

April 29, 2022

Business Intelligence Project



Stefan Lupu 285057

Supervisor: Io Odderskov IOOD@via.dk

Software Technology Engineering

7th Semester - BUI

29.04.2022

Table of Contents

Business Intelligence Project	1
Table of Contents	2
Chapter 1	4
Introduction	4
Requirements	5
Facts and Dimensions	6
Cube	8
Create/Deploy/Browse a cube in Analysis Services (SSAS)	9
Chapter 2	11
Bringing in data from SQL Server tables	11
Building a basic column chart	13
Creating a slicer	14
Connect to an existing SSAS-model with Power BI	15
Viewing the data underlying a chart	16
Add a second table from an existing data source	17
Use a table visualization	18
Work with a matrix visualization	19
Use Card and Multi-row Card Visualizations	20
Work with pie charts and donut charts	21
Begin a page and a visualization from the ribbon	22
Work with a bubble map	23
Chapter 3	24
Create a hierarchy in one or more of the dimensions	24
Create named calculations in the fact table and dimension	26
Add Calculations to the Cube	28
Create a KPI (Key Performance Indicator)	29
Create Multiple Partitions from a Single Table and Add them to the Cube	30
Limit access to cube members	32
Chapter 4	33
Create Tabular BI Semantic Model	33
Import SQL Server tables into the model	35
New Measure / New Column / Hide Column	37
Create a KPI	39
Deploy Tabular Model Project	40

Chapter 1

Introduction

Fictional company Wide World Importers - *WWI* buys goods from suppliers including novelty and toy manufacturers, and other novelty wholesalers. They stock the goods in their *WWI* warehouse and reorder from suppliers as needed to fulfill customer orders. They also purchase large volumes of packaging materials, and sell these in smaller quantities as a convenience for the customers.

Recently *WWI* started to sell a variety of edible novelties such as chili chocolates. The company previously did not have to handle chilled items. Now, to meet food handling requirements, they must monitor the temperature in their chiller room and any of their trucks that have chiller sections.

The overall purpose of Business Intelligence is to support and facilitate the present or future business decisions in the sense of: efficiency improvements, gain new insights, sales insights, competitive advantage and not in the last order the access to the analytics of the real-time data. In order to achieve these goals it is mandatory to inspect the requirements imposed by the customer which is Wide World Importers in this case. BI makes it possible to merge different data from distinct sources, analyze the information into a suitable format, and then communicate the information to relevant stakeholders. This allows businesses to see the big picture and make smart business decisions.

Requirements

Main purpose of the project is to satisfy the all the requirements that are provided by the customer, in this case is the Wide World Importers and also all the data and analytics made on the provided data must be in an interactive environment so all the business users (*CEO, Sales Manager, Order Manager, Marketing Manager, Finance Manager etc.*) could interact with it in a convenient format. Apart from the format that must be provided there are few requirements related to the data that needs to be operated. Sales data provided by the customer must be used as the core of the application and then all the analytics must be made related to the sales data using the additional information from the data that describes the core facts which are sales. Non-functional requirements for the project are related to the usability and effectiveness of the entire system. In this case it has been underlined the next non-functional requirements:

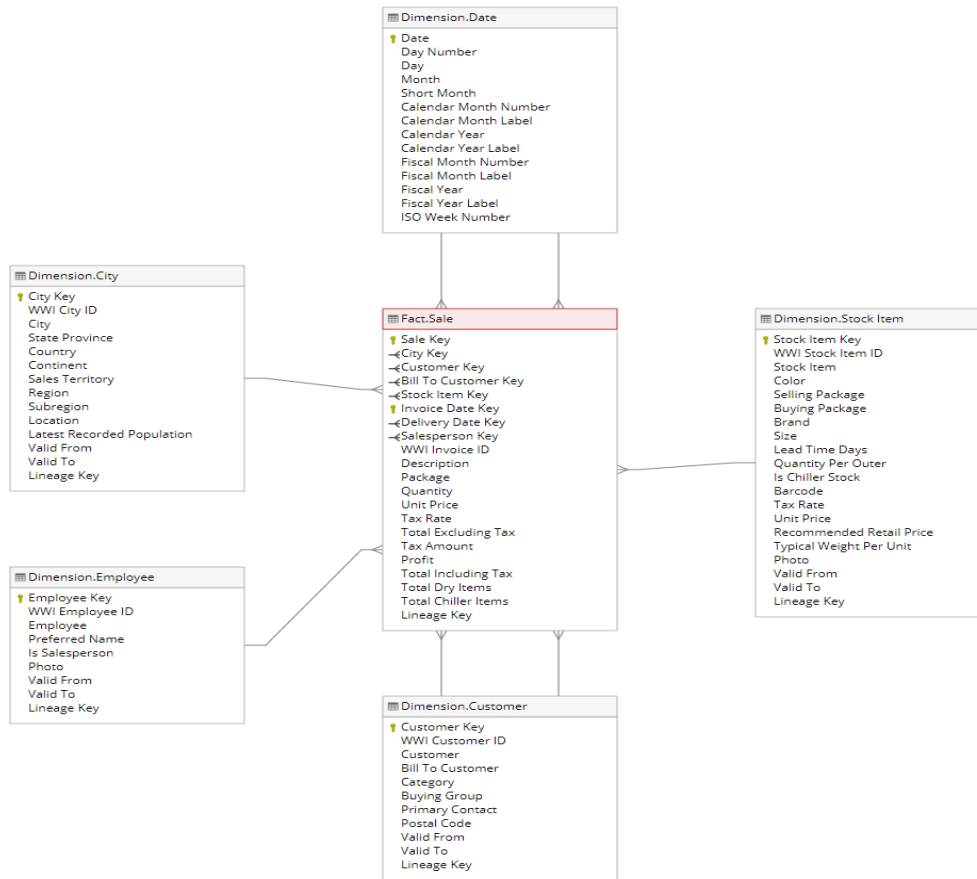
- ❖ *View, analyze and monitor the sales information with the capability of drill down. Optionally this could also be implemented for orders and/or transaction information.*
- ❖ *Perform multidimensional analysis on sales, order and/or transaction information.*
- ❖ *Allow users to quickly navigate around the data to discover new areas of concern.*

All in all the project has to fulfill all the features, functions, and tasks that are in demand. They give a clear set of parameters to work on and determine the various goals to be done for the customer.

Facts and Dimensions

Following the requirements that have to be fulfilled, it can be concluded that the core of the data mart in the project must be the Sale table. In these circumstances the Sales table is a fact table for this data mart. The fact table is the central table in the star schema and it stores the quantitative information for analysis. Fact table consists of two types of columns, foreign keys columns that are related to the dimensions and the measure columns. Companion tables for the fact table are dimensions that are the contextual descriptors of the business processes. Dimension tables are designed to group and filter the data from the fact table. In this case it will use the Employee, City, Stock, Customer, Date dimensions. All in all the data mart is created as a star schema that consists of all dimensions and the fact table that are needed to fulfill the requirements. This data mart is a subject-oriented database that is a partition segment from the entire enterprise database. It is used in order to perform sample analytics such as sales dates, delivery dates, profitability over time, profitability by sales person.

VIA - Software Technology Engineering - Course Assignment



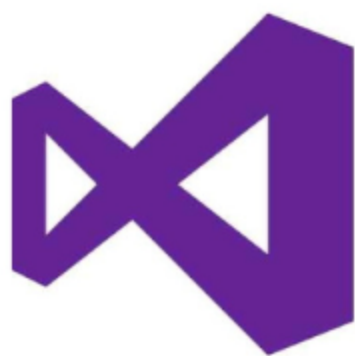
Source:

https://dataedo.com/samples/html/WideWorldImportersDW/doc/WideWorldImportersDW_6/modules/Sale_123/module.html

Business Process	Analytics	Fact	Dimensions
Sale	sales dates, delivery dates, profitability over time, profitability by sales person	Sale	Employee, City, Stock, Customer, Date

Cube

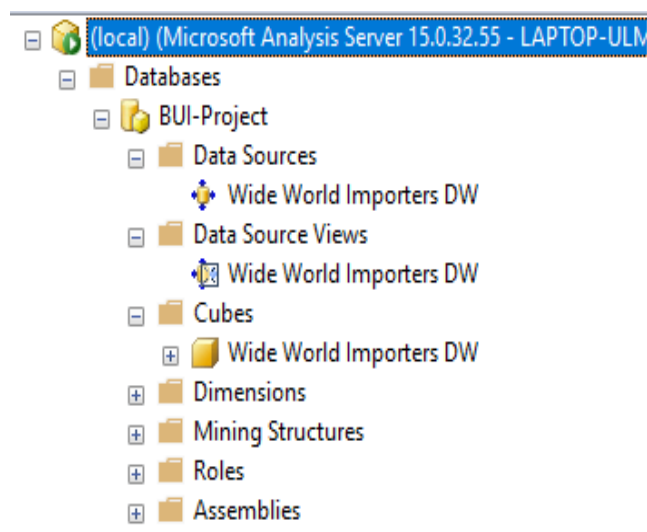
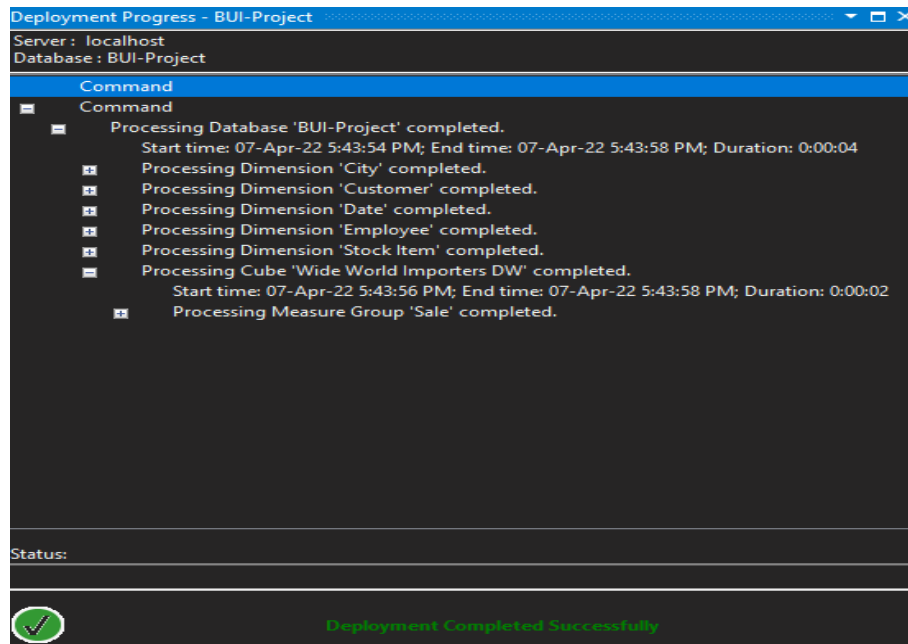
With the focus on developing a good product, it has been used different tools for specific purposes. It has been used Microsoft SQL Server (RDBMS) - relational database management systems that is used to store all the information in relational databases, as well as to manage, extract or work with such data without problems. SQL Server Management Studio (SSMS) - tool to access, develop and manage SQL Server databases as well as manage and operate Analysis services, Reporting services and Integration services. Microsoft Power BI - is a suite of business intelligence (BI), reporting, and data visualization products and services for individuals and teams. Visual Studio 2019 (SSDT) - modern development tool for building different services using the data from Microsoft SQL Server or other sources. In consideration of all the tools from above, it has been created, deployed and browsed the needed cube for the next analytics and developments.



Visual Studio

Create/Deploy/Browse a cube in Analysis Services (SSAS)

Using the tools listed in the previous section it has been created the cube needed. Creation of the cube starts with creating a new SQL Server Analysis Services project using VS, then creating all the needed elements such as Data Source, Data Source View and then the Cube. When the cube is done it has to be deployed and then it can be seen in the Analysis Services.



VIA - Software Technology Engineering - Course Assignment

As a result the cube is available to be used for other purposes or it can be browsed directly in the SSAS.

