# Customer Ecosystem - Technical Design Document.

* [Version History](#CustomerEcosystemTechnicalDesignDocumen)
* [Document Details](#CustomerEcosystemTechnicalDesignDocumen)
  + [Purpose of Document](#CustomerEcosystemTechnicalDesignDocumen)
  + [Scope of Document](#CustomerEcosystemTechnicalDesignDocumen)
  + [Application Functionality](#CustomerEcosystemTechnicalDesignDocumen)
  + [Business Context](#CustomerEcosystemTechnicalDesignDocumen)
  + [Audience](#CustomerEcosystemTechnicalDesignDocumen)
  + [Abbreviations](#CustomerEcosystemTechnicalDesignDocumen)
  + [Glossary](#CustomerEcosystemTechnicalDesignDocumen)
* [Design Overview](#CustomerEcosystemTechnicalDesignDocumen)
  + [Architecture Overview:](#CustomerEcosystemTechnicalDesignDocumen)
  + [The high-level architecture of the Customer Ecosystem is defined below.](#CustomerEcosystemTechnicalDesignDocumen)
  + [Detailed Design:](#CustomerEcosystemTechnicalDesignDocumen)
  + [Fast Track Data Warehouse Data Ingestion:](#CustomerEcosystemTechnicalDesignDocumen)
  + [Detailed Workflows:](#CustomerEcosystemTechnicalDesignDocumen)
* [Technical Specification:](#CustomerEcosystemTechnicalDesignDocumen)
  + - [ODL Customer Segment](#CustomerEcosystemTechnicalDesignDocumen)
* [Output Specification:](#CustomerEcosystemTechnicalDesignDocumen)
* [6. CSM0140 Customer](#CustomerEcosystemTechnicalDesignDocumen)
  + [ODL Layer](#CustomerEcosystemTechnicalDesignDocumen)
* [PySpark Job:](#CustomerEcosystemTechnicalDesignDocumen)
* [Offer Allocation & Segments Table](#CustomerEcosystemTechnicalDesignDocumen)
* [Orchestration:](#CustomerEcosystemTechnicalDesignDocumen)
* [Frequency:](#CustomerEcosystemTechnicalDesignDocumen)
* [Security Control:](#CustomerEcosystemTechnicalDesignDocumen)

# Version History

| **Version Number** | **Date** | **Revision Details** | **Author** |
| --- | --- | --- | --- |
| 0.1 | 01 Apr 2023 | Initial draft | [Rachel](file:///C:\display\~vn51cev) Peake |

# Document Details

## Purpose of Document

The Technical Design Document (TDD) captures the high-level summary of the solution being developed to accomplish the project's requirements.  The TDD is also used to create the Product Documentation needed to turn over a system to Production and other documentation necessary for installation and implementation.

## Scope of Document

This document covers the following for Customer Ecosystem

Functional overview

Technical overview  
Operational/Support information

## Application Functionality

## Business Context

Asda has successfully invested in advanced marketing technology and customers insights to power decisions. Price, Promo and Range insights will enrich this capability.

The business require a tool to enable the Commercial team (Buyers & surrounding teams) to make informed Promotion decisions. The business have engaged Quantium (third party Retail data experts) to deliver a Promotion Optimisation tool (Q-Promotion). The solution requires input from the Technology and Data teams to ensure strategic alignment and to progress the work.

## Audience

This document could be referred by following

* 1. Project Manager
  2. Business Analyst
  3. Developers
  4. Testers
  5. Support Personnel

## Abbreviations

|  |  |
| --- | --- |
| **Abbreviation** | **Description** |
| ETL | Extract, Transform and Load |
| SSIS | SQL Server Integration Service |
| CSV | Comma Separated Value |
| DBA | Database Administrator |
| HDFS | Hadoop File System |
| SQL | Structured Query Language |
|  |  |

