

**SRI LANKA INSTITUE OF INFORMATION TECHNOLOGY (SLIIT)**

**IT3021 – DATA WAREHOUSE AND BUSINESS INTELLIGENCE**

ASSIGNMENT – 02

**S.Z. RAEESUL ISLAM**

**IT19991986**

Table of Contents

[**Figures of Contents** 3](#_Toc94188852)

[**1.** **Data Source** 4](#_Toc94188853)

[**2.** **SSAS Cube Implementation** 6](#_Toc94188854)

[**2.1. Cube Creation** 6](#_Toc94188855)

[**2.2.** **Data Source View** 7](#_Toc94188856)

[**2.3. Hierarchies** 8](#_Toc94188857)

[**2.3.1. Location Hierarchy** 8](#_Toc94188858)

[**2.3.2. Date Hierarchy** 8](#_Toc94188859)

[**2.4. KPI Values** 9](#_Toc94188860)

[**2.5. Cube Deployment** 11](#_Toc94188861)

[**3.** **Demonstration of OLAP Operations** 12](#_Toc94188862)

[**3.1.** **Roll-up** 12](#_Toc94188863)

[**3.2.** **Drill-down** 12](#_Toc94188864)

[**3.3.** **Slice** 15](#_Toc94188865)

[**3.4.** **Dice** 17](#_Toc94188866)

[**3.5.** **Pivot** 19](#_Toc94188867)

[**4.** **SSRS Reports** 20](#_Toc94188868)

[**4.1.** **Report with Matrix** 21](#_Toc94188869)

[**4.2.** **Report With Single Parameter** 22](#_Toc94188870)

[**4.3.** **Report with Multiple Parameters** 24](#_Toc94188871)

[**4.4.** **Drill-Down Report** 26](#_Toc94188872)

[**4.5.** **Drill-Through Report** 28](#_Toc94188873)

[**5.** **References** 31](#_Toc94188874)

# **Figures of Contents**

[Figure 1. Snowflake Schema for Data Warehouse 4](#_Toc91528599)

[Figure 2. Data Warehouse Snapshot 5](#_Toc91528600)

[Figure 3. Cube Creation Solution Explorer 6](#_Toc91528601)

[Figure 4. Cube Structure 7](#_Toc91528602)

[Figure 5. Location Hierarchy 8](#_Toc91528603)

[Figure 6. Date Hierarchy 8](#_Toc91528604)

[Figure 7. KPI Measure Amount 9](#_Toc91528605)

[Figure 8. KPI Measure Balance 10](#_Toc91528606)

[Figure 9. Cube Deployment Successfully 11](#_Toc91528607)

[Figure 10. Roll-up Pivot Table 13](#_Toc91528608)

[Figure 11. Roll-up Pivot Chart 13](#_Toc91528609)

[Figure 12. Drill-down Pivot table 14](#_Toc91528610)

[Figure 13. Drill-down Pivot Chart 14](#_Toc91528611)

[Figure 14. Pivot Table and Slicer 15](#_Toc91528612)

[Figure 15. Pivot Chart for Slicing Operation 16](#_Toc91528613)

[Figure 16. Pivot Table and Slicers for Dicing Operation 17](#_Toc91528614)

[Figure 17. Pivot Chart and Slicer for Dicing Operation 18](#_Toc91528615)

[Figure 18. Pivot Table 19](#_Toc91528616)

[Figure 19. Change the Angle of the Pivot Table 19](#_Toc91528617)

[Figure 20. SSRS Matrix Report 21](#_Toc91528618)

[Figure 21. Selection of Loan Purpose 22](#_Toc91528619)

[Figure 22. SSRS Single Parameter Report 22](#_Toc91528620)

[Figure 23. SSRS Single Parameter Report with Select All 23](#_Toc91528621)

[Figure 24. Region Selection 24](#_Toc91528622)

[Figure 25. State Selection 24](#_Toc91528623)

[Figure 26. SSRS Multiple Parameter Report 24](#_Toc91528624)

[Figure 27. SSRS Multiple Parameter All Selected Report 25](#_Toc91528625)

[Figure 28. SSRS Report before Drill-down 26](#_Toc91528626)

[Figure 29. SSRS Report After Drilled down 27](#_Toc91528627)

[Figure 30. Level – 1 SSRS Report for Drill Through 28](#_Toc91528628)

[Figure 31. South Region Drilled Through 29](#_Toc91528629)

[Figure 32. Year 1993 Drilled Through 29](#_Toc91528630)

# **Data Source**

The data source used for the analysis purpose is ‘**CreditCardAnalysis\_DW**,’ which was developed in the **assignment 1**. This data warehouse was created based on a data set from a bank in Czech Republic to mine and analyze this bank data in order to extrapolate from it the type of customer who makes a good candidate for a credit card. The data set has been modified to develop a scenario that meets the requirement of the assignment. Its features allow viewing a transaction from multiple dimensions, from Accounts, Date, Permanent Order, Client, Loan, District and Disposition.