The image displays the SQL Server Data Tools (SSDT) interface for the Wide World Importers DW cube. The top section shows the cube design, and the bottom section shows the cube browser.

Measures:

- Wide World Importers DW
 - Sale
 - WWI Invoice ID
 - Quantity
 - Unit Price
 - Tax Rate
 - Total Excluding Tax
 - Tax Amount
 - Profit
 - Total Including Tax
 - Total Dry Items
 - Total Chiller Items
 - Lineage Key
 - Sale Count

Dimensions:

- Wide World Importers DW
 - Delivery Date
 - Stock Item
 - Invoice Date
 - Customer
 - City
 - Bill To Customer
 - Employee

Data Source View:

The Data Source View shows the following tables and their relationships:

- Employee**
 - Employee Key
 - WWI Employee ID
 - Employee
 - Preferred Name
 - Is Salesperson
 - Photo
 - Valid From
 - Valid To
 - Lineage Key
- Date**
 - Date
 - Day Number
 - Day
 - Month
 - Short Month
 - Calendar Month Number
 - Calendar Month Label
 - Calendar Year
 - Calendar Year Label
 - Fiscal Month Number
 - Fiscal Month Label
- Sale**
 - Sale Key
 - Invoice Date Key
 - City Key
 - Customer Key
 - Bill To Customer Key
 - Stock Item Key
 - Delivery Date Key
 - Salesperson Key
 - WWI Invoice ID
 - Description
 - Package
- City**
 - City Key
 - WWI City ID
 - City
 - State Province
 - Country
 - Continent
 - Sales Territory
 - Region
 - Subregion
 - Location
 - Latest Recorded Brand
- Customer**
 - Customer Key
 - WWI Customer ID
 - Customer
 - Bill To Customer
 - Category
 - Buying Group
 - Primary Contact
 - Postal Code
 - Valid From
 - Valid To
 - Lineage Key
- Stock Item**
 - Stock Item Key
 - WWI Stock Item ID
 - Stock Item
 - Color
 - Selling Package
 - Buying Package
 - Brand
 - Size
 - Lead Time Days
 - Quantity Per Outer
 - Chiller Stock

Wide World Importers DW.cube [Design] - X

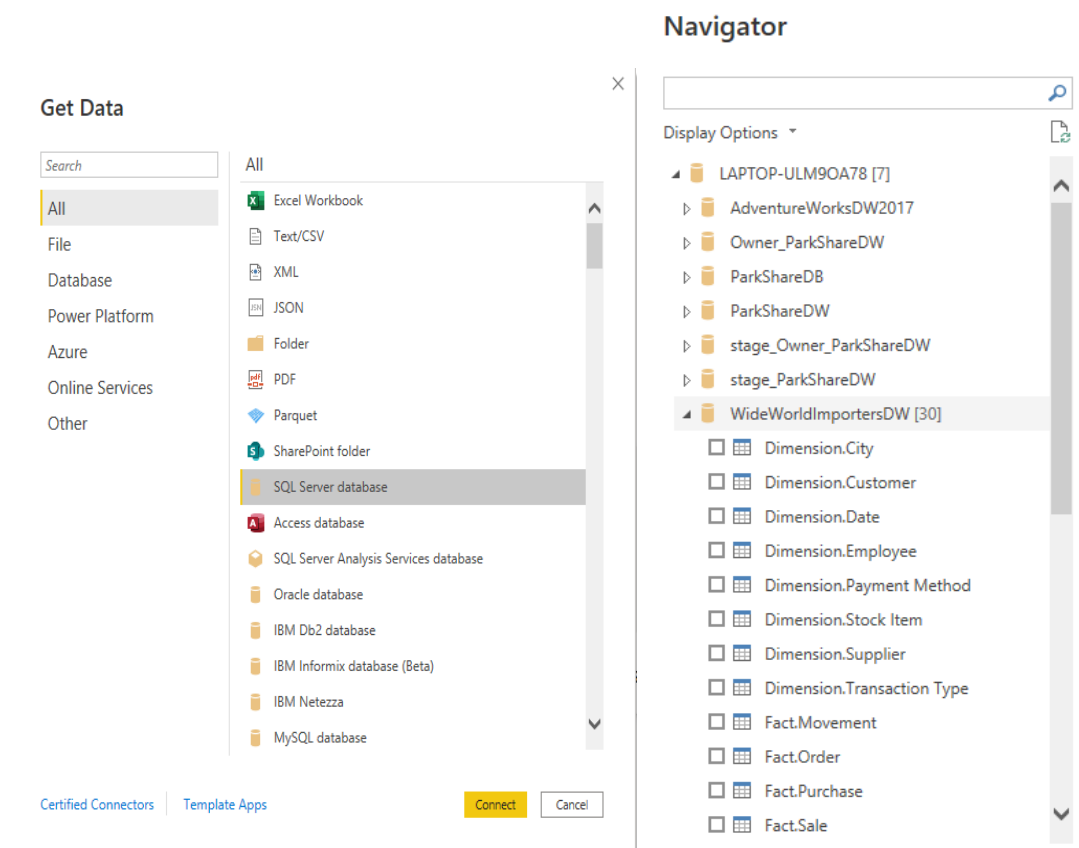
The bottom section shows the cube browser. The left pane displays the cube structure, and the right pane shows the data table.

City Key	Profit	Tax Amount	Tax Rate	Unit Price
37955	604...	17587.44	2505	4857.4
37965	506...	15360.98	1830	5400.98
37981	61069	18439.79	2265	10729.81
38005	104...	29887.59	2985	9145.14
38043	437...	13646.65	2025	4799.9
38135	507...	16497.86	2100	8904.86
38199	443...	13360.03	1980	3728.03
38209	645...	19524.7	2580	11294.78
38288	121...	37863.12	5035	16325.56
38318	158...	47994.86	5350	18312.52
38343	122...	37406.05	4770	14454.15
38468	130...	40477.62	5310	16570.37
38542	749...	22022.02	2835	7644.13
38583	125...	35529.41	4850	14817.86
38704	131...	38998.56	5410	13699.65
38737	643...	20304.89	2735	8121.02
38744	115...	36040.57	5190	13141.2
38845	443...	13877.79	2310	5255.62
38886	445...	13375.38	2175	4101.72
39003	477...	13464.59	2190	6364.65
39192	185...	5742.84	630	1686.66
39204	247...	7142.57	795	3798.79
39209	197...	6162.21	705	1906.68
39216	173...	5321.44	690	1922.29
39240	147...	4278.58	375	2646.44
39257	26374	7656.8	990	2818.38
39337	560...	16482.71	2100	7402.33
39346	733...	23811.01	2475	11091.43
39371	676...	16338.06	1035	6664.74

Chapter 2

Bringing in data from SQL Server tables

Business Intelligence is a set of processes that convert data into meaningful information that drives profitable business actions. Business Intelligence tools help to analyze the data as well as to create reports, dashboards, charts that provide customers with detailed information about the nature of the business. For the beginning it can be used the base Data Warehouse that is available in the database engine to get in touch with the available data.



At this step, the data available in the data warehouse can be used in 2 ways: Import or Direct Query. For this project the Direct Query is not the best solution because it does not support

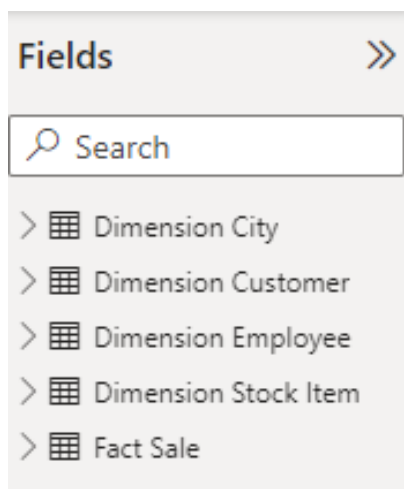
the DAX functions so it is going to be used the Import feature. From the WWI data warehouse it is going to be imported only the Sale Fact table and the dimensions that belong to it (*City, Customer, Date, Employee, Stock Item*).

Load

- Dimension City
116,295 rows from LAPTOP-ULM9OA78/WideWorldImportersDW.
- Dimension Customer
403 rows from LAPTOP-ULM9OA78/WideWorldImportersDW.
- Dimension Employee
213 rows from LAPTOP-ULM9OA78/WideWorldImportersDW.
- Dimension Stock Item
672 rows from LAPTOP-ULM9OA78/WideWorldImportersDW.
- Fact Sale
228,265 rows from LAPTOP-ULM9OA78/WideWorldImportersDW.

Cancel

At the end, in the PowerBI project are available to use all the imported fields.

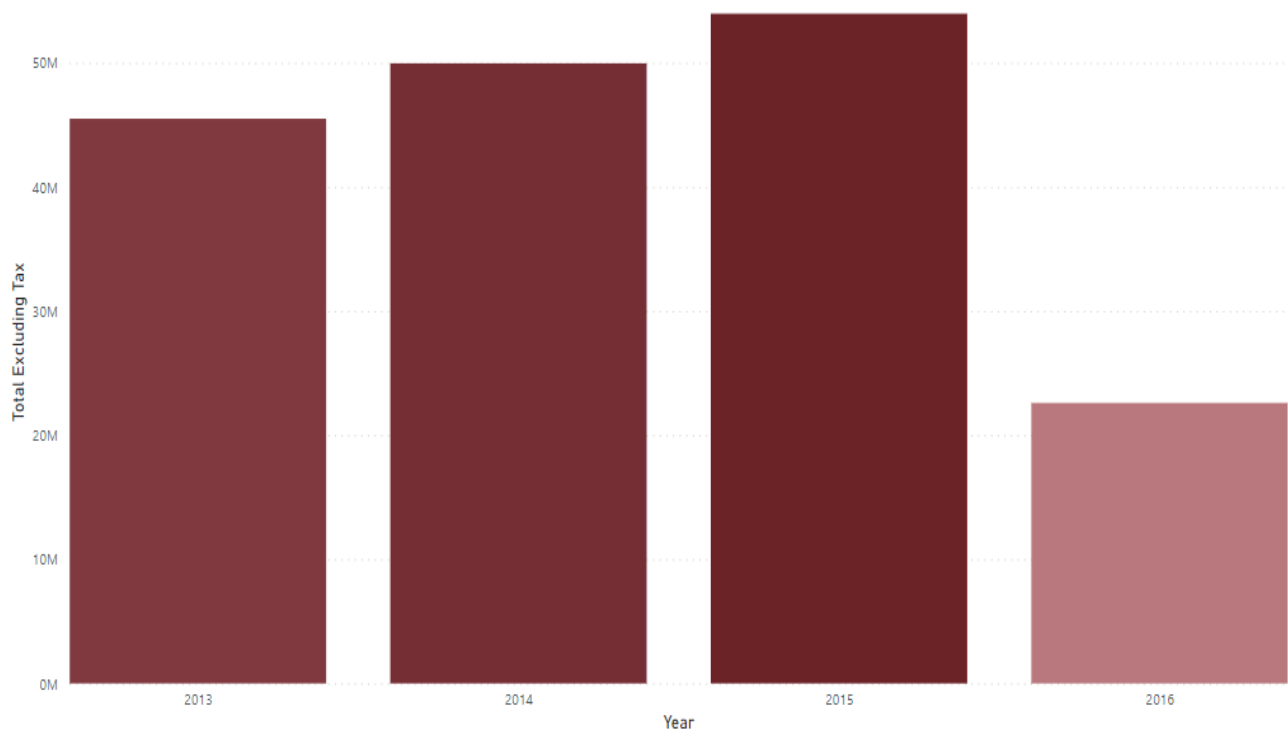


Building a basic column chart

A column chart is one of the most used charts in visualization of data because it is simple to create and also easy to understand. Usually column charts display the data which does not have a negative sign. Column chart divides one category of data in different clusters and makes a comparison between them. Column chart is going to be used to demonstrate the total sales divided by years with the possibility of drilling down on different time frames. For a more intuitive reading the chart a different colors were added and it shows with a more intensity the highest value of profit per year.

