**BI Solution Design**

Business Requirements

Report which presents the merchants information from Platform 1 and Platform 2

Data Source

1. **Platform 1**

Server: Microsoft SQL Server

Database: Platform\_1

Table: Platform\_1\_Merchant

1. **Platform 2**

Server: PostgreSQL

Database : Platform\_2

Table : Platform\_2\_Merchant

Architecture

The design will be based on Microsoft Power BI application

Visualisation

Dataset

Dataflow

DF\_P1

DF\_P2

MSSQL

Report

DF\_P1

PostgreSQL

DF\_P2

**Dataflow**

There will be 2 data flows with each of the data flow will be created from each data sources

|  |  |  |
| --- | --- | --- |
| Data flows | Description | SQL Statement to Import |
| DF\_P1 | Data flow task to extract the information from MS SQL for Merchant information in Platform 1 | SELECT  tradingname,  email,  billingstatus,  countrycode,  customers,  Platform = ‘Platform 1’  FROM Platform\_1\_Merchant |
| DF\_P2 | Data flow task to extract the information from PostgreSQL for Merchant information in Platform 2 | SELECT  TradingName,  Email,  BillingStatus =  (SELECT BillingStatus FROM Platform\_2\_Billing as B  WHERE M.BillingStatusID = B.BillingStatusID ),  CountryCode =  (SELECT CountryCode FROM Platform\_2\_Country as C  WHERE M.CountryCodeID= C. CountryCodeID),  CustomerCount,  Platform = ‘Platform 2’  FROM Platform\_2\_Merchant as M |

**Note:**

**SQL subquery is needed in DF\_P2 to transform the CountryCodeID and BillingStatusID into readable data**

Further transformations are performed in the dataflow:

* To standardise the name of the fields in both DF\_P1 and DF\_P2

|  |  |  |
| --- | --- | --- |
| **Standardised field name** | **DF\_P1 field name** | **DF\_P2 field name** |
| MerchantName | tradingname | TradingName |
| MerchantEmail | email | Email |
| BillingStatus | billingstatus | BillingStatus |
| CountryCode | countrycode | CountryCode |
| CustomerCount | customers | CustomerCount |
| Platform | Platform | Platform |

* To ensure the CountryCode data in both dataflows are in same type and format i.e. if data in DF\_P2 is “inactive”, perform “Replace Value” under Transform ribbon to “INACTIVE”
* To ensure the BillingStatus data in both dataflows are in same type and format format i.e. if data in DF\_P2 is “nz”, perform “Replace Value” under Transform ribbon to “NZ”

Upon standardisation completed, each data flow will be saved and publish for further usage

**Dataset**

The task needed is to create a single dataset for visualisation and report creation. From the available data flow DF\_P1 and DF\_P2, it will be used as source for the newly created Merchant dataset named

DS\_Merchant.

The process for the creation of the dataset is:

1. Create a new dataset called DS\_Merchant
2. Import DF\_P1 and DF\_P2 from the source
3. Append the source into single table Merchant
4. Right click and uncheck Enable Load to source DF\_P1 and DF\_P2 to hide the source tables
5. The dataset is saved through clicking the Apply and Close button

**Visualisation**

Before creation of visualisation, dataset DS\_Merchant is import into the Data Model of the new report. As this will be a single table, no relationship is required for the dataset structure.

Required DAX Measures:

1. SumCustomers = sum([CustomersCount])
2. SumMerchant = sum([MerchantName])

Recommended visualisation

|  |  |  |  |
| --- | --- | --- | --- |
| Visual | Type | Fields | Remark |
| Card 1 | Card | SumCustomers | Represent customers numbers |
| Card 2 | Card | SumMerchant | Represent merchant numbers |
| Summary | Matrix | All fields | Represent all information on the merchants |
| Merchant by Country | Maps | SumMerchant  CountryCode | Represent demography of the merchant |
| Customers by Platform | Pie Chart | SumCustomers  Platform | Represent customers breakdown by platform |