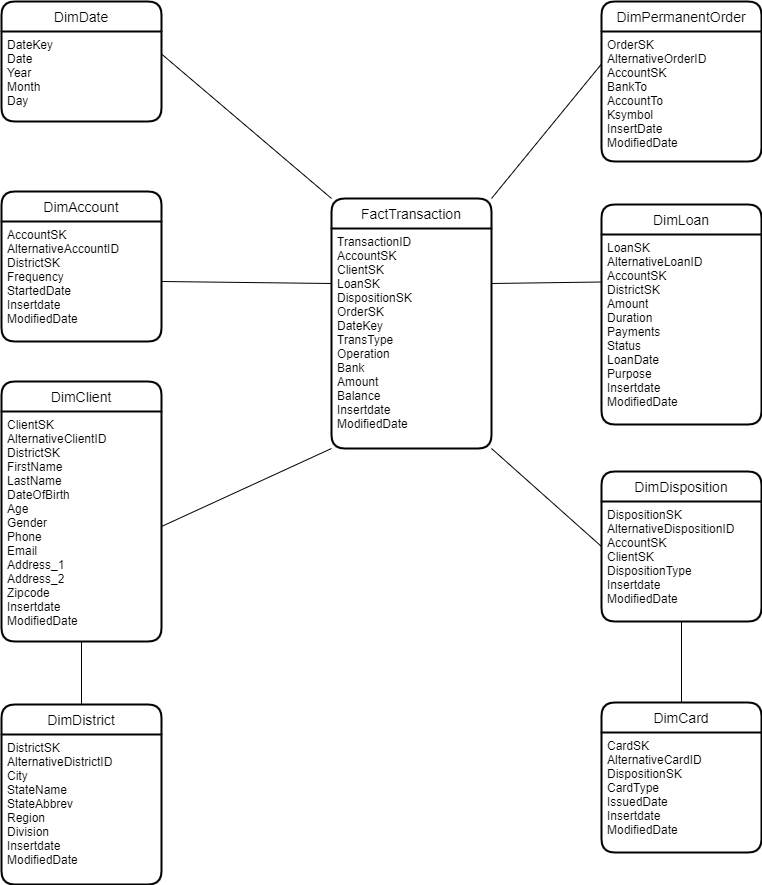
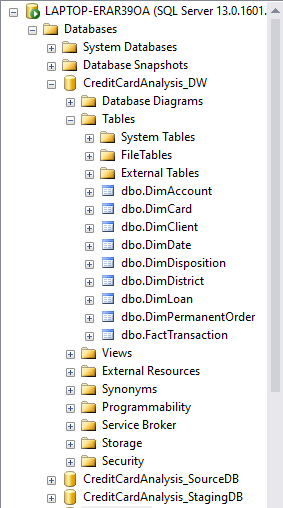


Figure 1. Snowflake Schema for Data Warehouse

The Data Warehouse design was implemented using the **Snowflake Schema**. Snowflake Schema is an extension of **Star Schema** and consists of some dimensions that are normalized. According to the schema above, there are **8 Dimensions** and **1 Fact table**.

**Assumption:**

**Client** Dimension is considered as a **Slowly Changing Dimension (SCD)**



**Dimension Tables:**

1. DimAccount
2. DimCard
3. DimClient
4. DimDate
5. DimDisposition
6. DimDistrict
7. DimPermanentOrder
8. DimLoan

**Fact Table:**

1. FactTransaction

Figure 2. Data Warehouse Snapshot

# **SSAS Cube Implementation**

**OLAP** **Cube** is a method for storing data in Multidimensional Forms. It will allow to analyze a multidimensional data from multiple perspectives. The advantage of using a cube is that it pre-calculates most of the queries, that is time consuming to execute over relational tables that contains joins and aggregates. The main components of the cube are:

1. **Dimensions**: Define the structure of the cube that is used for OLAP operations.

2. **Measures**: Provide aggregated numeric values of interest to the end user.

## **2.1. Cube Creation**

* As the first step an analysis service project in the name ‘**CreditCardAnalysis\_SSAS**’ was created was the data source was configured in order to extract data to the cube.
* A data source view ‘**DSV\_CreditCardAnalysis**’ was created and all necessary table links were created.
* A cube named ‘**CreditCardAnalysis\_Cube**’ was created, by selecting the necessary measures. Then necessary attributes and hierarchies were added before the deployment of the cube.

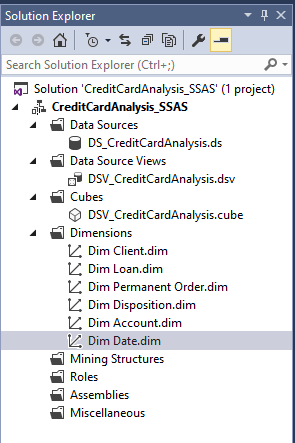


Figure 3. Cube Creation Solution Explorer

## **2.2.** **Data Source View**

* It represents the cube structure, measures, and dimensions.