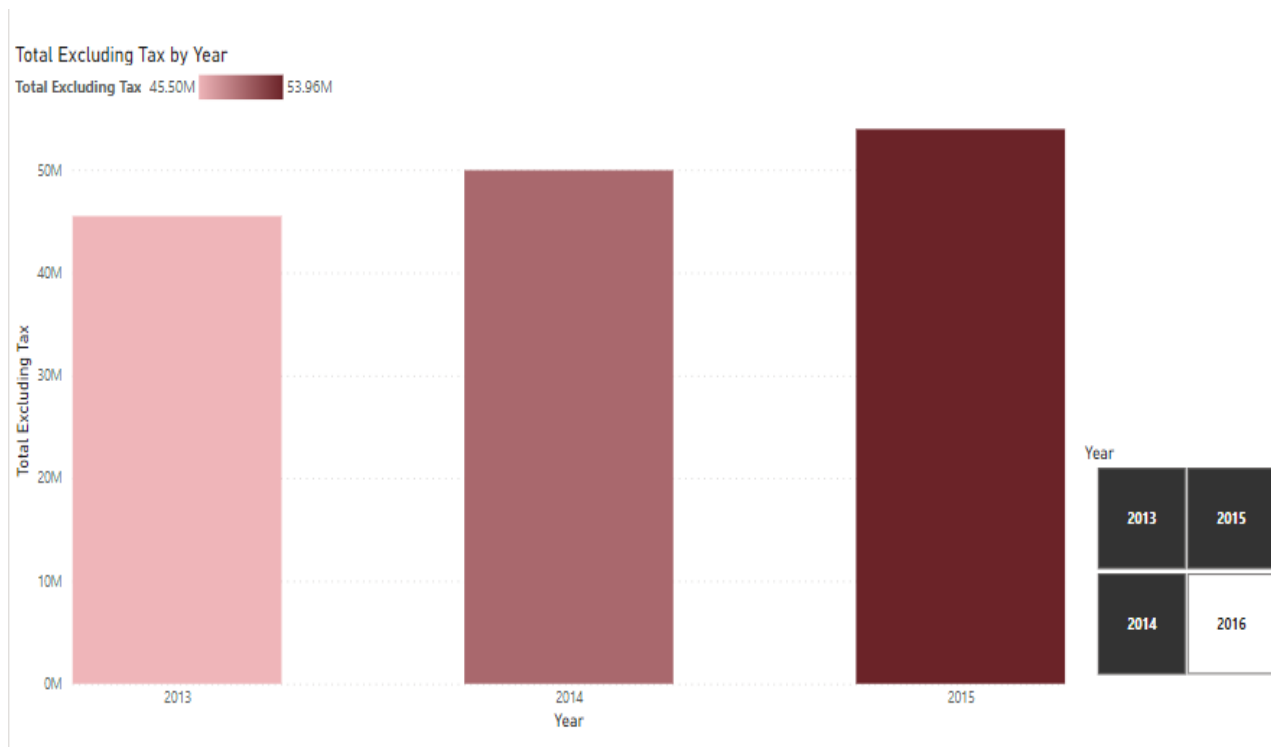
Total Excluding Tax by Year

Total Excluding Tax 0.23M 53.96M



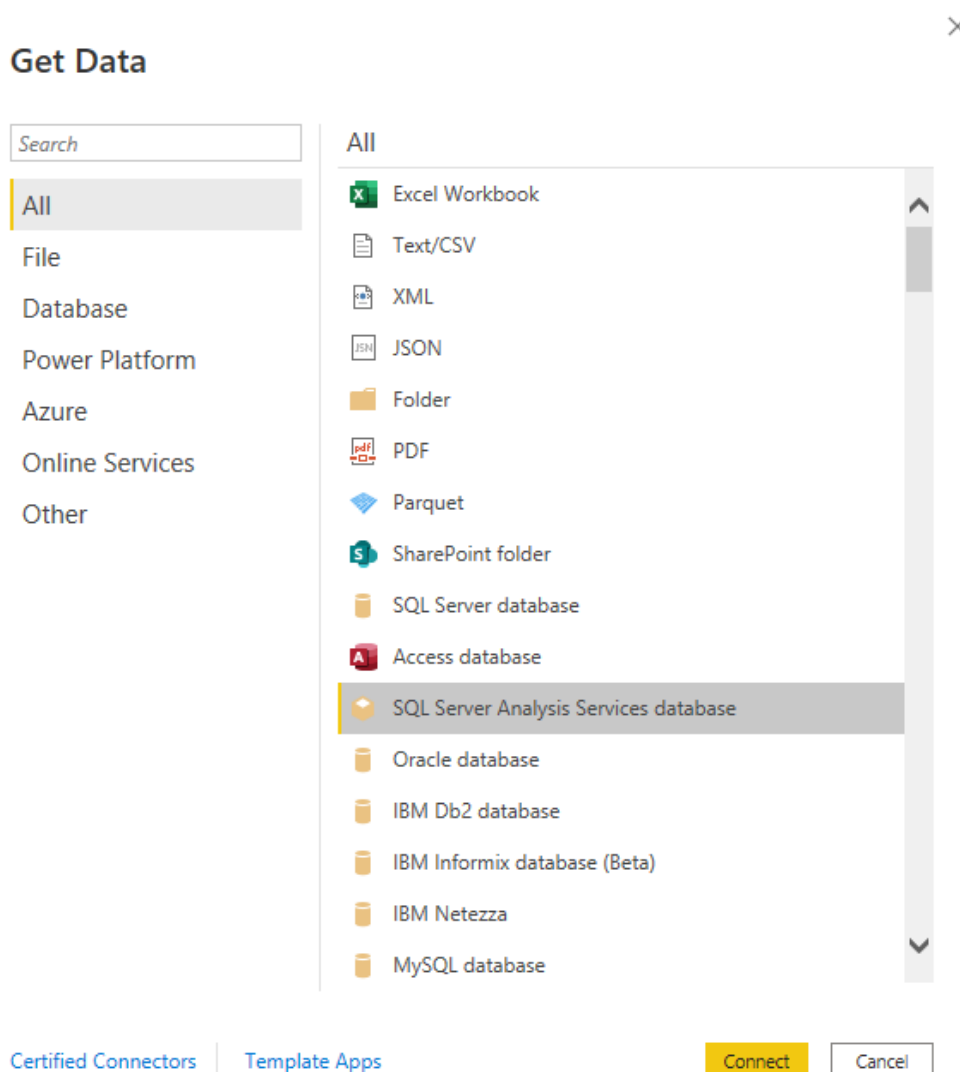
Creating a slicer

Slicer charts or grids are usually visual filters applied on another chart. With this tool it can be filtered or sorted the data from the base chart by selecting the needed fields or desired data. It is going to be used on the previous column chart in order to have the possibility to choose the years that can be displayed.



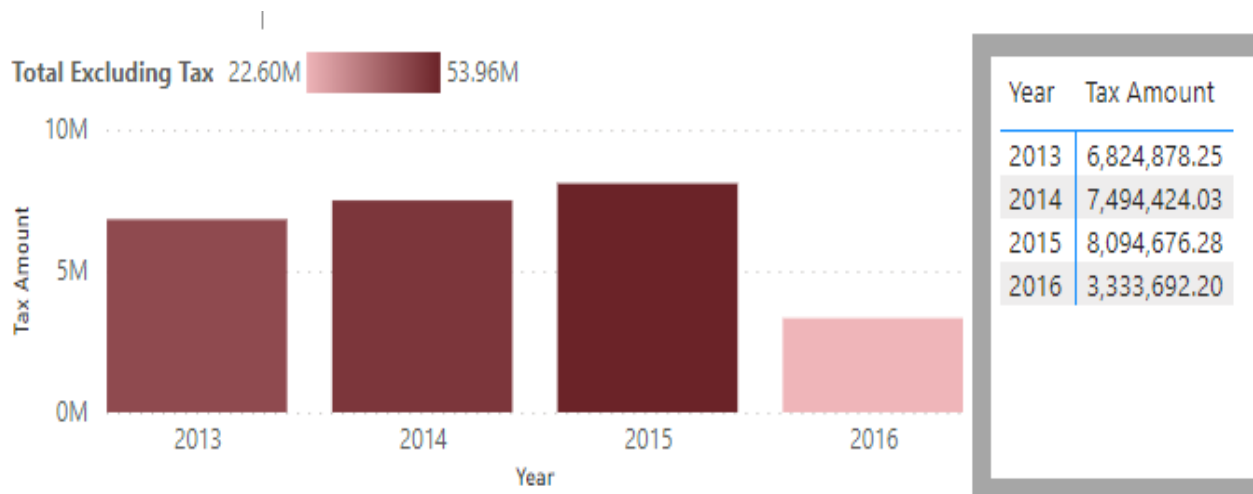
Connect to an existing SSAS-model with Power BI

In one of the previous sections - [Bringing in data from SQL Server tables](#) it has been shown how to import the data from an SQL server. In order to use the data from an SSAS project instead, at the level of choosing the data from what source had to be imported it is needed to select the *SQL Server Analysis Services database* and then, the other steps in order to get the data are the same.



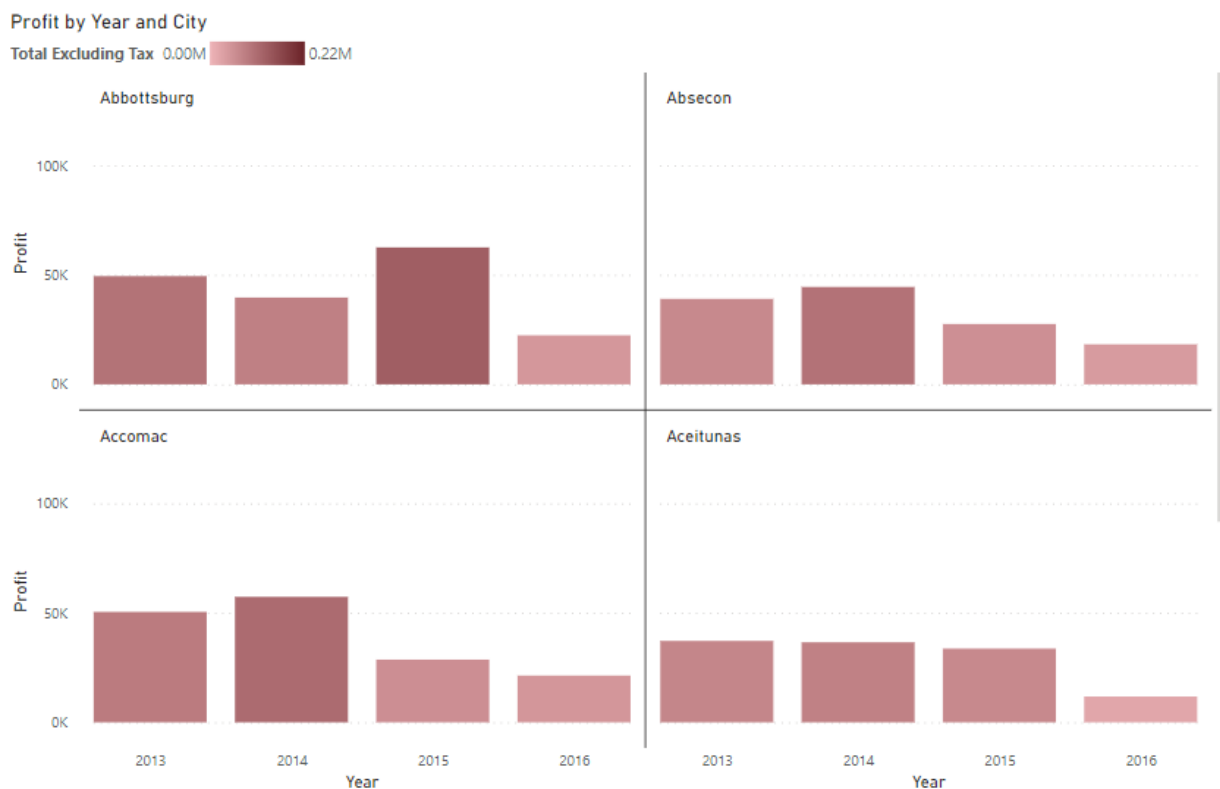
Viewing the data underlying a chart

In order to have an idea on what data is based on the chart made in Power BI for different business needs, it is needed to have the raw data nearby the main chart. It makes it easier to understand the quantity and the difference of the raw data. For this purpose a table with raw data for the column chart was added.



