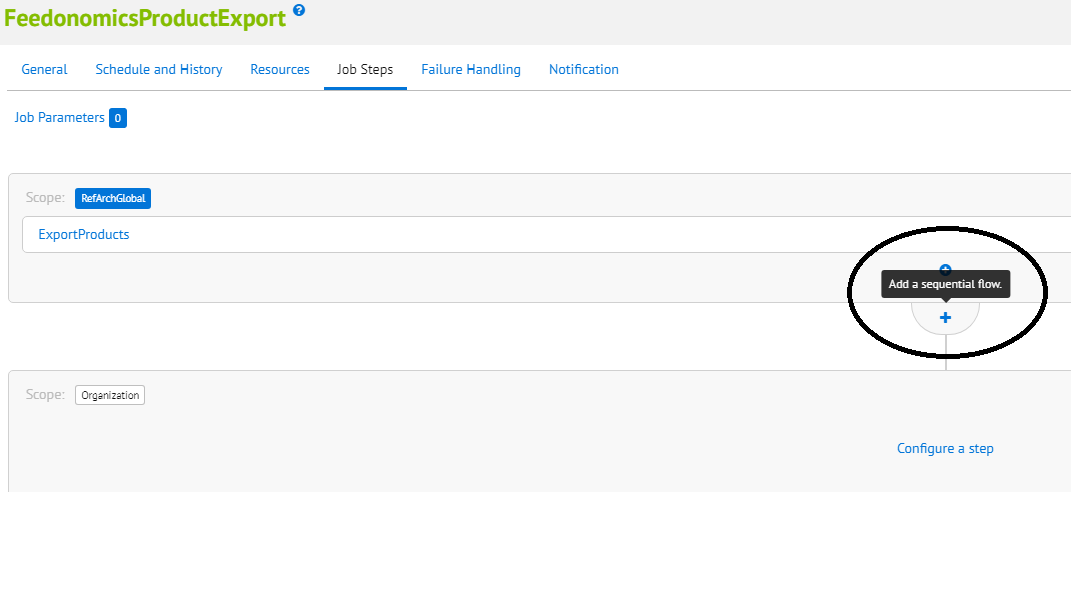
**\*\*\*Select the relevant sites for this job component to export feed files.**



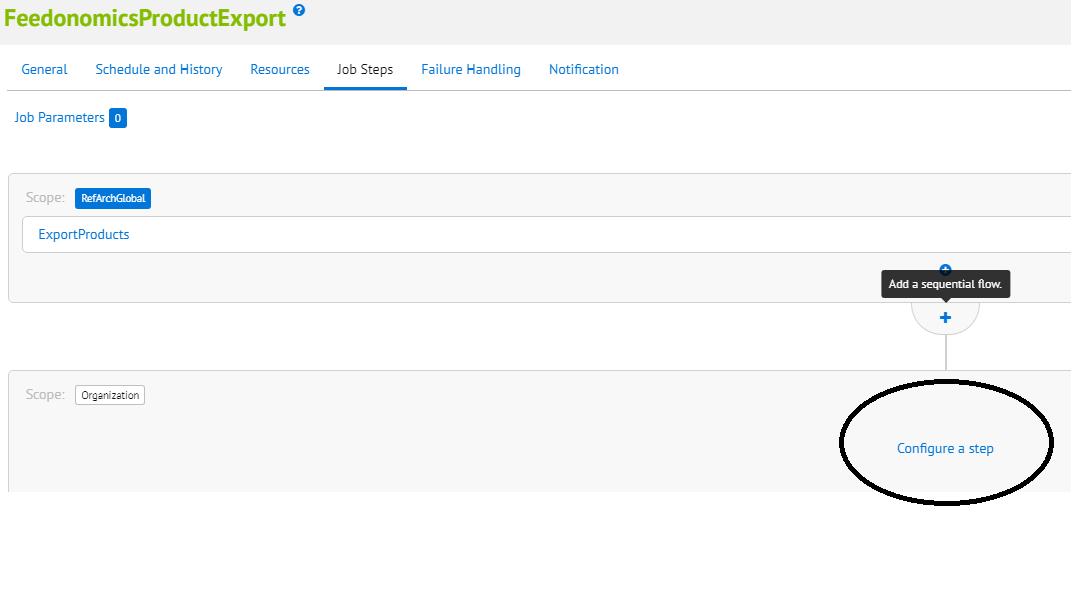
The default import of the metadata will create this JOB for default locale only. As per the requirement edit the localeID parameter value to set the required locale.