Diagram

Description automatically generated

Figure 4. Cube Structure

## **2.3. Hierarchies**

* **Hierarchies** are a useful tool in SSAS to reduce complexity between attributes and guide users into a certain drill-down behavior.

### **2.3.1. Location Hierarchy**

* The higher level is the **Region**, which contains multiple **State**, and the States contain multiple **Cities**

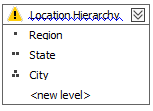


Figure 5. Location Hierarchy

### **2.3.2. Date Hierarchy**

* The higher level is the **Year**, which then is followed by the lower levels **Quarter**, **Month**, and **Date**.

Text

Description automatically generated

Figure 6. Date Hierarchy

## **2.4. KPI Values**

In **SQL Server Analysis Services**(SSAS), add **Key Performance Indicators** (KPIs) can be added to our database cube in order to evaluate business performance, as reflected in the cube data. A KPI is associated with a measure group and is made up of a set of calculations. Typically, the calculations are a combination of calculated members and Multidimensional Expressions (MDX) statements.

Relevant KPI’s used:

1. **KPI Amount**: Total amounts

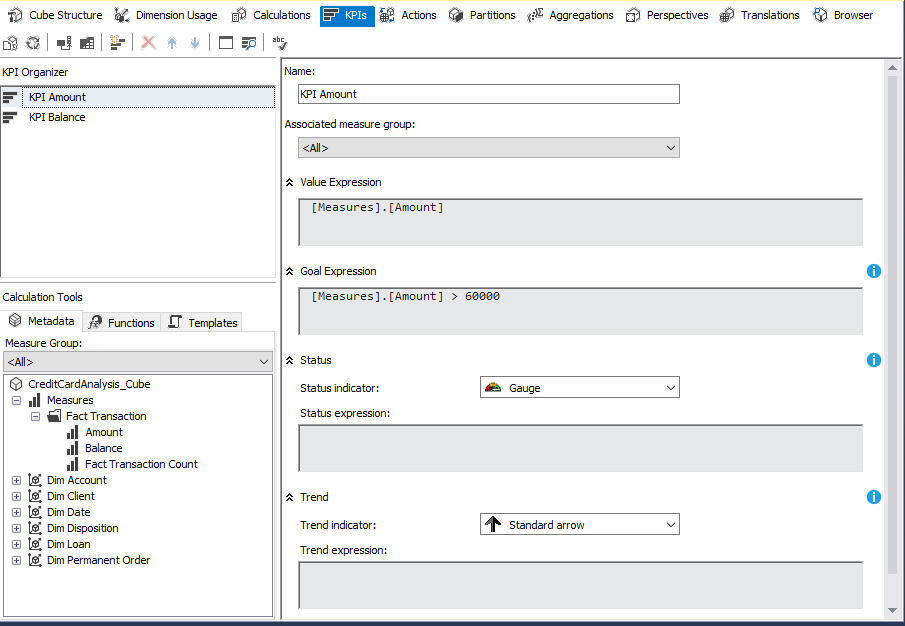


Figure 7. KPI Measure Amount

1. **KPI Balance**: Total balances

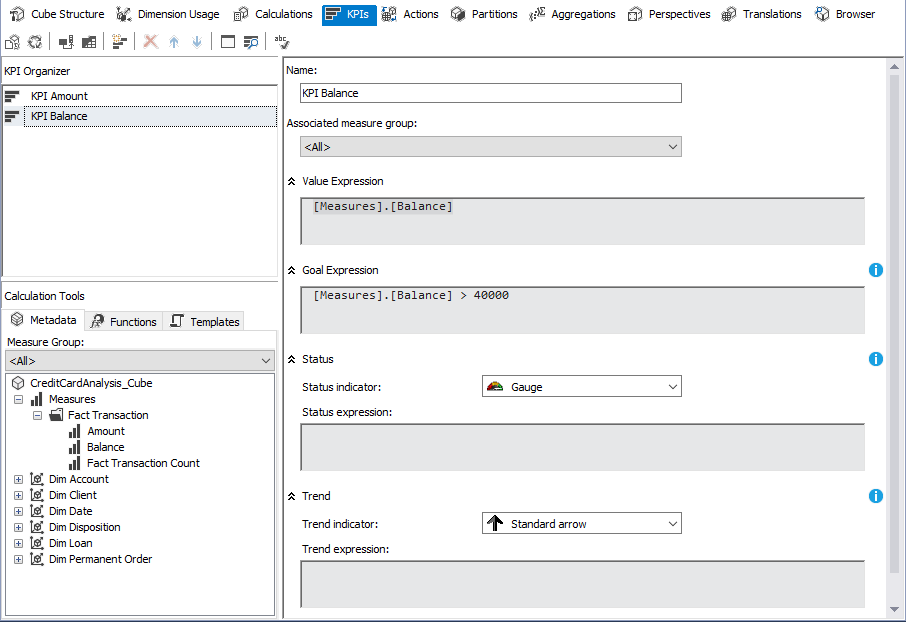


Figure 8. KPI Measure Balance

## **2.5. Cube Deployment**

* After setting all attributes, hierarchies and KPI’s, finally the cube was deployed.