Add a second table from an existing data source

Improving the quality of data analytics can be made in different ways. Data analytics help a business to optimize its performance. Using descriptive analytics of data that describes what has happened over a given period of time (*are sales stronger this year than last?*). It can have a more precise overview of the business sales or other fields by adding different types of data to the analytics. In order to be more accurate, the City table has been added to the column chart, even so the data can be inspected by the year in each city. The technical step of adding a table into the Power BI can be seen [here](#).



Use a table visualization

A table is a grid that contains the related data in a series of rows and columns. Tables are useful if you are comparing the same category for many values. As an example in this project can be used the sales amount of each employee made every year. A table visualization is used to notice and contrast detailed data and precise values (*instead of visual representations*), to display data in a tabular format, to show numerical data by categories. To make it a lot easier to understand what is shown in the visualization or in a graph it is recommended to rename different tools that it is used. As an example it can be renamed the report tab used as also as the chart title.

Profit Overview			Employee	
Employee	Year	Profit		
Amy Trefl	2013	2,268,193.90	Alica Fatnowna	Jack Potter
Amy Trefl	2014	2,411,471.60		
Amy Trefl	2015	2,561,652.25	Amy Trefl	Jai Shand
Amy Trefl	2016	1,084,786.25		
Anthony Grosse	2013	2,236,904.60	Anthony Grosse	Katie Darwin
Anthony Grosse	2014	2,401,861.70		
Anthony Grosse	2015	2,584,552.50		
Anthony Grosse	2016	1,064,613.55		
Archer Lamble	2013	2,294,677.65	Archer Lamble	Kayla Woodcock
Archer Lamble	2014	2,676,553.25		
Archer Lamble	2015	2,762,899.00		
Archer Lamble	2016	1,208,287.55		
Hudson Hollinworth	2013	2,135,113.05	Ethan Onslow	Lily Code
Hudson Hollinworth	2014	2,549,861.35		
Hudson Hollinworth	2015	2,717,043.20	Eva Muirden	Piper Koch
Hudson Hollinworth	2016	1,139,418.10		
Hudson Onslow	2013	2,441,166.65	Henry Forlonge	Sophia Hinton
Hudson Onslow	2014	2,411,920.40		
Hudson Onslow	2015	2,724,025.25		
Hudson Onslow	2016	1,080,736.85		
Jack Potter	2013	2,294,500.95	Hudson Hollinworth	Stella Rosenhain
Jack Potter	2014	2,304,168.30		
Jack Potter	2015	2,797,565.45		
Jack Potter	2016	1,089,872.35	Hudson Onslow	Taj Shand
Kayla Woodcock	2013	2,312,402.65		
Kayla Woodcock	2014	2,614,734.95		
Kayla Woodcock	2015	2,696,247.25		
Kayla Woodcock	2016	1,095,964.25		
Lily Code	2013	2,172,384.00	Isabella Rupp	
Lily Code	2014	2,450,434.95		
Lily Code	2015	2,824,656.00		
Lily Code	2016	1,076,190.75		
Total		85,609,724.00		

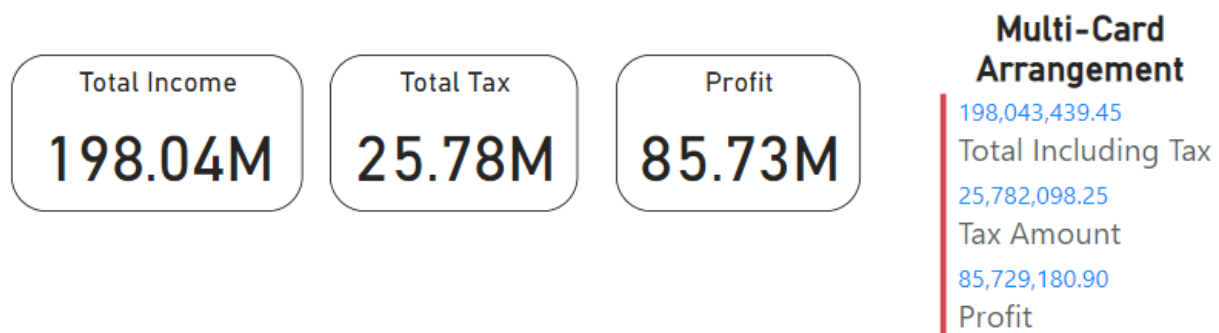
Work with a matrix visualization

At the first look the table view and the matrix one are looking the same. Both charts are displaying the data in the same format as a table and it is like the Excel tables. To understand the difference between those two visualizations is essential to display data in an efficient way. The basic difference is that table view shows the data in a two dimensional format. On top of that, matrix charts can have a specific option to specify the rows and columns as well as the layers of data to have the possibility to drill-up and down functionality. Matrix also provides the function to calculate the subtotals of the layers. All in all even if matrix's functionality is not used at its maximum, this is the better way to display the data.

Year	Amy Trefl	Anthony Grosse	Archer Lamble	Hudson Hollinworth	Hudson Onslow	Jack Potter	Kayla Woodcock	Lily Code	Sophia Hinton	Taj Shand	Total
2013	2,268,193.90	2,236,904.60	2,294,677.65	2,135,113.05	2,441,166.65	2,294,500.95	2,312,402.65	2,172,384.00	2,286,315.65	2,224,906.10	22,666,565.20
2014	2,411,471.60	2,401,861.70	2,676,553.25	2,549,861.35	2,411,920.40	2,304,168.30	2,614,734.95	2,450,434.95	2,508,122.20	2,515,992.70	24,845,121.40
2015	2,561,652.25	2,584,552.50	2,762,899.00	2,717,043.20	2,724,025.25	2,797,565.45	2,696,247.25	2,824,656.00	2,647,267.70	2,629,708.80	26,945,617.40
2016	1,084,786.25	1,064,613.55	1,208,287.55	1,139,418.10	1,080,736.85	1,089,872.35	1,095,964.25	1,076,190.75	1,088,072.35	1,224,478.00	11,152,420.00
Total	8,326,104.00	8,287,932.35	8,942,417.45	8,541,435.70	8,657,849.15	8,486,107.05	8,719,349.10	8,523,665.70	8,529,777.90	8,595,085.60	85,609,724.00

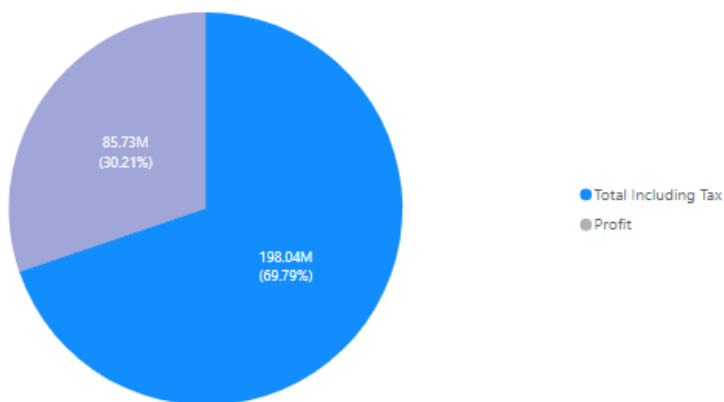
Use Card and Multi-row Card Visualizations

When it comes to understanding data to its highest level, there is no need to rely on powerful tools for illustration. Multi-Row cards or Single cards in Power BI are an excellent strategy to show the data in a simple way. Visual-oriented ways for easier understanding can be very easy and powerful compared to other charts, visualizations. Card visualizations are used to track and show data in an encapsulated form. With this type of displaying the information it can be displayed the data in a single form or in a group form.

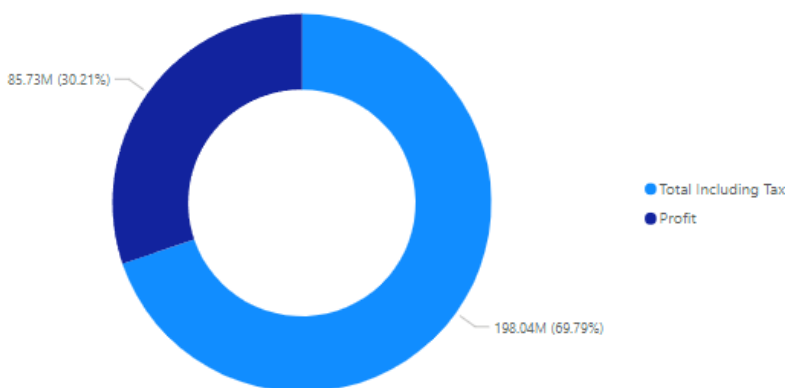


Work with pie charts and donut charts

A **pie chart** is a circular statistical visualization, and it displays the whole data divided in parts. Each slice of a pie chart represents the percentages, the sum of all slices should be equal to 100%. It helps users to understand the data quickly. They are widely used in education, the business world, and communication media. In the following example it can be seen that only 30% of the income is the profit.



Doughnuts are similar to pie charts. It can easily understand the data because doughnut charts show the whole data into the sections. It is the most useful chart when it needs to be displayed in various sections that make up the final value.

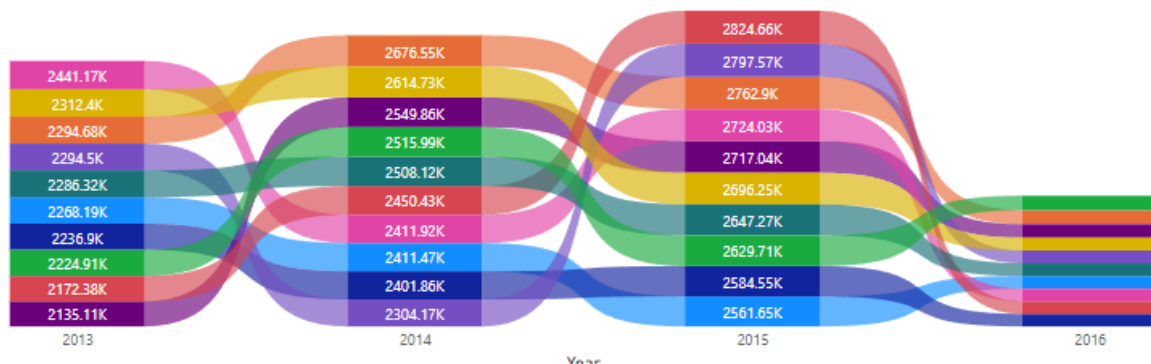


Begin a page and a visualization from the ribbon

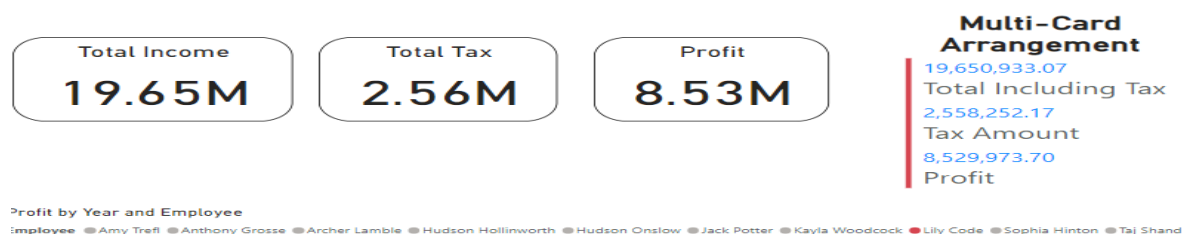
The Ribbon chart is a combination or union of different other charts, specifically a stacked chart or column chart added with a line chart. The tidy part is that the ribbon chart actually displays the changes in the rank of the ribbons, like in a line chart, and shows changes in rank and size. The core part for the chart is displaying the rank change over time with the difference of the rank value being on the top of the ribbon.