## Glossary

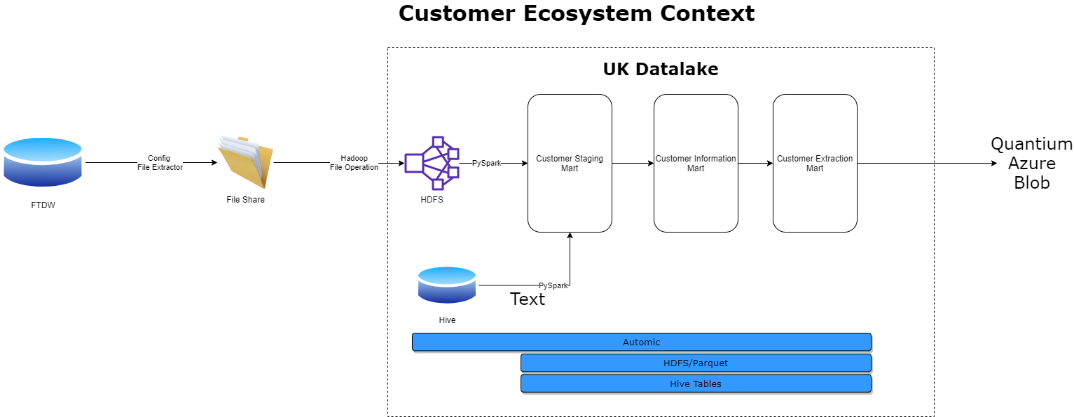
Include any descriptions of the terminology used in the document as required.

|  |  |
| --- | --- |
| **Term** | **Description** |
| FTDW | Fast Track Data Warehouse |
| Datalake | Hadoop Datalake, where the Customer Ecosystem Data resides. |
| RDL | Regional Datalake |
|  |  |
|  |  |

# Design Overview

## Architecture Overview:

## The high-level architecture of the Customer Ecosystem is defined below.



## Detailed Design:

Data required for the Customer Ecosystem is primarily sourced from 2 main systems:

* FTDW - Fast Track Data Warehouse
* Data lake

## Fast Track Data Warehouse Data Ingestion:

This Module's objective is to extract the required tables in FTDW in a '|' delimited file and transfer it to HDFS (Data lake). Two SSIS components (pre-build) are used for extracting and transferring the files.

**ConfigFileExtractor**: A SSIS package <<packagename>> is scheduled to run every 10 mins scans the table FTDW\_MetaConfig.dbo.ConfigFileExtractor and extracts the data and transfers the files to Local File share. For any new table that needs an extract, a new record to be inserted to the FTDW\_MetaConfig.dbo.ConfigFileExtractor table with all the details as shown in below example.

**Example**: Trading Hierarchy File

INSERT INTO FTDW\_MetaConfig.dbo.ConfigFileExtractor

(FolderDestination, FileName, AppendDateToFileName, SourceDB, SQLQry, StartTime, BCPTimeoutOverrideSecs,

ExtractEnabled, ColumnDelimiter, BCPCodePage, BCPRowTerminator)

VALUES ('\\ftdw\CommercialTransformation\Dev\CSM\' ,

'vwDailySales\_Datamart\_TradingHierarchy.csv',

0,

'LDM\_PROD\_SQLServer',

'SELECT [Seq\_No],[OD\_Number],[OD\_Name],[OD\_Hide],[OD\_NewDesc],[Category],[Catg\_Hide],[Sup\_Catg],[SC\_Hide],[Comm\_Dir],[CD\_Hide],[Trad\_Divn],[TD\_Hide],[Tot\_Exc\_Petl],[TEP\_Hide],[Tot\_Inc\_Petl],[TIP\_Hide],[LDM\_DT\_LastLoad] FROM [LDM\_PROD\_SQLServer].[dbo].[vwDailySales\_Datamart\_TradingHierarchy]',

'23:00:00.000',

900,

1,

'|',

'-C "28591"',

'\n');

**HadoopFileOperation**: A SSIS Package <<packagename>> is used to transfer the file from the local file system to HDFS on the DEV cluster. For any new files to the transferred, a new record to be inserted to FTDW\_MetaConfig.dbo.ConfigHadoopFileOperation table with all the details as shown in below example.