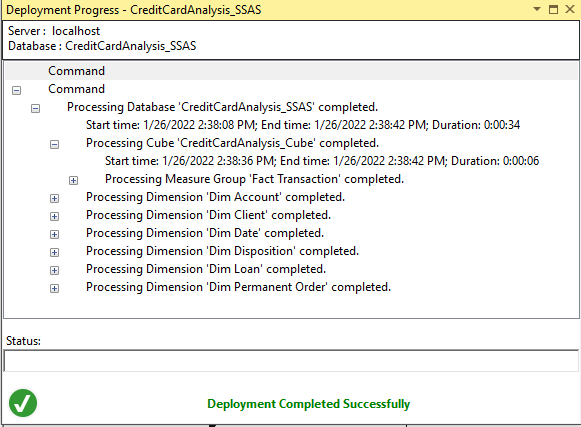


Figure 9. Cube Deployment Successfully

# **Demonstration of OLAP Operations**

* **OLAP** operation is an important part of **Business Intelligence**, that provides powerful capabilities for data mining and trend analysis. They are capable of solving problems in both business and IT departments. OLAP helps to analyze big data amounts from different perspectives rapidly.
* **MDX** query can be used to connect to the excel workbooks to get data to the semantic layer for respective demonstrations. This method needs to build up MDX query through SSAS project by browsing data. The same can be done using ‘data’ tab in excel. This will enable the connection with the whole set of facts and dimension tables. For the assignment purpose, the second approach of connecting to the entire data set was used.
* Five analytical operations can be performed using OLAP:

1. Roll-up
2. Drill-down
3. Slice
4. Dice
5. Pivot

## **Roll-up**

* **Roll-up** is also known as ‘consolidation’ and ‘aggregation,’ which can be performed in two ways:
  1. Reducing dimensions
  2. Climbing up a dimension hierarchy

## **Drill-down**

* **Drill-down** is the opposite of roll-up. It means to step down a hierarchy, which will enable navigation through details of a dimension. This operation fragments data into smaller parts. It can be done via:

1. Moving down a hierarchy
2. Increasing the dimension.

**Report 1:**

* **Total loan Amounts** and **Balances** based on **Roll-ups** and **Drill-down** of **Region** and **State**. The report displays the **Total Loan Amount**, **Total Account Balances**, and the achievement of the **KPI goal** for each **Year**, based on the **Roll-ups** and **Drill-downs** of **Regions**, **State** and **Cities**.

Roll-Up (States rolled-up to Regions)

Figure 10. Roll-up Pivot Table

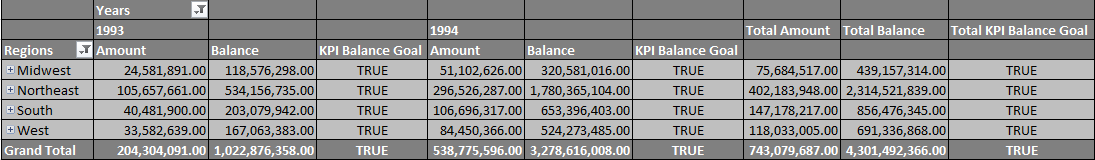
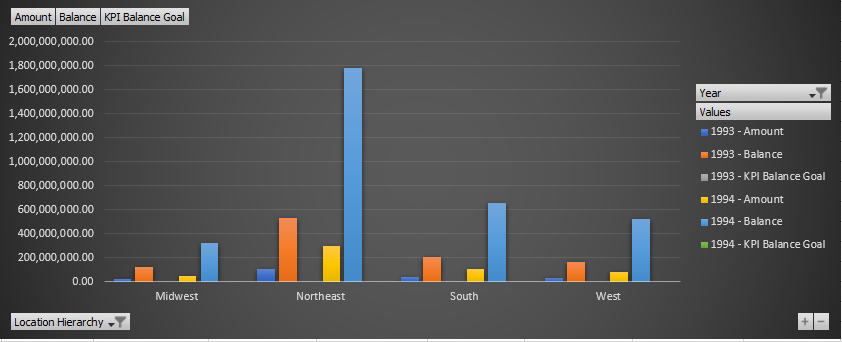


Figure 11. Roll-up Pivot Chart



* The **Pivot** **table** and the **Pivot** **chart** show the **Total Loan Amounts** and the **Balances** for all the rolled-up **Regions**, according to each **Year**.