Profit by Year and Employee

Employee ● Amy Trefl ● Anthony Grosse ● Archer Lamble ● Hudson Hollinworth ● Hudson Onslow ● Jack Potter ● Kayla Woodcock ● Lily Code ● Sophia Hinton ● Taj Shand

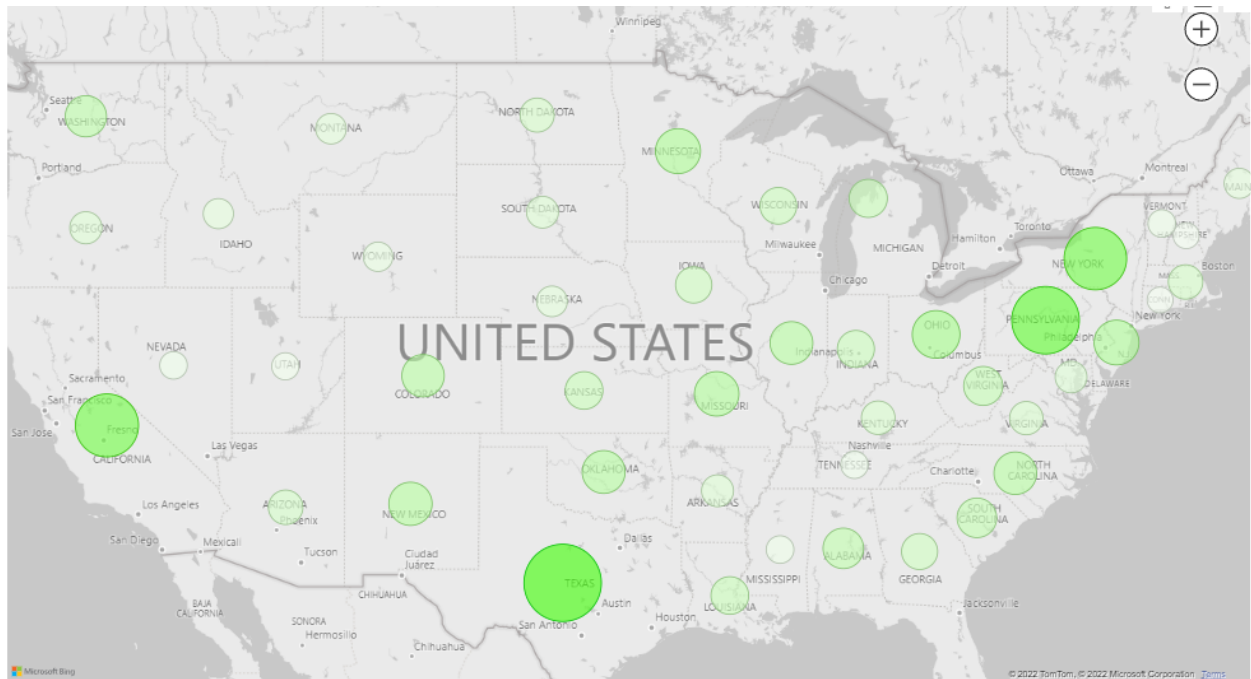


With the ribbon chart it is easy to show a different interaction between the charts as well as to display a specific data across the different charts which is very powerful and shows the direct interdependence. Below is an example with choosing only one employee to see the data.

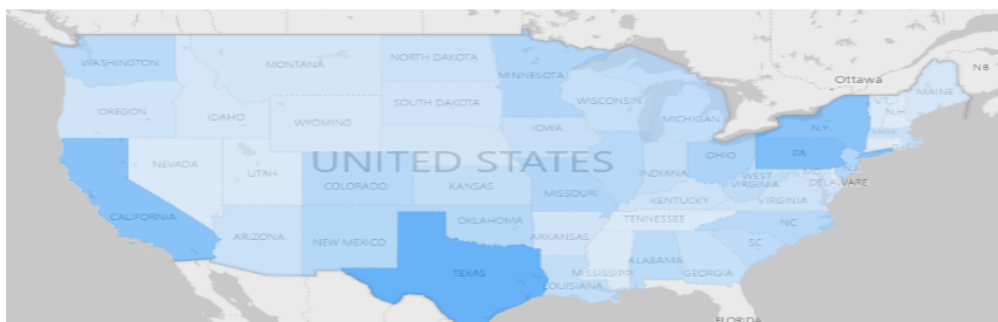


Work with a bubble map

A bubble map is used to represent the geographical distribution of data by plotting the same size points on the geographical background. It helps the user to grasp the overall distribution of the data, but it is a tough task if you want to observe specific data.



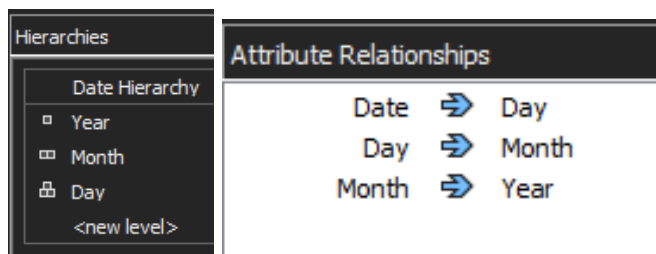
Here has been added a gradient color and different sizes of the bubbles to show the specific value for the profit in this region. Bigger and intensive colored bubble means that in this region profit is higher. The same effect can be displayed in a filled map.



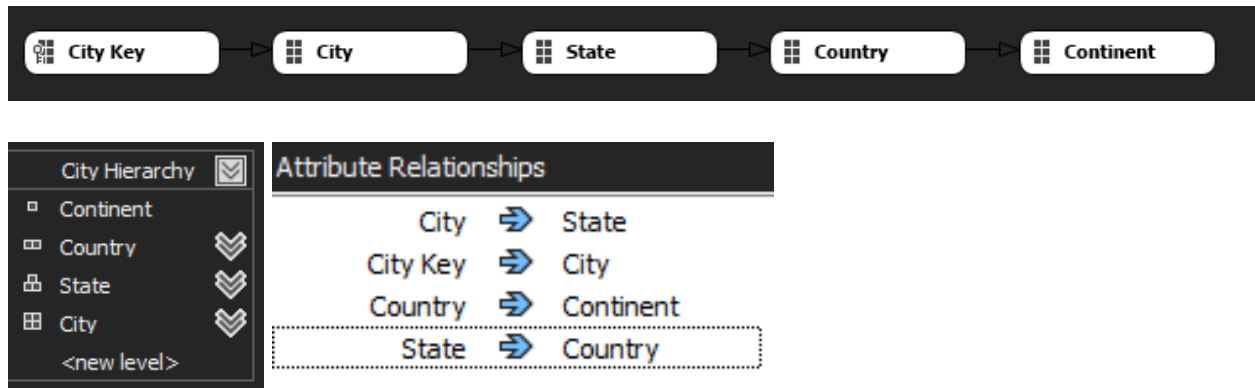
Chapter 3

Create a hierarchy in one or more of the dimensions

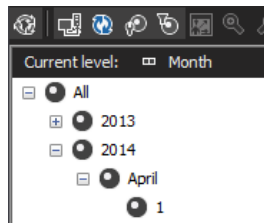
Hierarchies are a very useful tool in Analysis Service to reduce difficulty between different attributes and conduct users into distinct drill-down actions. Main purpose of a hierarchy is to group different attributes that belong to the same scope. A good example can be to group different attributes that belong to the date records such as: day, month, quarter, year or few attributes that belong to the city environment such as: state province, country, city. In order to create a hierarchy it is needed to define the attribute relationship. The attribute relationship should be rigid to not be changed over the time, also the `AttributeHierarchyVisible` property should be set to false in order to make visible only the hierarchy without other attributes, because it can make confusions for the one who is working with the cube.



Same things should be done to create the City Hierarchy:

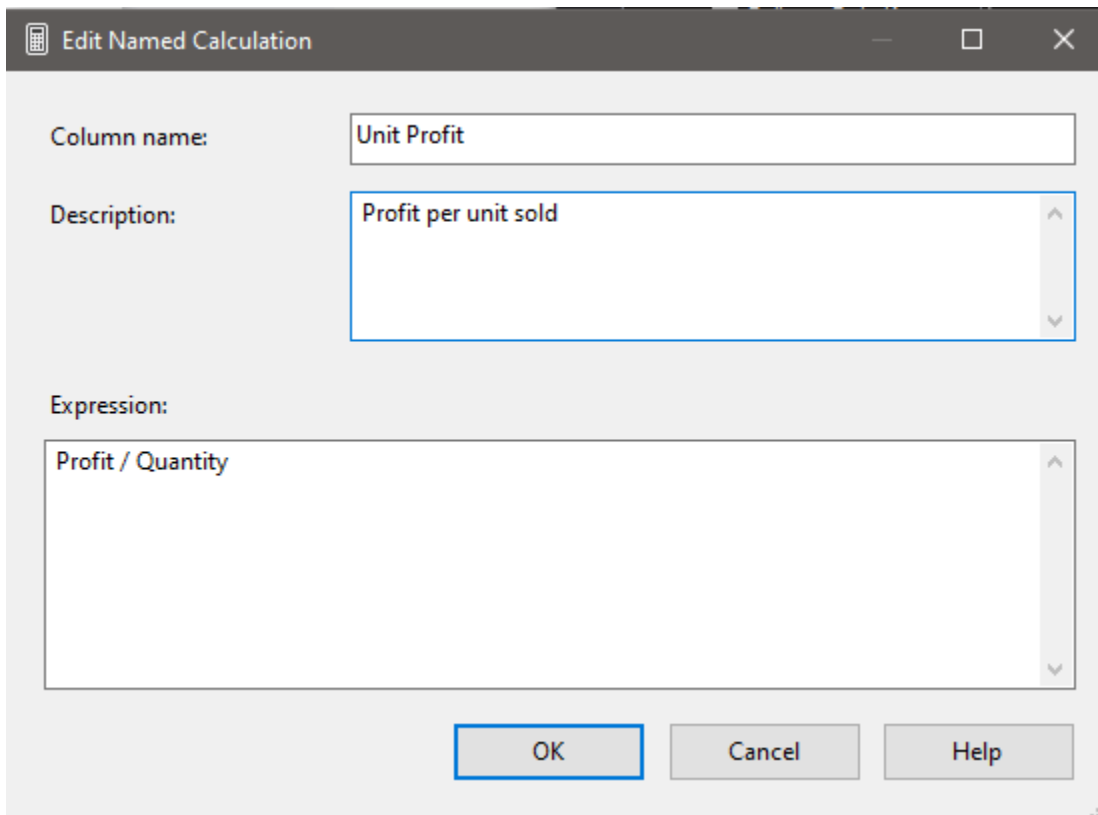


At the end, the changes can be seen in the Browsing mode where it can be expanded.

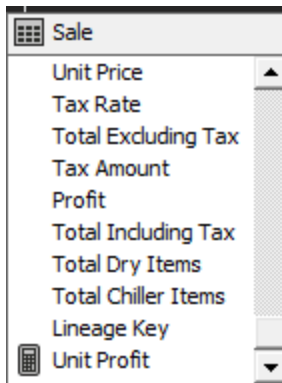


Create named calculations in the fact table and dimension

A calculated column allows adding different new data to the model. Instead of just importing or pasting new data, with DAX formula can be created the new column's values. For this purpose was created a Unit Profit column that holds the data where the profit on each item is divided by the quantity of this product and as result it is a profit per each unit sold.

