

Media Import Half Interface

Technical Design Documentation

Document History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Updated By | Date | Comments |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Document Sign-Off

|  |  |  |
| --- | --- | --- |
| Name | Role | Signature/Date |
| Pascal ESPINOUSE |  |  |
| Jeremy REYNARD |  |  |
| Maxime WEHRLIN |  |  |
| Daniel CRISAN |  |  |

Table of Contents

[1. Introduction 3](#_Toc417031994)

[1.1 Purpose 3](#_Toc417031995)

[1.2 Intended Audience 3](#_Toc417031996)

[1.3 Document Scope 3](#_Toc417031997)

[1.4 Functional Rules 3](#_Toc417031998)

[1.5 Assumptions/Restrictions 3](#_Toc417031999)

[1.6 Documents Structure 4](#_Toc417032000)

[1.7 Additional Resources 4](#_Toc417032001)

[1.7.1 Useful Site 4](#_Toc417032002)

[1.7.4 Product Media Data 5](#_Toc417032003)

[2. High Level Design Diagram 6](#_Toc417032004)

[2.1 Overview 6](#_Toc417032005)

[2.1.2 Strategy 6](#_Toc417032006)

[2.1.2 Impex example 7](#_Toc417032007)

[2.1 Sequence Diagram 8](#_Toc417032008)

[2.2 Use Case Diagram 9](#_Toc417032009)

[3. Detailed Application Design 10](#_Toc417032010)

[2.1 Class Diagram 10](#_Toc417032011)

[2.2 Beans 10](#_Toc417032012)

[2.3 Facade 10](#_Toc417032013)

[2.4 Strategy 10](#_Toc417032014)

[2.5 Controller 10](#_Toc417032015)

[2.6 View/JSP 11](#_Toc417032016)

[2.7 ER Diagram 11](#_Toc417032017)

[3. Logging 13](#_Toc417032018)

[4. Configuration and Start 14](#_Toc417032019)

[4.1 Configuration Steps for the Export Product Media module 14](#_Toc417032020)

[4.2 Start 14](#_Toc417032021)

[6. Appendix A – Glossary of terms 15](#_Toc417032022)

1. Introduction
   1. Purpose

This document is the Technical Design Document which provides information of exposition of the image Data and steps to configure the module. It covers:

* High Level Design
* Application Architecture
* Database Architecture
* Configurations Steps
  1. Intended Audience

This document is aimed at Hybris developers and Technical members who want to implement the Exposition Image resizing module.

* 1. Document Scope

This document covers the details about the sprint 1 Exposition Image resizing module.

* 1. Functional Rules

The Exposition image resizing module permits to export some data about products with file format CSV

* 1. Assumptions/Restrictions

This module has been created for hybris version 5.4.0.0 with the bases accelerator and the bases interfaces, Talend 5.6 and Mysql 5.6 Database.

* 1. Documents Structure

|  |  |  |
| --- | --- | --- |
| Section | | Description |
| 1 | Introduction | Describes the purpose and intended audience of the document and explains its structure. |
| 2 | High Level Design | Overview of the module. |
| 3 | Detailed Application Design | Application architecture applicable to this module. |
| 4 | Detailed Database Design | Conceptual and logical architecture applicable to this module. |
| 5 | Exception Management & Logging | Information about the Exception handling and logging. |
| 6 | Installation & Configuration | Steps to install and configure the module. |
| 7 | Appendices | Supporting information. |