In case of multi-locale please follow the below steps:

1. Click on “+” button to add a new sequential flow.

****

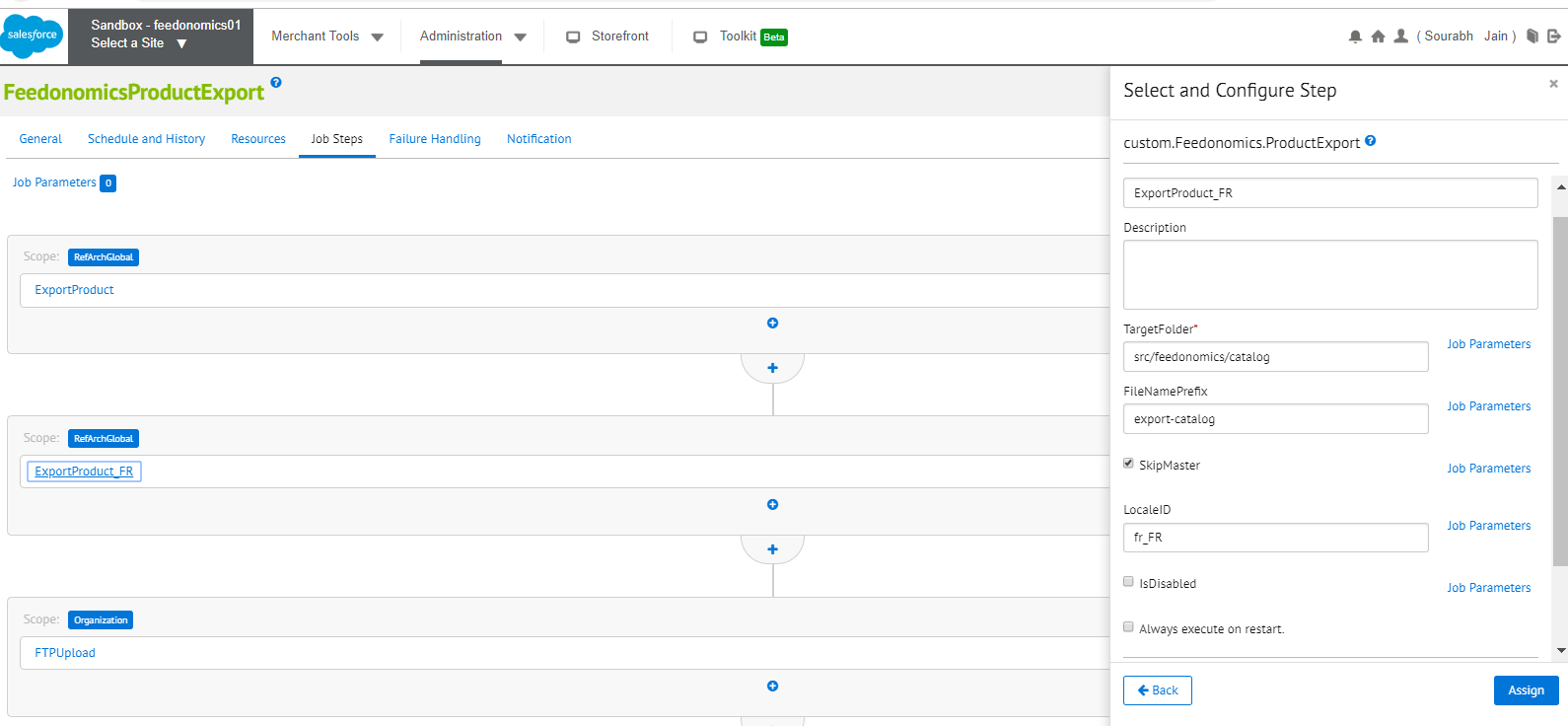
1. Click on “Configure a step”.



