

Product Restriction

Functional Design Document

Document History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Updated by** | **Date** | **Comments** |
| 1.0 | Jayachandra Gudiwada | 03/19/2015 | DRAFT |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Document Sign-off

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Role** | **Date** | **Signature** |
| **Pascal ESPINOUSE** |  |  |  |
| **Aliasgar Muchhala** |  |  |  |
| **Gyaneshwar Dubey** |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Document references

|  |  |  |  |
| --- | --- | --- | --- |
| **Reference and Document Title** | | **Version Number** | **Date** |
|  |  |  |  |

# Table of Contents

[1. Introduction 3](#_Toc414877253)

[1.1 Document Purpose 3](#_Toc414877254)

[1.2 Intended Audience 3](#_Toc414877255)

[1.3 Document Scope 3](#_Toc414877256)

[1.4 Assumption 3](#_Toc414877257)

[1.5 Document Structure 3](#_Toc414877258)

[1.6 Additional Resources 4](#_Toc414877259)

[1.7 Interface List 4](#_Toc414877260)

[2. Product Restriction interface 5](#_Toc414877261)

[2.1 Description 5](#_Toc414877262)

[2.2 Extraction Half interface connectivity overview 5](#_Toc414877263)

[2.3 Product restriction Interface behavior 5](#_Toc414877264)

[2.4 Naming convention for File 5](#_Toc414877265)

[2.5 Data 6](#_Toc414877266)

[2.6 File / Message processing 6](#_Toc414877267)

[2.7 Transformations 7](#_Toc414877268)

[2.8 Exception Management and Logging 8](#_Toc414877269)

[2.8.1 Exception Management 8](#_Toc414877270)

[2.8.2 Logging / Audit 8](#_Toc414877271)

[2.9 Security requirements 9](#_Toc414877272)

1. Introduction
   1. Document Purpose

The purpose of this document is to define functional requirement of product restriction interface. The document also provides detailed descriptions of each business rule.

* 1. Intended Audience

This document is aimed at Hybris developers and Technical members who like to understand a functional requirement of product restriction interface.

* 1. Document Scope

The scope of this document is to covers the details about the product restriction functionality interface which will suppose to be implementing using hot folder addon.

* 1. Assumption
* Business Unit & SKU ID should be predefined in the Hybris database before pass this SKU using CSV file.
* All the three column that need to be pass in the CSV file is mandatory.
  1. Document Structure

|  |  |  |
| --- | --- | --- |
| Section | | Description |
| 1 | Introduction | Describes the purpose and intended audience of the document and explains its structure. |
| 2 | Brief Description | Overview of the module. |
| 3 | Appendices | Supporting information |

* 1. Additional Resources

|  |  |  |  |
| --- | --- | --- | --- |
| # | Description | URL | Purpose |
| 1 | Wiki Portal | <https://wiki.hybris.com> | To gather more information |
| 2 | Data importation | <https://wiki.hybris.com/display/accdoc/Data+Importing+Capability+in+the+Commerce+Accelerator> | Documentation of import process |

* 1. Interface List

|  |  |  |
| --- | --- | --- |
| # | Interface name | Product restriction Sample data CSV file |
| 1 | Product Restriction |  |

1. Product Restriction interface
   1. Description

The product restriction interface is useful to restrict the products visibility to base on the B2BUnit.It manage using CSV file that will put on the network folder and process it by Hybris System. The Hybris system will process this file and validate a business rule. If validation is failed than it will move a file in an error folder else it will move a file in archive.

* 1. Extraction Half interface connectivity overview

NFS/ERP

Hybris GUI

Hybris integration

Description: Using NFS, ERP or Customer has to place product restriction data in the hybris intergration folder (**hybris\data\acceleratorservices\import\master\customers**). From this folder product restriction customer data will be processed and the same can be reflected in Hybris GUI.

* 1. Product restriction Interface behavior

The poller refreshes the directory at a given frequencies (default: 1000 ms).If a matching file is present in the directory it imports and process with parameters starts automatically (conversion into IMPEX format, importation, clean).The interface error module formalizes the errors messages to make them understandable by the user. This error message will be logged in a logger file.

* 1. Naming convention for File

|  |  |  |  |
| --- | --- | --- | --- |
| **Pre Fix** | **Post Fix** | **Extension** | **Name of the File** |
| ProductRestrictionCustomerData | -xxx..n  Here **x** is number | .csv | ProductRestrictionCustomerData-xxx..n.csv |

Example: ProductRestrictionCustomerData -001.csv

Description:

Prefix: ProductRestrictionCustomerData

Postfix: -001

Extenstion: .csv

The file must be placed in the folder: hybris/data/acceleratorservices/import/master/customers

* 1. Data

Below is the sample data



Description of Product restriction CSV File

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| # | Column Name | Data Type | Mandatory (true/false) | Default Value | Reserved words | Description |
| 1 | Product Code | String | true | / | / | This is dynamic value but it should be defined in Hybris database |
| 2 | B2BUnit UID | String | true | / | / | This is dynamic value but it should be defined in Hybris database |
| 3 | Is Visibility | Boolean | true | false | true or false |  |