* 1. Additional Resources

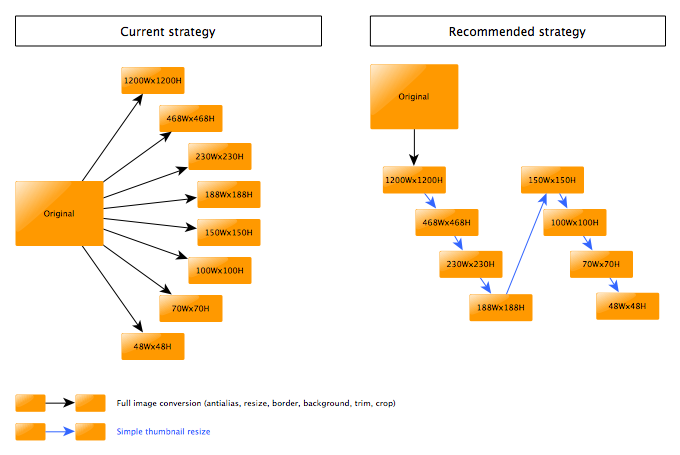
1.7.1 Useful Site

|  |  |  |  |
| --- | --- | --- | --- |
| # | Description | URL | Purpose |
| 1 | Wiki Portal | <https://wiki.hybris.com> | To gather more information |
| 2 | Coconet |  | To refer functional specifications |
| 3 | Data importation | <https://wiki.hybris.com/display/accdoc/Data+Importing+Capability+in+the+Commerce+Accelerator> | Documentation of import process |

1.7.4 Product Media Data

|  |  |  |
| --- | --- | --- |
| File ProductMediaData with data |  | Sample of ProductMediaData with data |
| Linked image |  |  |

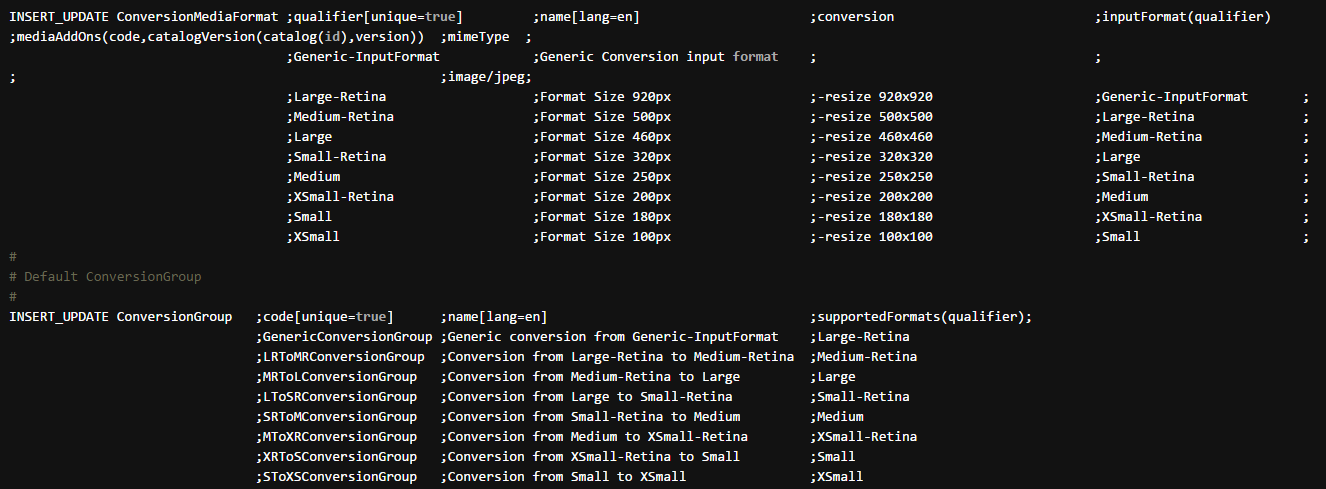
1. High Level Design Diagram
   1. Overview  
      1. Strategy

* 

Instead of resize images from the same original image, we use the previous image to create the new resized image.

This solution is more powerful because filters are applied one for the first image resizing. And it’s more powerful to resize an image from a smaller image.