1. Under the settings select “custom.Feedonomics.ProductExport” as the job component.
2. Enter ID (i.e. a name for job step, like “ExportProduct\_fr\_FR”)
3. Enter the LocaleID of the locale for which feed needs to be exported. Refer to Administration >  Global Preferences >  Locales for the list of localedIDs
4. Once added, set the scope by clicking the “Organization” box next to Scope and select the Site for which the feed needs to be exported.

Follow the steps above to set up separate job step for each Site – Locale combination. **It is advisable to select only one site\_id per locale job step.**

Create as many as JOB components as a sequential flow, use the exact default component definition and update the localeID value as per requirement. In the example below, JOB will export two product feeds. One is default locale and the other is in the FR locale.



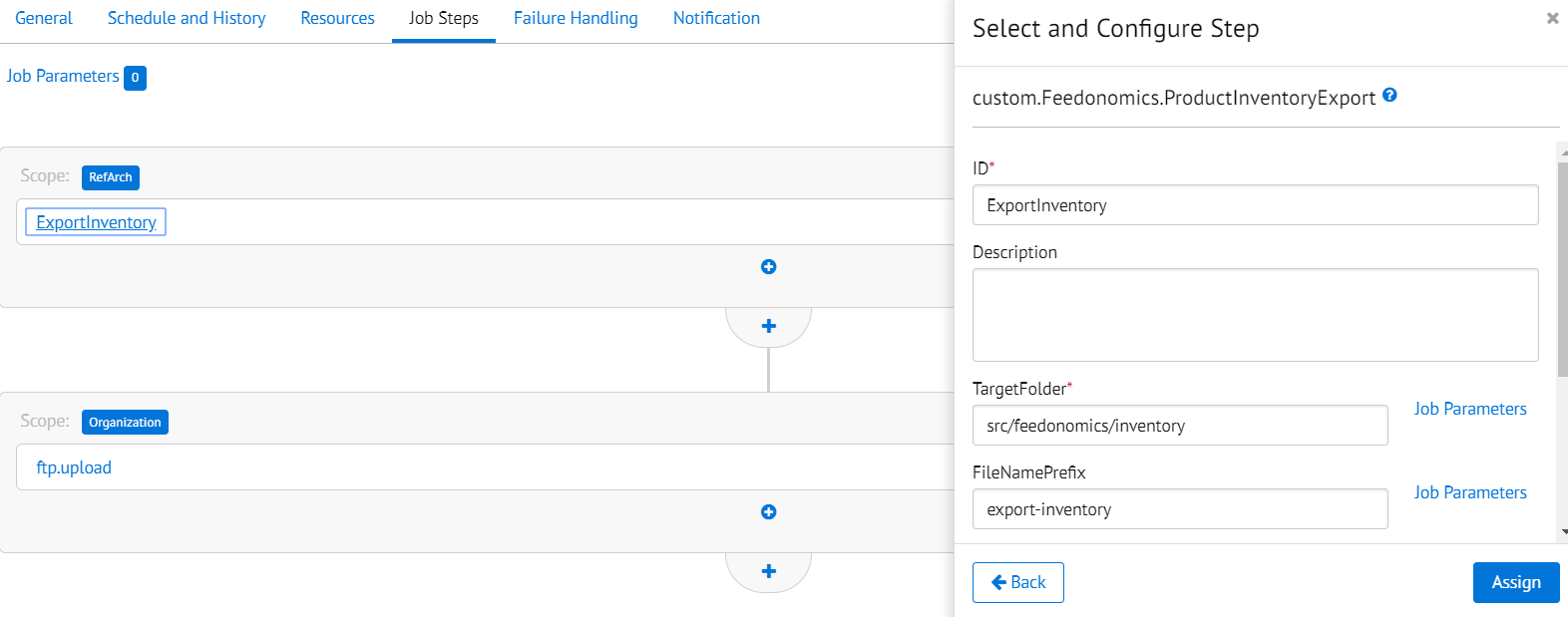
In case of multi storefront multi locale export either create a new job definition or create a separate job component if particular locale id is not available for other site.