Drill-down(Regions have been drilled down to States)

Figure 12. Drill-down Pivot table

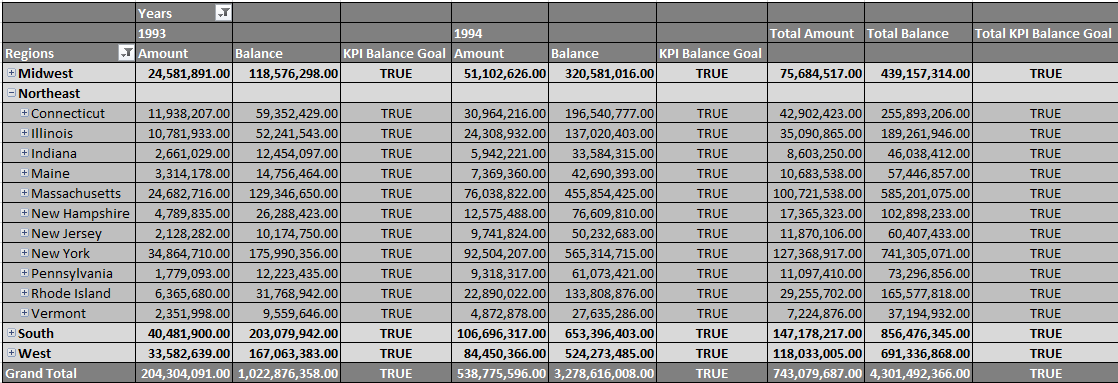
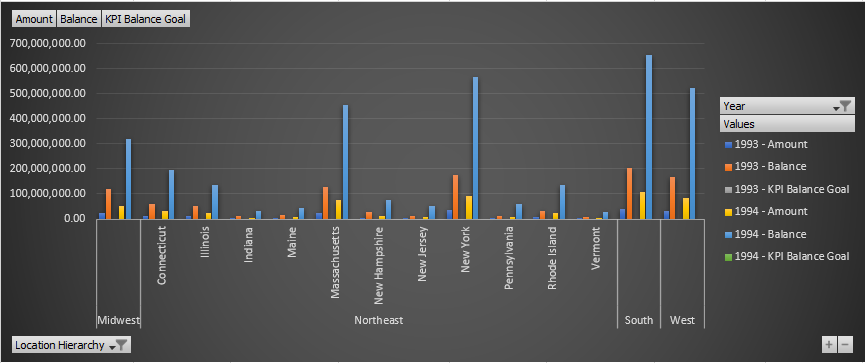


Figure 13. Drill-down Pivot Chart



* The **Pivot table** and **Pivot chart** represent the **Total Loan Amounts** and **Balances** for the **drilled** **downed** States of the Region ‘**Northeast**,’ for each year.

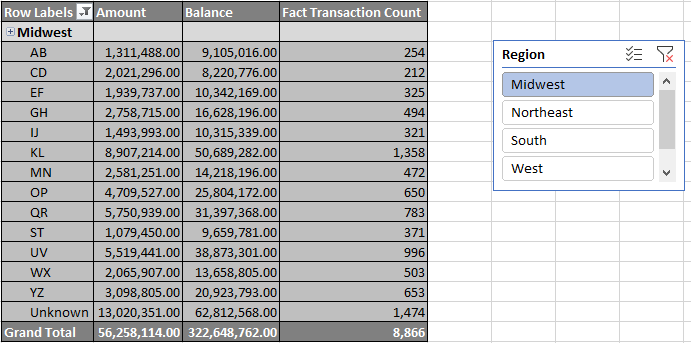
## **Slice**

* **Slice** create a rectangular subset of the cube, by selecting a single value for one of its dimensions. A slice function is much like a report or a query that it returns data based on a request for what to see.

**Report**:

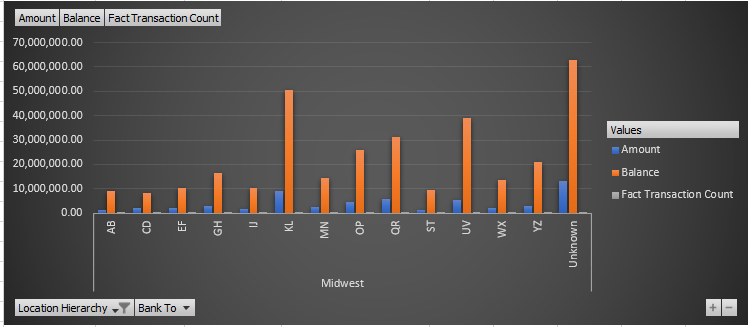
* + **Total Loan Amounts, Balances, Number of Transactions** and the **KPI goals** based on **Regions** and the **Banks**. The report represents the **Total Loan Amounts, Balances**, **Number of Transactions**, and the achievement of the **KPI goals** based on a selected **Region** and the **Banks** belonging to the **Region**.