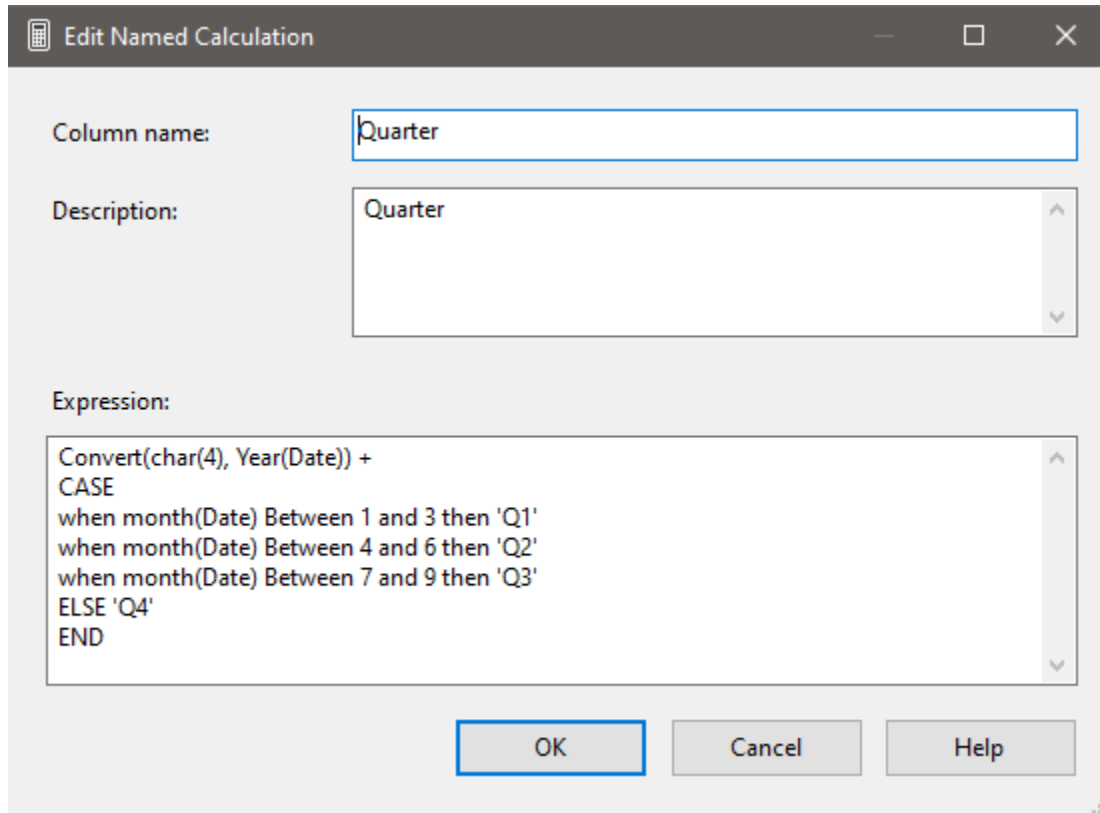


The screenshot shows the 'Edit Named Calculation' dialog box. It has three main sections: 'Column name', 'Description', and 'Expression'. The 'Column name' field is set to 'Unit Profit'. The 'Description' field is set to 'Profit per unit sold'. The 'Expression' field is set to 'Profit / Quantity'. At the bottom, there are three buttons: 'OK', 'Cancel', and 'Help'. The 'OK' button is highlighted with a blue border.



The screenshot shows a list of columns for a table named 'Sale'. The columns are: Unit Price, Tax Rate, Total Excluding Tax, Tax Amount, Profit, Total Including Tax, Total Dry Items, Total Chiller Items, Lineage Key, and Unit Profit. The 'Unit Profit' column is highlighted with a blue background.

On the dimension it has been added a column called Quarter that holds the data which divides the year in four periods.



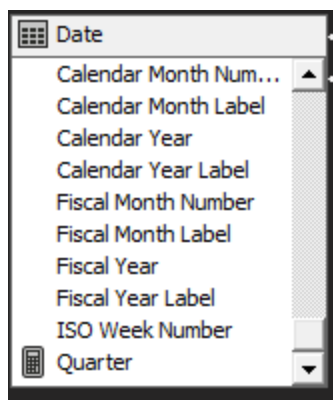
Edit Named Calculation

Column name:

Description:

Expression:

```
Convert(char(4), Year(Date)) +  
CASE  
when month(Date) Between 1 and 3 then 'Q1'  
when month(Date) Between 4 and 6 then 'Q2'  
when month(Date) Between 7 and 9 then 'Q3'  
ELSE 'Q4'  
END
```



Date

- Calendar Month Num...
- Calendar Month Label
- Calendar Year
- Calendar Year Label
- Fiscal Month Number
- Fiscal Month Label
- Fiscal Year
- Fiscal Year Label
- ISO Week Number
- Quarter**

The calculations are calculated during processing time. At this level it is just stored in the DSV, and also it does not affect the Data Sources.

Add Calculations to the Cube

A calculated member in the Cube is based on the combination of the cube's data. The data for the new columns are calculated at the query time. A new column has been created to display the values related to the total price for each item sold. It is calculated by multiplying the units sold with the unit price of the product.

SUM(YTD([Date].[Date Hierarchy].CURRENTMEMBER),[Measures].[Sale Count])

19

```
□ CALCULATE;  
□ CREATE MEMBER CURRENTCUBE.[Measures].TotalPrice  
  AS [Measures].[Quantity]*[Measures].[Unit Price],  
  FORMAT_STRING = "Currency",  
  NON_EMPTY_BEHAVIOR = { [Quantity], [Unit Price] },  
  VISIBLE = 1 , ASSOCIATED_MEASURE_GROUP = 'Sale' ;
```

VIA - Software Technology Engineering - Course Assignment

The screenshot shows the 'Script Organizer' on the left and the 'Properties' panel on the right. The 'Script Organizer' lists three items: 'Command', 'CALCULATE', and 'TotalPrice'. The 'TotalPrice' item is selected. The 'Properties' panel shows the following details for 'TotalPrice':

- Name: TotalPrice
- Parent Properties:
 - Parent hierarchy: Measures
 - Parent member: (empty)
- Expression:
 - [Measures].[Quantity]*[Measures].[Unit Price]
 - No issues found
- Additional Properties:
 - Format string: "Currency"
 - Visible: True
 - Non-empty behavior: Quantity, U...
 - Associated measure group: Sale
 - Display folder: (empty)
- Color Expressions: (checked)
- Font Expressions: (checked)

The screenshot shows a list of measures under the 'Sale' measure group. The measures are:

- Unit Price
- Tax Rate
- Total Excluding Tax
- Tax Amount
- Profit
- Total Including Tax
- Total Dry Items
- Total Chiller Items
- Lineage Key
- Unit Profit

Create a KPI (Key Performance Indicator)

A Key Performance Indicator has four important properties which are value, goal, status and trend. KPI shows how well the business is achieving its objectives that are imposed on the employees or other core goals. Overall KPIs display the estimative success at reaching the objectives. For this purpose a Sale KPI has been added to the cube in order to see the objectives status.

The screenshot displays the configuration interface for a KPI named 'SaleKPI'. The interface is organized into sections with expandable/collapsible headers. The 'Name' field is set to 'SaleKPI'. The 'Associated measure group' is set to 'Sale'. The 'Value Expression' is '[Measures].[Profit] / 100000'. The 'Goal Expression' is '5'. The 'Status' section has a 'Status indicator' set to 'Faces' and a 'Status expression' using MDX:

```
case
when kpiValue('SaleKPI') > kpiGoal('SaleKPI') then 1
when kpiValue('SaleKPI') < kpiGoal('SaleKPI') then -1
end
```

. The 'Trend' section has a 'Trend indicator' set to 'Standard arrow' and a 'Trend expression' using MDX:

```
when ([Date].[Year], [Measures].[Profit]) > ([Date].[Year].PREVMEMBER, [Measures].[Profit]) then 1
when ([Date].[Year], [Measures].[Profit]) < ([Date].[Year].PREVMEMBER, [Measures].[Profit]) then -1
else 0
end
```

. Each expression field has a 'No issues found' status and line/column indicators.

KPI

Name:

Associated measure group:

Value Expression

Ln: 1 Ch: 29 SPC CRLF

Goal Expression

Ln: 1 Ch: 2 SPC CRLF

Status

Status indicator:

Status expression:

Ln: 4 Ch: 4 SPC CRLF

Trend

Trend indicator:

Trend expression:

Ln: 5 Ch: 4 SPC CRLF

The value, goal and status for the SaleKPI has been set as also as using the MDX expression the Trend of the KPI.

Create Multiple Partitions from a Single Table and Add them to the Cube

Multidimensional projects usually use extremely large amounts of data. Partitioning the data can increase the manageability because old data can be removed in an easier way. When old data is cleaned from the source database, the matching partition can be deleted, so it is not necessary to reprocess the entire data from the cube. For this purpose has been created three different partitions that match three periods of time of the facts. Data that is before 2013, data that is between 2013 and 2015 and the data that is in 2015 and newer.

Item	Partition Name	Source	Estimated Rows	Storage Mode	Aggregation Design
1	Sale_2013	SELECT [Fact].[Sale].[Sale Key],[Fact].[Sale].[City Key],[Fact].[Sale].[Customer...	0	MOLAP	
2	Sale_2014	SELECT [Fact].[Sale].[Sale Key],[Fact].[Sale].[City Key],[Fact].[Sale].[Customer...	0	MOLAP	
3	Sale_2015	SELECT [Fact].[Sale].[Sale Key],[Fact].[Sale].[City Key],[Fact].[Sale].[Customer...	0	MOLAP	

Partition Source - Sale_2013

Binding type: Query Binding

Data source: Wide World Importers DW

```

SELECT [Fact].[Sale].[Sale Key],[Fact].[Sale].[City Key],[Fact].[Sale].[Customer Key],[Fact].[Sale].[Bill To
Customer Key],[Fact].[Sale].[Stock Item Key],[Fact].[Sale].[Invoice Date Key],[Fact].[Sale].[Delivery Date Key],
[Fact].[Sale].[Salesperson Key],[Fact].[Sale].[WWI Invoice ID],[Fact].[Sale].[Description],[Fact].[Sale].[Package],
[Fact].[Sale].[Quantity],[Fact].[Sale].[Unit Price],[Fact].[Sale].[Tax Rate],[Fact].[Sale].[Total Excluding Tax],[Fact].
[Sale].[Tax Amount],[Fact].[Sale].[Profit],[Fact].[Sale].[Total Including Tax],[Fact].[Sale].[Total Dry Items],[Fact].
[Sale].[Total Chiller Items],[Fact].[Sale].[Lineage Key],Profit / Quantity AS [Unit Profit]
FROM [Fact].[Sale]
WHERE YEAR([Fact].[Sale].[Delivery Date Key]) < 2014
      
```

Check

This query must exclude any rows from this table that are already included in other partitions. If these rows are not excluded, duplicate aggregation will occur in measure groups with more than one partition.

OK Cancel Help

Partition Wizard

Restrict Rows
Choose whether to use a query to restrict the rows that are included in the partition.

☒ Specify a query to restrict rows

Query:

```
To Customer Key],[Fact].[Sale].[Stock Item Key],[Fact].[Sale].[Invoice Date Key],[Fact].[Sale].[Delivery Date Key],[Fact].[Sale].[Salesperson Key],[Fact].[Sale].[WWI Invoice ID],[Fact].[Sale].[Description],[Fact].[Sale].[Package],[Fact].[Sale].[Quantity],[Fact].[Sale].[Unit Price],[Fact].[Sale].[Tax Rate],[Fact].[Sale].[Total Excluding Tax],[Fact].[Sale].[Tax Amount],[Fact].[Sale].[Profit],[Fact].[Sale].[Total Including Tax],[Fact].[Sale].[Total Dry Items],[Fact].[Sale].[Total Chiller Items],[Fact].[Sale].[Lineage Key],Profit / Quantity AS [Unit Profit]
FROM [Fact].[Sale]
WHERE YEAR([Fact].[Sale].[Delivery Date Key]) > 2013
AND YEAR([Fact].[Sale].[Delivery Date Key]) < 2015
```

This query must exclude any rows from this table that are already included in other partitions. If these rows are not excluded, duplicate aggregation will occur in measure groups with more than one partition.