* 1. File / Message processing

Below steps should follow during process a product restriction file:

Step 1: The product restriction file should be put in a NFS folder

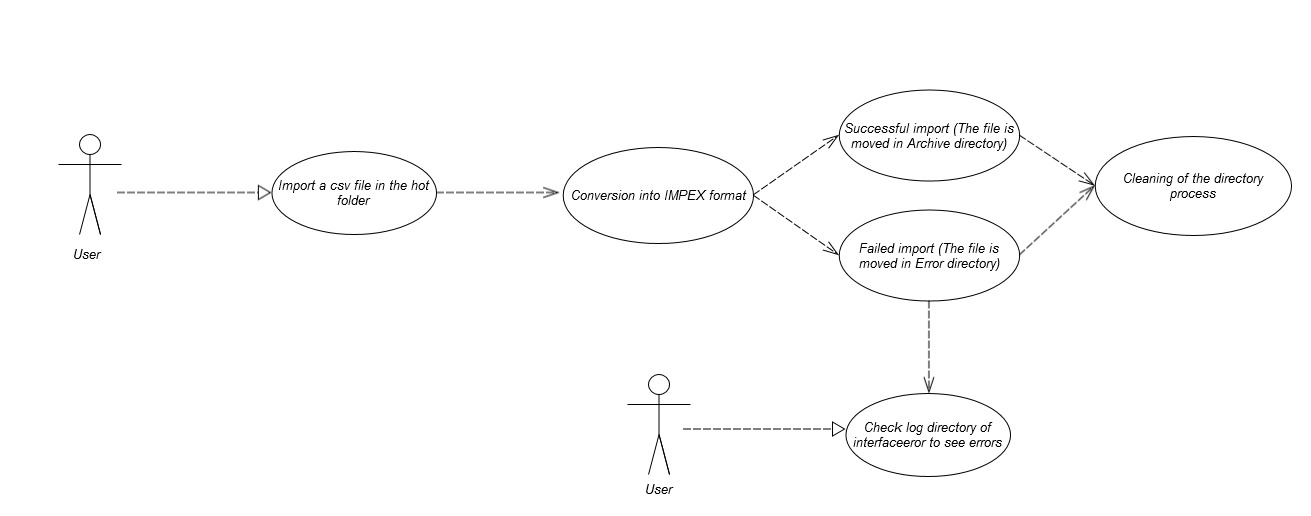
Step 2: The file name should be validate by Hybris System.

Step 3: If file name is valid then only it is picked by Hybris System and transform

Step 4: If file having valid data then it will upload a data in a Hybris database and move a file to archive folder.

Step 5: If file having invalid data than it will not upload anything in a Hybris Database and move a file to error folder.

Step 6: In logger file entire message should be logged either for Successful process or unsuccessful



* 1. Transformations

The below steps will be followed to process a CSV data file after placing the CSV file in hybris system.

Step 1: The file will be moved to processing folder to setup header.

Step 2: Before setting the headers the file naming conversion will checked. If the naming conversion having any issue then it will be rejected and it will be moved to reject folder.

Step 3: CSV file will be transformed into impex file.

Step 4: To read the CSV file CSV reader has to use.

Step 4: if there is any problem in CSV file then it will moved to error folder. The errors will be logged into the log file.

Step 5: After converting CSV to impex file then impex import will be executed to process the impex file.

Step 6: once the file is processed successfully then the processed file will be moved to archive folder.

* 1. Exception Management and Logging
     1. Exception Management

|  |  |
| --- | --- |
| **Exception** | **Message** |
| UnresolvedValueException | Exception caught when :   * error finding existing item * no existing item found for update * cannot create due to unresolved mandatory/initial columns |
| InsufficientDataException | Exception caught when :   * target type is not permitted by current header – {header} * cannot import localized column without default session language |
| ImpexException | Exception caught when :   * unknow mode (INSERT | UPDATE | INSERT\_UPDATE | REMOVE) * batch inserts are not supported – found more than one item for value line {valueLine} * Configured type {type} is abstract and you want to insert {valueLine}. Please specify the subtype of the item you want to create at the first column of the value line or change the configured type * Composed type {code} of found item with PK {PK} is not an instance of specified type {typecode} * Composed type {code} of found item with PK {PK} is not permitted in Insert\_Update mode, because of {typeErrorMsg} |
| HeaderValidationException | Exception caught when :   * cacheUnique modifier is not allowed for Singleton types, please remove cacheUnique modifier |

* + 1. Logging / Audit

Product restriction file process fails then logs check in the **hybris\log\interfaceerror** folder.If the file gets rejected then it will be tracked in **hybris\log\interfaceerror\reject**

List of error messages:

1. Missing a mandatory field for SKU(product id).

Original file: ProductRestrictionCustomerData-0012.csv;line: 14;message: Missing value for 0;original line: ;Pronto Goods;true

2. Original file: ProductRestrictionCustomerData-0012.csv;line: 14;message: Missing value for 1;original line: 23456;;true

* 1. Security requirements

NFS folder must have a read and write access permission to place a file in Hybris system for further process.