Figure 14. Pivot Table and Slicer



* The **Pivot table** displays **Bank** wise **Total Loan Amounts**, **Balances**, and the **Number of** **Transactions**, based on the **slicing** of Region ‘**Midwest’**

Figure 15. Pivot Chart for Slicing Operation



* The **Pivot chart** displays the **Bank** wise **Total Loan Amounts, Balances**, and the **Number of Transactions,** based on the **slicing** of Region ‘**Midwest’**

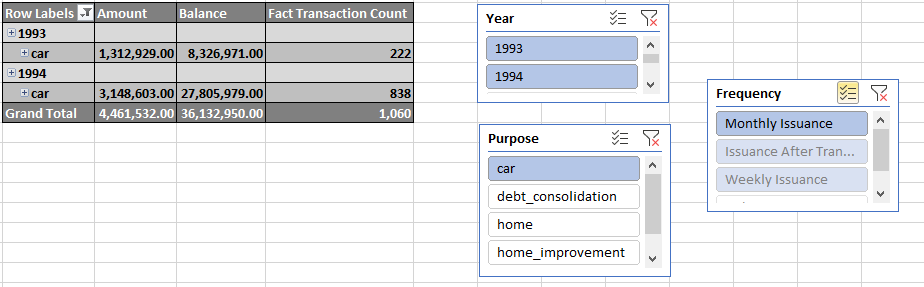
## **Dice**

* **Dice** operation selects two or more dimensions from a cube, and results in a sub cube by selecting specific values on those selected dimensions. Dicing on the other hand, is more of a zoom feature that selects a subset over all the dimensions, but for specific values of the dimension.

**Report**:

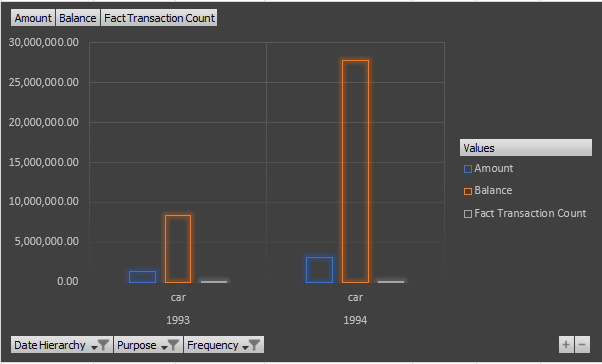
* + **Total Loan Amount**, **Balances** and **Number of Transactions** based on **Loan Purpose, Year,** and **Frequency of the Loan**. The report represents the **Total Loan Amounts, Balances, and Number of Transactions**, that could be gathered for a **selected Year, selected Purpose of the Loan, and a selected Frequency of Loans.**

Figure 16. Pivot Table and Slicers for Dicing Operation



* The **Pivot table** represents **the Total Loan Amounts, Balances, the Number of Transactions** for the selected years ‘**1993** and **1994’**, selected **Loan Purpose** ‘**car’** and the **Frequency ‘Monthly Insurance’**

Figure 17. Pivot Chart and Slicer for Dicing Operation



* The **Pivot chart** represents the **Total Loan Amounts**, **Balances**, the **Number of Transactions** for the selected Years ‘**1993** and **1994’**, selected Loan Purpose ‘**car’** and the Frequency ‘**Monthly Issuance**.’

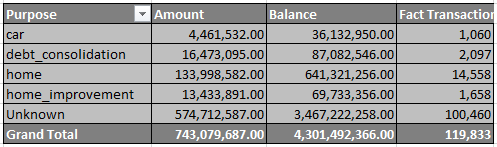
## **Pivot**

* **Pivot** operation provides a new perspective to the cube by rotating the data axes of the cube. It may contain swapping the rows and columns or moving one of the Row dimensions into the column dimensions.

**Report**:

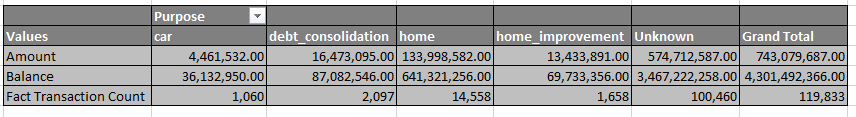
* + **Loan Amounts, Balances** and **Number of Transactions** based on **Loan Purpose**

Figure 18. Pivot Table



* The report with **Loan Purpose** as the rows and the **Total Loan Amounts, Balances, and the Number of Transaction** as columns.

Figure 19. Change the Angle of the Pivot Table



* The report has now **changed** the perspective, as **the Total Loan Amounts, Balances and the Number of Transactions** is transposed to rows and the **Loan Purpose** to columns.

# **SSRS Reports**

**SQL Server Reporting Services** (SSRS) is a reporting software that allows you to produce formatted reports with tables in the form of data, graph, images, and charts. These reports are hosted on a server that can be executed any time using parameters defined by the users. SSRS reports can be developed using tools like report builder and deployed in SSRS Web portal for viewing.

**Report Builder:**

* + It is a standalone application for creating paginated reports. Once designed the report can be deployed to the report server and displayed on the SSRS web portal.