Partition Wizard

Restrict Rows
Choose whether to use a query to restrict the rows that are included in the partition.

☒ Specify a query to restrict rows

Query:

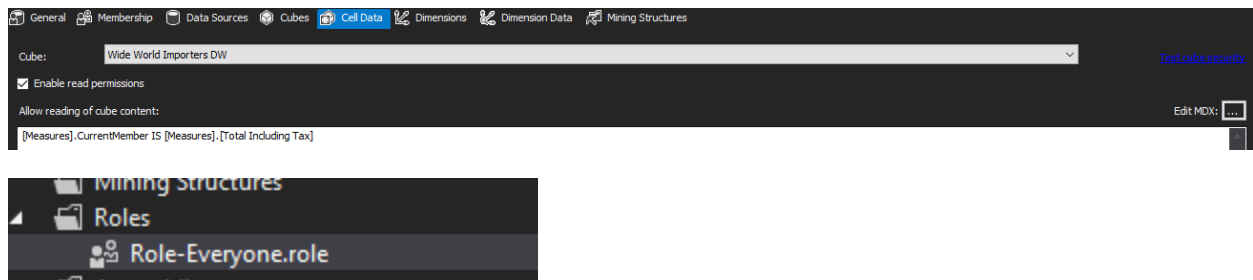
```
SELECT [Fact].[Sale].[Sale Key],[Fact].[Sale].[City Key],[Fact].[Sale].[Customer Key],[Fact].[Sale].[Bill To Customer Key],[Fact].[Sale].[Stock Item Key],[Fact].[Sale].[Invoice Date Key],[Fact].[Sale].[Delivery Date Key],[Fact].[Sale].[Salesperson Key],[Fact].[Sale].[WWI Invoice ID],[Fact].[Sale].[Description],[Fact].[Sale].[Package],[Fact].[Sale].[Quantity],[Fact].[Sale].[Unit Price],[Fact].[Sale].[Tax Rate],[Fact].[Sale].[Total Excluding Tax],[Fact].[Sale].[Tax Amount],[Fact].[Sale].[Profit],[Fact].[Sale].[Total Including Tax],[Fact].[Sale].[Total Dry Items],[Fact].[Sale].[Total Chiller Items],[Fact].[Sale].[Lineage Key],Profit / Quantity AS [Unit Profit]
FROM [Fact].[Sale]
WHERE YEAR([Fact].[Sale].[Delivery Date Key]) >= 2015
```

This query must exclude any rows from this table that are already included in other partitions. If these rows are not excluded, duplicate aggregation will occur in measure groups with more than one partition.

In order to be able to query the partitions it has been changed the binding type from table binding to the query binding.

Limit access to cube members

A cube project is the primary query object in an Analysis Services data model. When connecting to multidimensional data from Excel or SSAS for ad hoc data exploration, users typically begin by selecting, querying, transforming a specific cube model. In order to restrict the access to the data that can be accessed, it can be created different roles of accessing the data. It can be as an Administrator, or processing the data or only to read it. Also it can give the access to a specific data as an example, below is a role for everyone to only read the data from the fact table and only the Total Including Tax column. Using MDX expression it can be set a different environment for accessing the data.

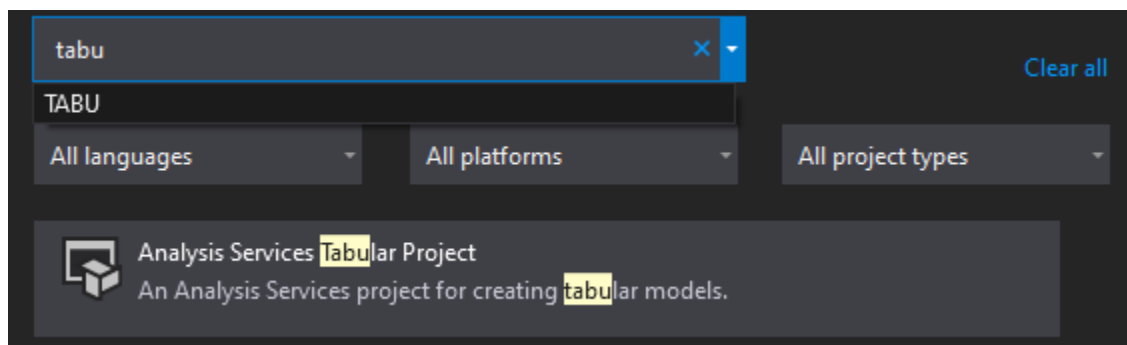


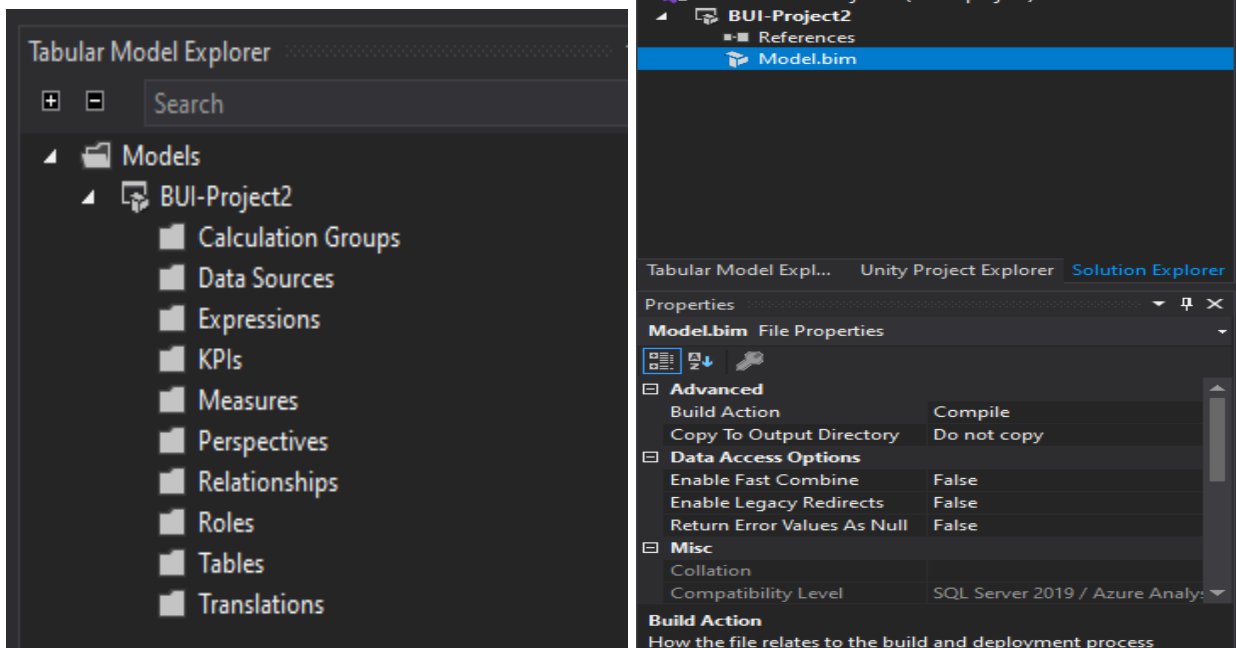
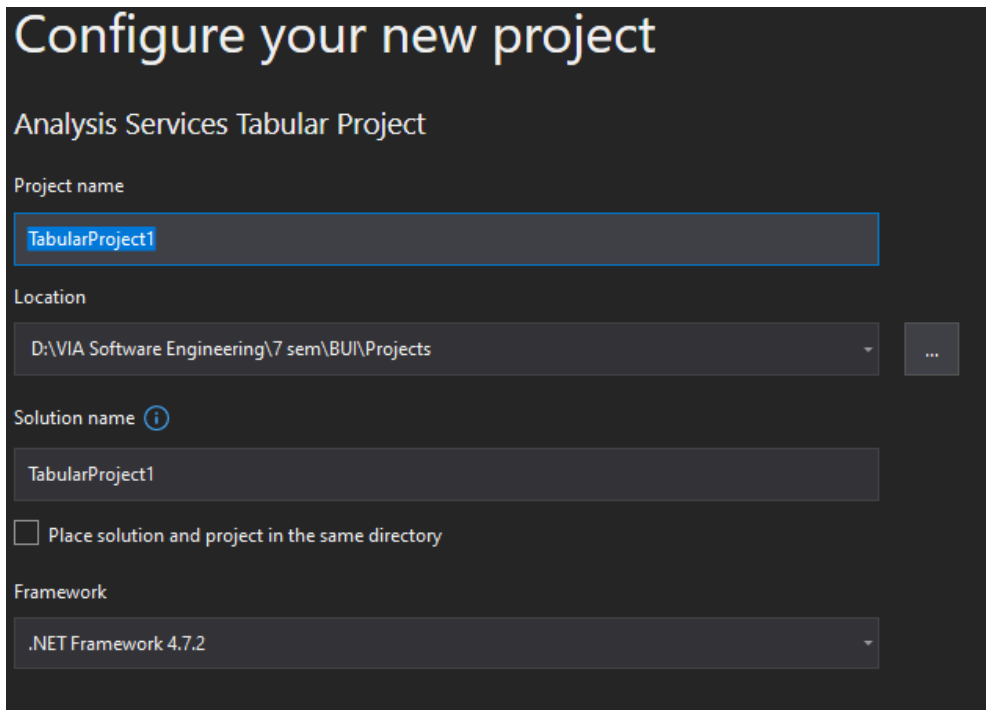
MDX - **Multidimensional Expressions** - can be used in many ways in the SSAS. It is the core query language for providing different environments, calculations and other nice features for the project.

Chapter 4

Create Tabular BI Semantic Model

Tabular models are Analysis Services projects that are developed locally (*in-memory*) or in Direct Query mode, accessing data directly from relational databases. Tabular Model is a collection of tables with high levels of performance, supporting two modes of interaction with the data: in-memory and DirectQuery. Tabular Model has the function to combine the MOLAP features like KPI, Measures and so on, with the Relational Database tables and relationships, in a simple way. Tabular model compared to other tools of BI, is providing better performance, low maintenance, ownership cost, simplicity and flexibility in data modeling. On this purpose a Tabular BISM has been developed starting by creating a Tabular BISM Project using the same tools as for the Multidimensional Model project.





At the end it has been created a new empty Tabular Model.

Import SQL Server tables into the model

In order to populate the Tabular Model with data, it has been connected to the SQL server from where it has been retrieved the needed data for the Tabular Model project such as Customer, City, Employee, Date, Stock Item, Sale tables.