**Example**: vwDailySales\_Datamart\_TradingHierarchy.csv

INSERT INTO FTDW\_MetaConfig.dbo.ConfigHadoopFileOperation

values

(

'RegionalDataLake\_HDUKDEV1',

NULL,

'\\ftdw\CommercialTransformation\Dev\CSM\',

'vwDailySales\_Datamart\_TradingHierarchy.csv',

NULL,

0,

10,

'/user/svc\_uk\_cust\_rdl\_dev/sourceFiles/ftdw/dim\_trading\_hierarchy/',

'vwDailySales\_Datamart\_TradingHierarchy.csv',

NULL,

'CopyToHDFS',

1,

'Daily',

NULL,

'23:30:00.0000000',

1,

NULL,

NULL);

## Detailed Workflows:

\*\*Visio diagram to be added showing full system workflows from FTDW to Azure blob.\*\*

# Technical Specification:

## CIP001a OFFERS

### i. Raw layer

|  |  |
| --- | --- |
|  | **Details** |
| **Description** | cdd\_raw\_fct\_pricing\_promo\_daily |
| **Job Name** | cdd\_raw\_fct\_pricing\_promo\_daily.py |
| **Job Time/Frequency** | Weekly ((Saturday PM/Sunday AM) |
| **Destination Table Name** | gb\_customer\_data\_domain\_raw. cdd\_raw\_fct\_pricing\_promo\_daily |
| **Refresh Type** | Full |
| **DMD** | <https://confluence.walmart.com/pages/viewpage.action?spaceKey=ASDADMA&title=Q-Promotion+-+CIP0001a+-+cdd_raw_Fct_Pricing_Promo_Daily> |
| **Dependencies** | **FTDW** – DW\_MERCH .dbo. VwFact\_Pricing\_Promo\_Daily **FileExtractor -** select \* from ConfigFileExtractor where FileName = 'QPROMO\_vwFact\_Pricing\_Promo\_Daily' **HadoopFileOperation -** select \* from ftdw\_metaconfig..ConfigHadoopFileOperation where FileName = ''QPROMO\_vwFact\_Pricing\_Promo\_Daily ' |
| **Branch Dependent Objects** | gb\_customer\_data\_domain\_odl.cdd\_odl\_qpromo\_offers\_stg1 |
| **Python Code** | ASDA-DIA/Customer\_Ecosystem/cip/cip/jobs/staging/cdd\_raw\_fct\_pricing\_promo\_daily.py |
| **Output File Location** | NA |

### ii. odl layer

|  |  |
| --- | --- |
|  | **Details** |
| **Description** | cdd\_odl\_qpromo\_offers\_stg1 |
| **Job Name** | cdd\_odl\_qpromo\_offers\_stg1.py |
| **Job Time/Frequency** | Weekly ((Saturday PM/Sunday AM) |
| **Destination Table Name** | gb\_customer\_data\_domain\_raw.cdd\_odl\_qpromo\_offers\_stg1 |
| **Refresh Type** | Full |
| **DMD** | No DMD, staging table for RPT layer, job too big for only RPT |
| **Dependencies** | gb\_customer\_data\_domain\_raw. cdd\_raw\_fct\_pricing\_promo\_daily |
| **Branch Dependent Objects** | gb\_customer\_data\_domain\_raw.cdd\_odl\_qpromo\_offers\_stg2 |
| **Python Code** | ASDA-DIA/Customer\_Ecosystem/cip/cip/jobs/information/cdd\_odl\_qpromo\_offers\_stg1 |
| **Output File Location** | NA |

|  |  |
| --- | --- |
|  | **Details** |
| **Description** | cdd\_odl\_qpromo\_offers\_stg2 |
| **Job Name** | cdd\_odl\_qpromo\_offers\_stg2.py |
| **Job Time/Frequency** | Weekly ((Saturday PM/Sunday AM) |
| **Destination Table Name** | gb\_customer\_data\_domain\_raw.cdd\_odl\_qpromo\_offers\_stg2 |
| **Refresh Type** | Full |
| **DMD** | No DMD |
| **Dependencies** | gb\_customer\_data\_domain\_raw.cdd\_odl\_qpromo\_offers\_stg1 |
| **Branch Dependent Objects** |  |
| **Python Code** | ASDA-DIA/Customer\_Ecosystem/cip/cip/jobs/information/cdd\_odl\_qpromo\_offers\_stg2.py |
| **Output File Location** | NA |