* + 1. Impex example

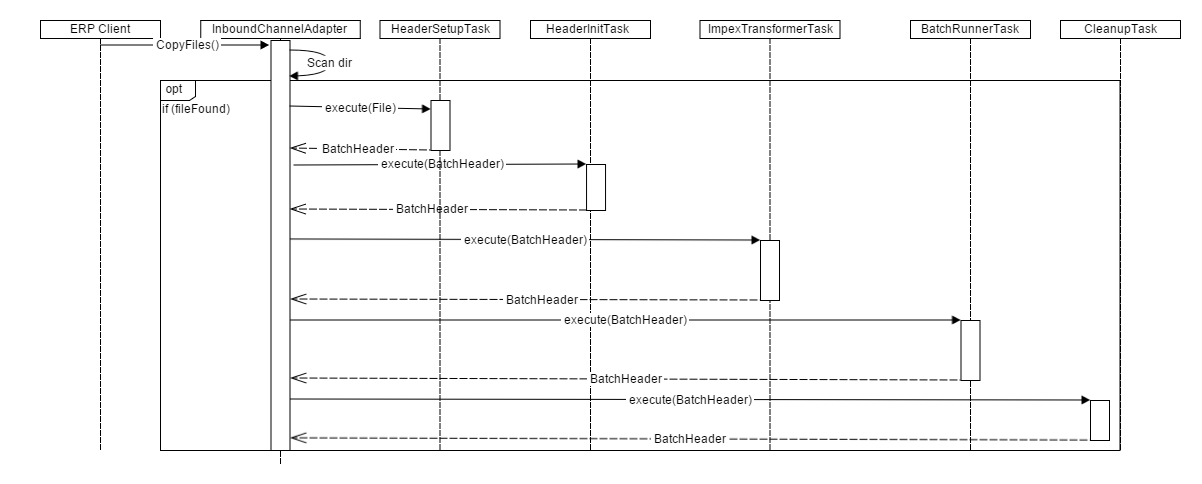


This example is used to configure the cron job for image resizing.

The cron job takes the previous resized image to create the new one.

* 1. Sequence Diagram

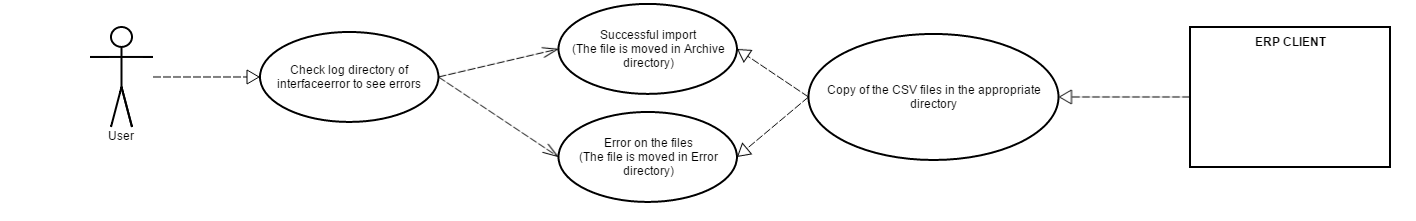
Sequence of events during the hot folder process



|  |  |  |
| --- | --- | --- |
| **#** | **Component** | **Description** |
| 1 | **ERP client** | The system of the client which provide the CSV files |
| 2 | **InboundChannelAdapter** | Spring integration scans folder and looks for new files that matches filename prefix |
| 3 | **HeaderSetupTask** | The task which create BatchHeader initialize import process with parameters Catalog and Net |
| 4 | **HeaderInitTask** | The task which initialize import process with parameters Lang and SequenceID |
| 5 | **ImpexTransformerTask** | The task that is responsible of the conversion into IMPEX format |
| 6 | **BatchRunnerTask** | The task which does the import process |
| 7 | **CleanupTask** | The task which clean the directory “process” and move files in directory “error” or “archive” |

* 1. Use Case Diagram

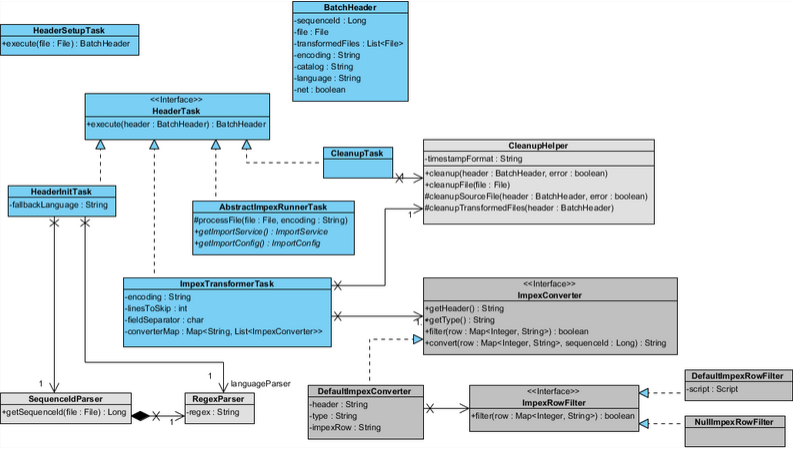
Use case diagram for the hot folder product addon