The screenshot shows the 'Get Data' window on the left and the 'Navigator' window on the right. In the 'Get Data' window, the 'Database' category is selected, and 'SQL Server database' is chosen. In the 'Navigator' window, the 'Fact.Sale' table is selected in the 'Display Options' pane. The preview of the 'Fact.Sale' table shows the following data:

Sale Key	City Key	Customer Key	Bill To Customer Key	Stock Item Key
1	68644	0	0	0
2	47692	0	0	0
3	47692	0	0	0
4	68504	105	1	1
5	70644	57	1	1
6	70644	57	1	1
7	70644	57	1	1
8	56153	0	0	0
9	56153	0	0	0
10	56153	0	0	0
11	50969	0	0	0
12	50969	0	0	0
13	50969	0	0	0
14	58731	376	202	202
15	58731	376	202	202
16	58731	376	202	202
17	58731	376	202	202

Model.bim - City Key - 98596

City Key	WWI City ID	City	State Province	Country	Continent	Sales Territory	Region	Subregion	Location	Latest Recorded Population	Valid From	Valid To	Lineage Key	Add Column
1	98656	11945	Fort...	Arkansas	United St...	North America	Southeast	Americas	Northern A...	POINT (9...	0	01-30-14 4...	1	
2	98662	12112	Fox	Arkansas	United St...	North America	Southeast	Americas	Northern A...	POINT (9...	0	01-30-14 4...	1	
3	98667	12518	Gaine...	Arkansas	United St...	North America	Southeast	Americas	Northern A...	POINT (9...	0	01-30-14 4...	1	
4	98668	12581	Gamaled	Arkansas	United St...	North America	Southeast	Americas	Northern A...	POINT (9...	0	01-30-14 4...	1	
5	98674	12834	Genoa	Arkansas	United St...	North America	Southeast	Americas	Northern A...	POINT (9...	0	01-30-14 4...	1	
6	98678	12957	Gifford	Arkansas	United St...	North America	Southeast	Americas	Northern A...	POINT (9...	0	01-30-14 4...	1	
7	98683	13162	Glencoe	Arkansas	United St...	North America	Southeast	Americas	Northern A...	POINT (9...	0	01-30-14 4...	1	
8	98684	13179	Glendale	Arkansas	United St...	North America	Southeast	Americas	Northern A...	POINT (9...	0	01-30-14 4...	1	
9	98686	13419	Goodwin	Arkansas	United St...	North America	Southeast	Americas	Northern A...	POINT (9...	0	01-30-14 4...	1	
10	98691	13577	Grand...	Arkansas	United St...	North America	Southeast	Americas	Northern A...	POINT (9...	0	01-30-14 4...	1	
11	98693	13716	Grape...	Arkansas	United St...	North America	Southeast	Americas	Northern A...	POINT (9...	0	01-30-14 4...	1	
12	98694	13738	Grave...	Arkansas	United St...	North America	Southeast	Americas	Northern A...	POINT (9...	0	01-30-14 4...	1	
13	98695	13739	Gravelly	Arkansas	United St...	North America	Southeast	Americas	Northern A...	POINT (9...	0	01-30-14 4...	1	
14	98697	13767	Grays	Arkansas	United St...	North America	Southeast	Americas	Northern A...	POINT (9...	0	01-30-14 4...	1	
15	98700	13896	Green...	Arkansas	United St...	North America	Southeast	Americas	Northern A...	POINT (9...	0	01-30-14 4...	1	
16	98705	14016	Gregory	Arkansas	United St...	North America	Southeast	Americas	Northern A...	POINT (9...	0	01-30-14 4...	1	
17	98706	14043	Grider	Arkansas	United St...	North America	Southeast	Americas	Northern A...	POINT (9...	0	01-30-14 4...	1	
18	98715	14353	Halley	Arkansas	United St...	North America	Southeast	Americas	Northern A...	POINT (9...	0	01-30-14 4...	1	
19	98716	14354	Halliday	Arkansas	United St...	North America	Southeast	Americas	Northern A...	POINT (9...	0	01-30-14 4...	1	
20	98719	14545	Hanover	Arkansas	United St...	North America	Southeast	Americas	Northern A...	POINT (9...	0	01-30-14 4...	1	
21	98721	14663	Harm...	Arkansas	United St...	North America	Southeast	Americas	Northern A...	POINT (9...	0	01-30-14 4...	1	
22	98723	14690	Harriet	Arkansas	United St...	North America	Southeast	Americas	Northern A...	POINT (9...	0	01-30-14 4...	1	
23	98728	14829	Harvey	Arkansas	United St...	North America	Southeast	Americas	Northern A...	POINT (9...	0	01-30-14 4...	1	
24	98731	14892	Hattie...	Arkansas	United St...	North America	Southeast	Americas	Northern A...	POINT (9...	0	01-30-14 4...	1	

Tabular Model Explorer

- Models
 - BUI-Project2
 - Calculation Groups
 - Data Sources
 - SQL/LAPTOP-ULM90A78;WideWorldImporters
 - Expressions
 - KPIs
 - Measures
 - Perspectives
 - Relationships
 - Fact Sale [Bill To Customer Key] - Dimension C
 - Fact Sale [City Key] - Dimension City [City Key]
 - Fact Sale [Customer Key] - Dimension Custom
 - Fact Sale [Delivery Date Key] - Dimension Date
 - Fact Sale [Invoice Date Key] - Dimension Date
 - Fact Sale [Stock Item Key] - Dimension Stock I
 - Roles
 - Tables
 - Dimension City
 - Dimension Customer
 - Dimension Date
 - Dimension Stock Item
 - Fact Sale
 - Translations

VIA - Software Technology Engineering - Course Assignment

As an alternative for importing data it can be transformed at the moment of the loading, so it can be changed in a way that it can be deleted columns that aren't needed in the model or other changes.

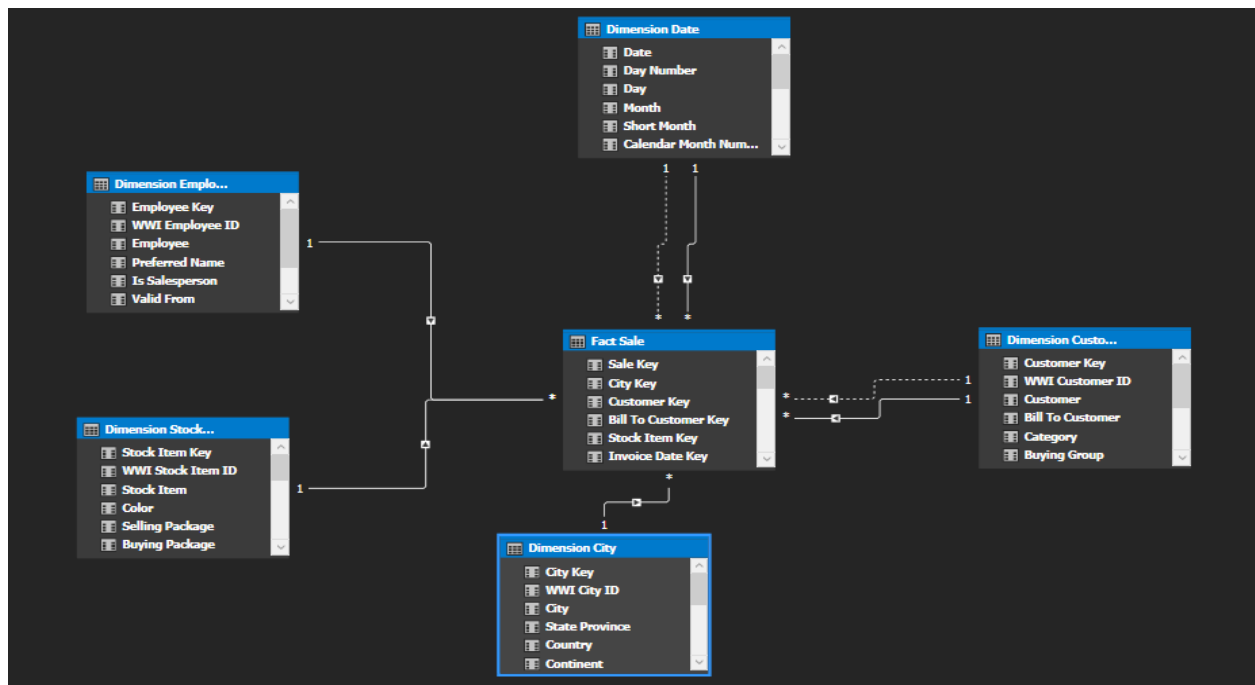
Dimension City - Power Query Editor

Source: Source[[Schema="Dimension",Item="City"]][Data]

City Key	WWI City ID	City	State Province	Country	Continent	Sales Territory	Region
0	Unknown	N/A	N/A	N/A	N/A	N/A	N/A
1	5450	Carrollton	New York	United States	North America	Midwest	Americas
2	5451	Carrollton	Virginia	United States	North America	Southeast	Americas
3	5452	Carrollton	Illinois	United States	North America	Great Lakes	Americas
4	5453	Carrollton	Missouri	United States	North America	Plains	Americas
5	5454	Carrollton	Ohio	United States	North America	Great Lakes	Americas
6	5455	Carrollton	Kentucky	United States	North America	Southeast	Americas
7	5456	Carrollton	Georgia	United States	North America	Southeast	Americas
8	5457	Carrollton	Alabama	United States	North America	Southeast	Americas
9	5458	Carrollton	Mississippi	United States	North America	Southeast	Americas
10	5459	Carrollton	Texas	United States	North America	Southwest	Americas
11	5460	Carrollton Manor	Maryland	United States	North America	Midwest	Americas
12	5461	Carrolltown	Pennsylvania	United States	North America	Midwest	Americas
13	5462	Carrothers	Ohio	United States	North America	Great Lakes	Americas
14	5463	Carrsville	Virginia	United States	North America	Southeast	Americas
15	5464	Carrsville	Kentucky	United States	North America	Southeast	Americas
16	5465	Carsons	Maryland	United States	North America	Midwest	Americas
17	5466	Carson	New Mexico	United States	North America	Southwest	Americas
18	5467	Carson	California	United States	North America	Far West	Americas
19	5468	Carson	Washington	United States	North America	Far West	Americas
20	5469	Carson	North Dakota	United States	North America	Plains	Americas
21	5470	Carson	Iowa	United States	North America	Plains	Americas
22	5471	Carson City	Nevada	United States	North America	Far West	Americas
23	5472	Carson City	Michigan	United States	North America	Great Lakes	Americas
24	5473	Carsonville	Michigan	United States	North America	Great Lakes	Americas
25	5474	Carsonville	Georgia	United States	North America	Southeast	Americas

16 COLUMNS, 999+ ROWS

PREVIEW DOWNLOADED AT 6:51 PM



TotalQuantity :=SUM('Fact Sale'[Quantity])

$$f_x = \text{'Fact Sale' [Quantity]} * \text{'Fact Sale' [Unit Price]}$$
[illegible]

In order to keep track of the information, data needed in the first look, it has been performed different DAX functions that bring the data from other columns or tables in the table that it is working on. *DAX - LOOKUPVALUE* - Returns the value for the row that meets all criteria specified by one or more search conditions. *DAX - RELATED* - The column that contains the values you want to retrieve. <https://docs.microsoft.com/en-us/dax>

Sale Country	
United States	
United States	
United States	
United States	
United States	
United States	
United States	
United States	
United States	

$$f_x = \text{LOOKUPVALUE}(\text{'Dimension City'[Country]}, [\text{City Key}], [\text{City Key}])$$

Sale Country	SaleCity
United States	Eagle Valley
United States	Lunds
United States	Kopperl
United States	White Horse Beach
United States	Chipita Park
United States	Mashulaville
United States	Beaver Bay
United States	Shawboro

$$f_x = \text{RELATED}(\text{'Dimension City'[City]})$$

To be in touch with the column and measure meanings it can be added a description, which has been performed for custom measures and columns.

Total Profit: 85729180.9
Average Profit: 375.568663176571
PROFIT YDT: 11174765.55

Display Folder	
Basic	
Description	Total profit made.
Format	General
Formula	SUM('Fact Sale'[Profit])
Measure Name	Total Profit

Column Description

Enter description for column 'SaleCity':

The city where the sale had been made.

Create a KPI

[For the same purpose for the project](#), it can be added a KPI to keep track of the business requirements.

Key Performance Indicator (KPI) ? X

KPI base measure (value): Average Profit v

KPI Status

Target

☐ Measure: v

☒ Absolute value: 500

200 400 Target

Select icon style:

Descriptions

OK Cancel

Average Profit:
375.568663176571

Deploy Tabular Model Project

Almost the same way as [here](#).

Deploy

Deploying

The deployment operation may take several minutes to complete.

✓

Success

7 Total0 Cancelled

7 Success0 Error

Details:

	Work Item	Status	Message
✓	Deploy metadata	Success. Metadata deployed.	
✓	Dimension City	Success. 116,295 rows transferred.	
✓	Dimension Customer	Success. 403 rows transferred.	
✓	Dimension Date	Success. 1,461 rows transferred.	
✓	Dimension Employee	Success. 213 rows transferred.	
✓	Dimension Stock Item	Success. 672 rows transferred.	
✓	Fact Sale	Success. 228,265 rows transferred.	

Stop Deployment

Close

40