In case locale ID is unavailable this JOB will throw an error.

## FeedonomicsInventoryExport

Feedonomics Inventory Export job runs in site context which exports inventory data of all the products assigned to the site. This job writes the inventory and pricing data in the specified CSV format file under IMPEX and later uploads the same to the FTP location specified in the feedonomics.ftp service. This job has two job components:

1. **custom.Feedonomics.ProductInventoryExport**: A custom feedonomics product inventory export job component has been created as part of this job which is managed by code’s steptype.json.



This component has four job parameters:

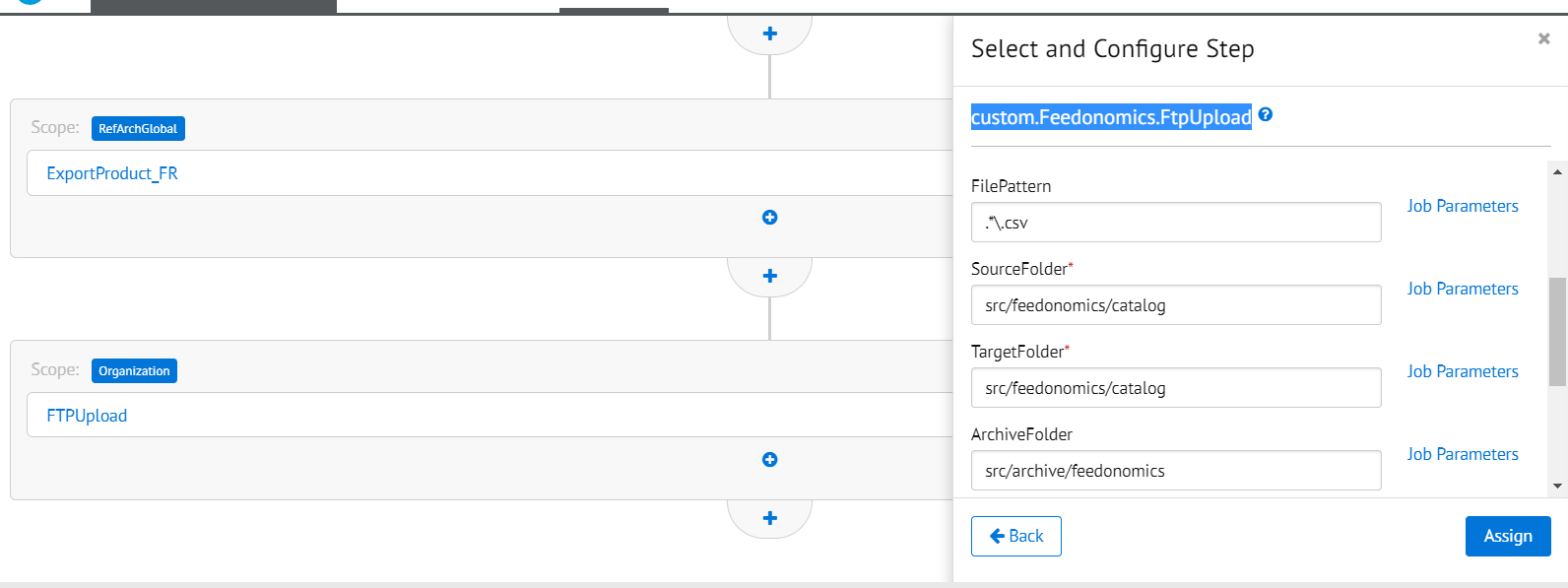
1. **TargetFolder**:Location relative to the IMPEX directory where product inventory export feed files will be written.
2. **FileNamePrefix**: File Name prefix of the product export feed file. If this file has been set up for dummy site ID and default locale is en\_US then the file name will be export\_inventory\_dummy\_en\_US.csv
3. **SkipMaster:** This parameter will skip the master products, default value is set as true. Uncheck this parameter to include master products in the feed.
4. **AvailableOnly**: This parameter sets the request for instock products only. Default value set as true, uncheck this parameter to includes all products.