### iii. rpt layer

|  |  |
| --- | --- |
| **---** | **Details** |
| **Description** | gb\_customer\_data\_domain\_rpt.cdd\_rpt\_qpromo\_offers |
| **Job Name** | cdd\_rpt\_qpromo\_offers.py |
| **Job Time/Frequency** | Weekly ((Saturday PM/Sunday AM) |
| **Destination Table Name** | gb\_customer\_data\_domain\_rpt.cdd\_rpt\_qpromo\_offers |
| **Refresh Type** | Full |
| **DMD** | Cust Ecosys – CSM00-- |
| **Dependencies** |  |
| **Branch Dependent Objects** | Extraction – offers.parquet |
| **Python Code** | ASDA-DIA/Customer\_Ecosystem/cip/cip/jobs/reporting/rpt\_qpromo\_offers.py |
| **Output File Location** | NA |

### iv. Output to CIP

|  |  |
| --- | --- |
| **Python Code** | ASDA-DIA/Customer\_Ecosystem/cip/cip/jobs/extraction/qpromo\_offers.py |
| **Output File Location** | /user/svc\_uk\_cust\_rdl/extracts/qpromo |

## CIP001b

### Raw layer

### Odl layer

### Rpt layer

### Output to CIP

## CIP001d Anaplan

### i.Raw layer

### ii. Odl layer

### Iii rpt layer

### iv. Output to CIP

## CIP002 Asda Reward Offers

### Raw layer

### Odl layer

### Rpt layer

### Output to CIP

## CIP003a Events

### Raw layer

### Odl layer

### Rpt layer

### Output to CIP

## CIP003b Local Events

### Raw layer

### Odl layer

### Rpt layer

### Output to CIP

## Data Extracts from Fast Track > RDL

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Feed** | **Source System** | **Source Db** | **Source Table** | **Local File Share** | **Extract Name** | **File Del.** | **HDFS Path** | **PySpark Job** | **Hive Table (raw layer)** | **Hive Table**  **(odl layer)** | **Hive Table**  **(rpt layer)** | **Freq.** | **Extract Type** |
| Calendar | FTDW | LDM\_PROD\_SQLServer | vwDailySales\_Datamart\_Dim\_Calendar | \\ftdw\CommercialTransformation\Dev\CSM\ | vwDailySales\_Datamart\_Dim\_Calendar.csv | | | /user/svc\_uk\_cust\_rdl\_dev/sourceFiles/ftdw/dim\_calendar/ | csm\_dim\_calendar.py | cdd\_raw\_dim\_calendar |  | cdd\_rpt\_dim\_calendar | Yearly | Full |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |

## Dependancies

\*\*Visual guide to dependencies to be added \*\*

# Output Specification:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Output Name** | **Database** | **Table** | **Frequency** | **Size** | **Azure Path** |
| --.parquet |  |  |  |  |  |
| --.parquet |  |  |  |  |  |
| --.parquet |  |  |  |  |  |
| --.csv |  |  |  |  |  |

# 6 PySpark Jobs:

# Three main steps:

* Load the data to data frame (Either from file or Hive DB)
* Apply schema / transformations
* Store the data frame to the hive as a Parquet file.

# Orchestration:

Automic workflow details

# Frequency:

Workflow to be triggered on Saturday evening and run overnight. Jobs to be checked on Sunday AM. Manual push of data feeds to blob storage on Sunday AM.

# Security Control:

Extracts pulled via metadata table using service account (automated)

Loaded to Hadoop using service account (automated)

Workflows in automic scheduled by those with access (manual trigger of workflow)

Push to blob storage via service account (manual push)