**SSRS Web Portal:**

* + The SQL Server Report Service Web Portal is a web-based experience. In this portal users can view reports, KPIs and navigate through the elements in the report server instance. User can also use the web portal to administer a single report server instance.

Graphical user interface, text, application

Description automatically generated

Figure 20. SSRS Web Portal Home Page

## **Report with Matrix**

* In SSRS, Matrix is very similar to a table, but it is configured to show data grouped by columns and rows, with aggregate data at the intersection.

**Report**:

* + **Total Loan Amount and Account balances**. The report contains the data of **Total Loan Amounts and Balances for each Order Type for each Year**.

Table

Description automatically generated

Figure 21. SSRS Matrix Report

SQL Query:

select dd.Year, po.OrderType, sum(ft.Amount) as 'Total Amount', sum(ft.Balance) as 'Total Balance'

from FactTransaction ft

inner join DimDate dd

on ft.DateSK = dd.DateKey

inner join DimPermanentOrder po

on ft.OrderSK = po.OrderSK

group by dd.Year, po.OrderType

## **Report With Single Parameter**

* In SSRS, allows Parameter us to pass input value to the report. Also, it offers ‘Select All’ option that helps to select all parameter values.

**Report:**

* + **Loan Purpose** wise **Total Loan Amount**.

Graphical user interface, text, application

Description automatically generated

Figure 22. Selection of Loan Purpose

Table

Description automatically generated

Figure 23. SSRS Single Parameter Report

Table

Description automatically generated

Figure 24. SSRS Single Parameter Report with Select All

SQL Query:

select ac.Frequency, dd.Year, ln.Purpose, sum(ft.Amount) as ' Total Amount'

from FactTransaction ft

inner join DimAccount ac

on ft.AccounSK = ac.AccountSK

inner join DimLoan ln

on ft.LoanSK = ln.LoanSK

inner join DimDate dd

on ft.DateSK = dd.DateKey

where ln.Purpose in (@Purpose)

group by ac.Frequency, dd.Year, ln.Purpose

-- Getting Loan Purpose --

select distinct(Purpose)

from DimLoan

## **Report with Multiple Parameters**

* In SSRS, Multiparameter-values allows us to pass either one or more input parameter values to the report. Also, it offers a “Select All” option that helps to select all parameter values.

**Report:**

* + **Region and State wise Total Amount and Balances**. The report allows to select the Region and State through a drop down. When the Regions are selected, the States belonging to the particular Region will be filtered and allowed for selection. On selection of view report, the report displays the Total Loan Amounts and the Total Balances for each Year, grouped according to type of Order, and the selected Region and States accepted as parameters.

Graphical user interface, application

Description automatically generatedGraphical user interface, text, application

Description automatically generated

Figure 25. Region Selection

Figure 26. State Selection

Graphical user interface, table

Description automatically generated

Figure 27. SSRS Multiple Parameter Report

Graphical user interface, table

Description automatically generated

Figure 28. SSRS Multiple Parameter All Selected Report

SQL Query:

select dt.Region, dt.State, dt.City, dd.Year, po.OrderType, sum(ft.Amount) as ' Total Amount',

sum(ft.Balance) as ' Total Balance'

from FactTransaction ft

inner join DimClient cl

on ft.ClientSK = cl.ClientSK

inner join DimDistrict dt

on cl.DistrictSK = dt.DistrictSK

inner join DimDate dd

on ft.DateSK = dd.DateKey

inner join DimPermanentOrder po

on ft.OrderSK = po.OrderSK

where dt.State in (@State)

group by dt.Region, dt.State, dt.City, dd.Year, po.OrderType

-- Getting State --

select dt.Region, dt.State, ac.DistrictSK, ac.AccountSK, ft.AccounSK

from DimDistrict dt

inner join DimAccount ac

on dt.DistrictSK = ac.DistrictSK

inner join FactTransaction ft

on ac.AccountSK = ft.AccounSK

where dt.Region in (@Region)

-- Region List --

select distinct(Region)

from DimDistrict

-- State List --

select distinct(State)

from DimDistrict

where Region in (@Region)

## **Drill-Down Report**

* In SSRS reports, Drill-down allows expand or collapse a section of a report to show or hide detail data. We can expand the data using the plus button and collapse data using the minus button.

**Report:**

* + **Drill down through Region, States, and Cities**. The feature allows a user to view Total Loan Amounts and Total Balances for each Year based on Region, State, and Cities. Initially the Totals for a Region will be displayed, which must then be expanded to view the same for a State and a City.