**\*\*\*Select the relevant site for this JOB step same as product export JOB.**

**To export inventory for more than one site, create another sequential flow and set “custom.Feedonomics.ProductInventoryExport” as the component.**

**FTP Upload**: For both the JOBs, FTP upload will be the last JOB step which can be easily configured via custom.Feedonomics.FtpUpload job component.

1. **custom.Feedonomics.FtpUpload:** This job step uploads the feed files present in source folder to the target folder located at FTP location described as JOB parameter.



**\*\*\*This step run in organization context only and will copy to FTP all files created in previous job steps.**

## External Interfaces

FTP service credentials are required with feedonomics before the JOB run. Please contact Feedonomics ([hello@feedonomics.com](mailto:hello@feedonomics.com)) to obtain these.

## Availability

Feedonomics services are expected to be available at all times. Should there be a service failure, you will be notified via error emails with an address sourced from a configurable site preference

## Support

Support is to be provided through [FeedSupport](https://app.feedonomics.support/portal/home).

#### **User Guide**

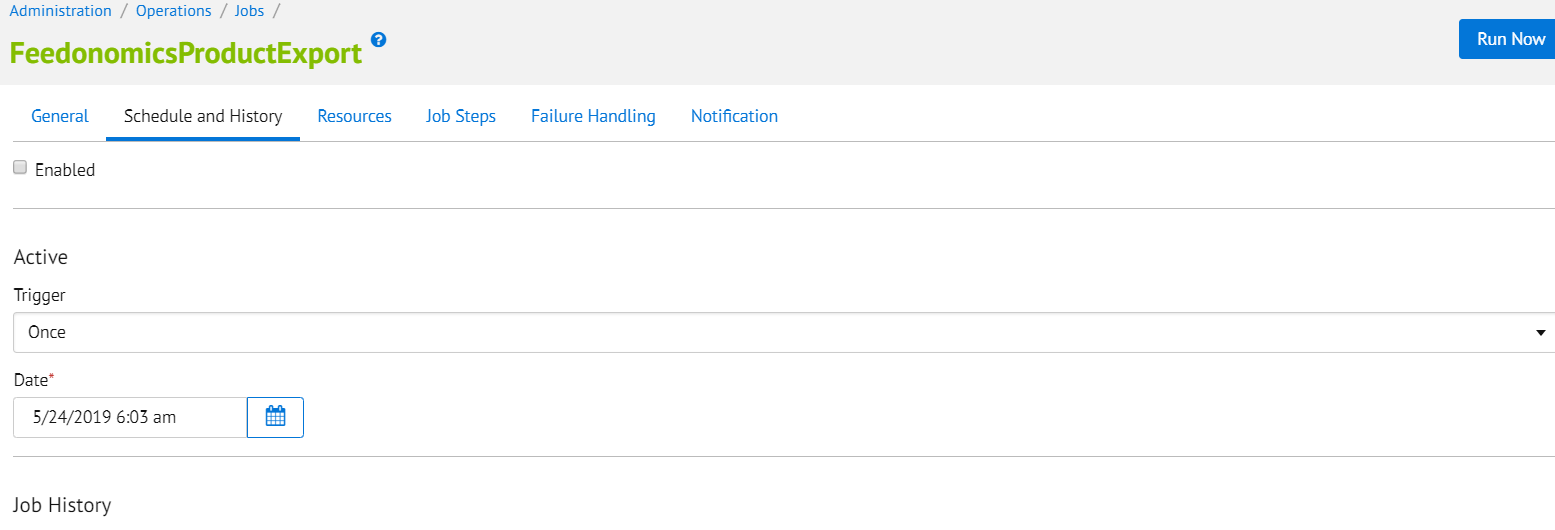
## Roles, Responsibilities

The error email address defined in the **JOB’s Notification** setting should be monitored proactively to address any issues with the jobs.