**Description:** User checks the result of the import process starts from ERP CLIENT.

3. Detailed Application Design

* 1. Class Diagram



* 1. Beans

**HeaderSetupTask:** The task which initialize import process with parameters.

**HeaderInitTask:** Define the langage of the import file and verify that the file isn’t a duplicate

**ImpexTransformerTask:** Define some variables for the conversion and do this.

**CleanupTask:** Clean the import files and move them into error folder or archive error.

* 1. Facade

N.A

* 1. Strategy

N.A

* 1. Controller

N.A

* 1. View/JSP

N.A

* 1. ER Diagram

**Media ER Diagram:**

| **Name** | **Description** | **Type** | **Default Value** | **Mandatory** | **Example** |
| --- | --- | --- | --- | --- | --- |
| Code | The media code | String | / | TRUE | 107701 |
| catalogVersion | Catalog version | CatalogVersion - Catalog version | Catalog:Staged | TRUE | Catalog version |
| mime | Mime type | String | / | TRUE | image/jpeg |
| folder | Folder of the media | MediaFolder - Media Folder | images | TRUE | images |
| realfilename | Real name of the file | String | / | TRUE | 49042000\_BLACK-1200Wx1200H.jpg |
| mediaFormat | Format of the Media | MediaFormat - Media Format | / | TRUE | 300Wx300H |
| media | Location of the exported picture | String | / | TRUE | c:\images\1128763\SUP123.jpg |

**Media Container ER Diagram:**

| **Name** | **Description** | **Type** | **Default Value** | **Mandatory** | **Example** |
| --- | --- | --- | --- | --- | --- |
| qualifier | The container code | String | / | TRUE | CMFContainer-1009 |
| medias | Link to the media | String | / | TRUE | /CMFGeneric/1234-1234 |
| catalogVersion | Catalog version | CatalogVersion - Catalog version | Catalog:Staged | TRUE | Catalog version |

**ProductMediaData Diagram:**

| **Name** | **Description** | **Type** | **Default Value** | **Mandatory** | **Example** |
| --- | --- | --- | --- | --- | --- |
| Product code | The product code | String | / | TRUE | 0010001 |
| Image path | Location of the exported picture | String | / | TRUE | /media/1234.jpg |

1. Logging

* A log file is created when an error occurred in one jobs, when the JAVA code generates an exception, when a warning is send by a component and when a component die, or when you try to create a null file.
* The location of the log folder is defined in the Context file
* The name of the log is the start Time of the job

|  |  |
| --- | --- |
| **Exception** | **Message** |
| IOException | Exception caught when :   * Failed or interrupted I/O operations. |
| FileNotFoundException | Exception caught when :   * The file is used by another process |
| MySQLSyntaxErrorException | Exception caught when :   * There are errors in the mysql request |

Log file example

1. Configuration and Start
   1. Configuration Steps for the Export Product Media module

If you want to modify the parameters of the **ExportProductMedia** module, go to the **Context.properties** file of the module.  
 You can change the database information connexion:

|  |  |
| --- | --- |
| The database used | base;Hybris |
| The host | host;localhost |
| The user | user;root |
| The password | pwd;root |
| The port | port;3306 |

You can change the name and the folder where the csv are imported: **outputFileName.** You can change the location of the log’s folder: **logFolder.** You can change the SQL request

|  |  |
| --- | --- |
| Name of the column in the database | columnName;products.code… |
| Table to Join to obtains the appropriate column if any | tableJoin; … |
| Clause for the request | where; WHERE medias.mime="image/jpeg" … |
| Version of the catalog | AND catalogversions.p\_version="Staged" |

* 1. Start

To launch the job, you must launch a script file with the time between the different exportation:  
 Example: “javac JobName TimeBetweenExportation“  
 Or directly with the Export<JobName>\_run.bat

6. Appendix A – Glossary of terms

|  |  |
| --- | --- |
| **Term/Abbreviation/Acronym** | **Meaning** |
| B2B | Business to Business |