Table

Description automatically generated

Figure 29. SSRS Report before Drill-down

Table

Description automatically generated

Figure 30. SSRS Report After Drilled down

SQL Query:

select dt.Region, dt.State, dt.City, dd.Year, sum(ft.Amount) as ' Total Amount', sum(ft.Balance) as ' Total Balance', count(ft.TransactionID) as 'No of Transaction'

from FactTransaction ft

inner join DimClient cl

on ft.ClientSK = cl.ClientSK

inner join DimDistrict dt

on cl.DistrictSK = dt.DistrictSK

inner join DimDate dd

on ft.DateSK = dd.DateKey

group by dt.Region, dt.State, dt.City, dd.Year

## **Drill-Through Report**

* In SSRS, a drill through allows a user click on a link or an area in a chart with summarized data, which then opens a separate, related report to show detailed data. Drill through reports commonly contain details about an item that is contained in an original summary report. The data in the drill through report is not retrieved until the user clicks the link in the main report.

**Report:**

* + **Level 1** – Year and Region wise Loan Amounts and Balances L1
  + **Level 2** – State wise Total Amounts and Balances report 2 & Year wise drill through.

This report displays two column charts. 1st chart represents the data of Total Loan Amounts and Total Balances for each **Region**. 2nd chart represents the Total Loan Amounts and Total Balances for each **Year**.

Graphical user interface, application

Description automatically generated

Figure 31. Level – 1 SSRS Report for Drill Through

* When the user clicks on a bar in the report contain Regions, it will display a detailed report that provides the same information based on each State belonging to the Region. The below diagram shows details when the state ‘**South’** is clicked.

A picture containing graphical user interface

Description automatically generated

Figure 32. South Region Drilled Through

* When the user clicks on a bar in the report containing Years, it will display a detailed report that provides the Total Loan Amount and Balance Based on Loan Purpose and Total Loan Amount and Balance Based on Quarter in that particular year. The below diagram shows details when the year ‘**1993’** is clicked.

Graphical user interface

Description automatically generated

Figure 33. Year 1993 Drilled Through

SQL Query:

select dt.Region, dt.State, dt.City, dd.Year, sum(ft.Amount) as ' Total Amount',

sum(ft.Balance) as ' Total Balance'

from FactTransaction ft

inner join DimClient cl

on ft.ClientSK = cl.ClientSK

inner join DimDistrict dt

on cl.DistrictSK = dt.DistrictSK

inner join DimDate dd

on ft.DateSK = dd.DateKey

group by dt.Region, dt.State, dt.City, dd.Year

-- State wise Report --

select dt.State, sum(ft.Amount) as ' Total Amount', sum(ft.Balance) as ' Total Balance',

count(ft.LoanSK) as 'No of Loans'

from FactTransaction ft

inner join DimClient cl

on ft.ClientSK = cl.ClientSK

inner join DimDistrict dt

on cl.DistrictSK = dt.DistrictSK

where dt.Region in (@Regions)

group by dt.State

-- Year wise Report --

select dd.Quarter, ln.Purpose, sum(ft.Amount) as ' Total Amount', sum(ft.Balance) as ' Total Balance'

from FactTransaction ft

inner join DimLoan ln

on ft.LoanSK = ln.LoanSK

inner join DimDate dd

on ft.DateSK = dd.DateKey

where dd.Year = (@Years)

group by dd.Quarter, ln.Purpose

# **References**

[1] [https://www.sqlshack.com/analysis-services-ssas-multidimensional-design-tips-relationshierarchies/#:~:text=sample%20OLAP%20cube.,Hierarchies,a%20certain%20drill%2Ddown%2 0behavior](https://www.sqlshack.com/analysis-services-ssas-multidimensional-design-tips-relationshierarchies/%23:~:text=sample%20OLAP%20cube.,Hierarchies,a%20certain%20drill%2Ddown%252%200behavior%20)

[2] [https://www.red-gate.com/simple-talk/databases/sql-server/bi-sql-server/adding-a-kpi-to-ansql-server-analysis-servicescube/#:~:text=In%20SQL%20Server%20Analysis%20Services,of%20a%20set%20of%20calcul ations.](https://www.red-gate.com/simple-talk/databases/sql-server/bi-sql-server/adding-a-kpi-to-ansql-server-analysis-servicescube/%23:~:text=In%20SQL%20Server%20Analysis%20Services,of%20a%20set%20of%20calcul%20ations.%20)

[3] <https://www.guru99.com/online-analytical-processing.html>

[4][https://www.google.com/search?q=SSRS&oq=SSRS&aqs=chrome..69i57j0i271l3j69i61j69i65j6 9i60l2.697j0j7&sourceid=chrome&ie=UTF-8](https://www.google.com/search?q=SSRS&oq=SSRS&aqs=chrome..69i57j0i271l3j69i61j69i65j6%209i60l2.697j0j7&sourceid=chrome&ie=UTF-8%20)

[5] "SSRS - How to build Drill-Through Report in SSRS," <https://www.youtube.com/watch?v=sPl-Zp0X5Pk&feature=youtu.be>

[6] Working with Multi-Valued Parameters in SSRS, <https://www.interfacett.com/blogs/usingparameters-with-multiple-values-in-sql-server-reporting-services/>