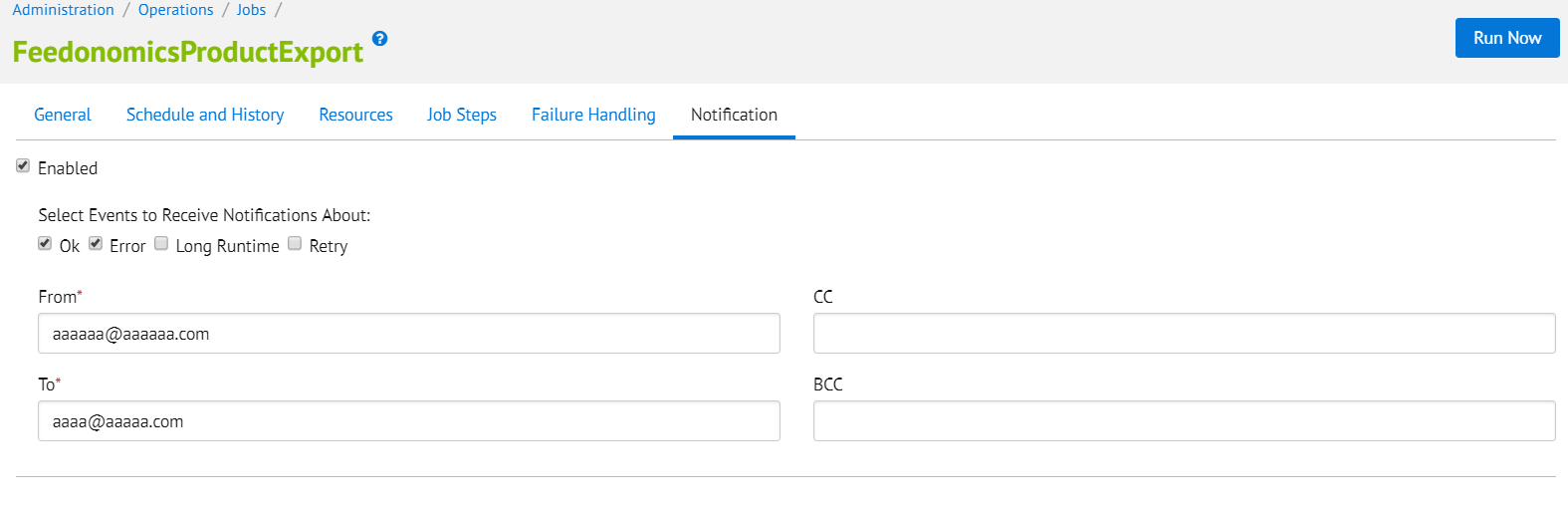
## Business Manager

After importing the metadata, schedule the JOBs as per the requirement.

FeedonomicsProductExport is to be scheduled once a day (preferably around 12:00 AM). FeedonomicsInventoryExport can be scheduled for a different time and frequency depending on how frequent inventory updates need to be pushed. It is advisable to schedule the FeedonomicsInventoryExport to execute after the inventory is updated and reconciled in the instance.



Set the email address for notification on JOB failure.

`

## Testing

In order to test this cartridge, run FeedonomicsProductExport and FeedonomicsInventoryExport jobs after configuring them. After successful export files will be archived to the respective archive folder at IMPEX location and will be uploaded to (S)FTP location. In case of error please refer to job error logs for detailed info and contact Feedonomics.

#### **Known Issues**

If any job runs more than 60 mins and gives the script timeout error then raise a Salesforce Support Ticket they will check the required configuration on your system like elastic search and will enable the same.

#### **Release History**

|  |  |  |
| --- | --- | --- |
| **Version** | **Date** | **Changes** |
| 19.1.0 | 5/30/2019 | Initial release |