CIS 611: ENTERPRISE DATABASE AND DATA WAREHOUSING

Project: Multidimensional OLAP Cube using Adventure Works Data Warehouse
Overview:

A data warehouse is a centralized repository that stores data from multiple sources and converts those data into a cube/multidimensional data, which are very efficient for querying and analysing the data. This Multidimensional cube has two uses such as OLAP and Data Mining which are two technologies that have been used for Business Intelligence. OLAP database are divided into cubes (depending on the requirement, per se: user can keep entire database in a single cube, or they can keep each record in per cube like financial data, sales, production, etc like that). Using this it is used to analyse the data and predict the future based on the existing data. The data’s that are store in the cube are all pre-aggregated, so even if it retrieves millions of information from the database, it will still be efficient and fast.

Likewise, Data mining will also do prediction by analysing the data and generates who all are the possible buyers. This prediction is done by using few algorithms such as Microsoft clustering, Microsoft Decision Trees, Microsoft Linear Regression, Microsoft Linear Regression, Microsoft Naïve Bayes, Microsoft Time Series, Microsoft Sequence Clustering, Microsoft Logistic Regression (These are some most commonly and popular data mining algorithms used for analysing data and predicting data).

Tools required:

1. Microsoft SQL Server 2016 or latest
2. Microsoft SQL Server Management Studio 2017 or latest
3. Microsoft Visual Studio SSDT 2017 or latest

The database used in this project is Adventure Works 2014 (Data warehouse).

1. Adventure Works Database:

This is a fictional multinational bicycle company, with their headquarters located in Bothell, Washington and consists around 300 employees, 29 as sales representatives. The company’s sales are done via two methods (i). Online sales and, (ii). Reseller sales. The company mainly focuses on United States, United Kingdom, Canada, Australia, France (Europe) and few more places.

The products that are sold by this company are:
- **Bikes** - They have many bike types but their preferences on bike products are narrowed down to 3 bikes (Mountain, Road and, Touring) based sales.
- **Clothing** – The clothing products that are sold are short, jerseys, socks, etc.
- **Components** – They sell all the components that are required for repairing and spares for their manufactured bikes only.
- **Accessories** – They sell products like bike stand, rack stand, peddles, handle bar rods, etc.

The database contains all the details relating to sales, production, geography, etc. The sales database contains more information like mode of sales: internet or reseller. We can analyse the sales based on the Geography (country, city, state, region), Promotions (reseller, old products, excess products, no discount, etc), etc.
2. Build Database for Data mining, Multidimensional:

Firstly, we will need DW (Data warehouse) database. To get the database that is used here: https://msftdbprodsamples.codeplex.com/downloads/get/880664

1. Extract the downloaded database to C:\Program Files\Microsoft SQL Server\MSSQL13. (username) \MSSQL\Backup\n
Then Open SQL Server and connect to Database Engine.
Now, right click on database engine and select restore option and in that go to the specified path where we stored/saved the extracted file. Or, you can use the query to import the database:

```sql
USE [master]
RESTORE DATABASE [AdventureWorksDW2014] FROM DISK = N'C:\Program Files\Microsoft SQL Server\MSSQL13.(username)\MSSQL\Backup\AdventureWorksDW2014.bak'
WITH FILE = 1, NOUNLOAD, STATS = 5
GO
```

3. Building Cube:

3.1) Open Visual studio SSDT, click new project -> new project.
Then, in new project in installed look for Business Intelligence and in that select Analysis Services Multidimensional and Data Mining and, give any name.

After clicking, it will look like this.
3.2) Data Source:

Data source is used to import data from multiple sources and store it. Use the data’s stored here we will build the cube.

**Steps to create data source:**

1. Right click on data source and click new, then click **next** on that page.
2. In this page **“Click create a data source based on an existing or new creation”** (if you use it for the first time), now a dialog box like this pop up. In this pop up, under server name: write your server name mentioned from SQL server or you can just keep a “.” And keep authentication to windows then click **“Test Connection”** which will show if it was success or not.
After this is done, next important thing is to select the database that you want to use. Then click ok.
Impersonation Information will pop up in that select "Use a specific Windows user name and password", in that type your Windows ID and password then click next.

Then give data source a name and click finish.

2.3) Data Source View:
• Now right click on data source view and click “new Data source view”, then click next on the page that that pops up. In this it will show relational data sources that are available from data source, select the appropriate database and click next.

• Then a new page called as select tables and views will come. Select necessary tables but I have imported all the data and click ok. In the next page give a name and finish it.
Once it is done, a star schema will be formed where it will point to the relationships between each table.

**Star Schema, Fact Tables and Dimension Tables:**

- **Star Schema:** This is the simplest way to see how each table to related to other tables and by which column. It is called as star schema as the table in centre points to other tables gives a look like star.
- **Fact Table:** It contains **pre-aggregated data** (especially with **numeric data**, and fact table is otherwise also known as **Measures**.)
- **Dimension table:** This contains all kinds of data in it, but it does not have pre-aggregated data for numeric like Fact Table. This table allows the user to create hierarchy for each table as per the requirement.

### 3.4) Cube:

OLAP Cube is used to storage MDX (Multidimensional) data, by using this we will analyse the data and predic the data patterns.

**Steps for Building and Deploying a Cube:**
• Right click on the cube and select “**New Cube**” and a wizard pops up in that click next.

• **Select Creation Method:** In this cube select “**Use existing table**” and click next.

![Cube Wizard](image)

**Select Creation Method**
Cubes can be created by using existing tables, creating an empty cube, or generating tables in the data source.

- [ ] Use existing tables
- Create an empty cube
- Generate tables in the data source

Template:

(No)

Description:
Create a cube based on one or more tables in a data source.

• **Select Measures Group Tables:** Measures is not but data that are used for calculation and in this process, it will collect all the data and pre-aggregate it.

  -> Select data that should be selected, or you can also click suggest (by default, it will select all the tables that contains numeric data type fields.) If it missed any data that needs to be measured, we can select that or go with that.

• **Select Measures:** By default, everything will be selected. Just click **Next**.

• **Select New Dimensions:** By default, all the dimension tables will be select. Just click **next**.
Completing the Wizard: Give the cube a name and click finish.

![Solution Explorer](image1)

After it is completed, this is how it will look.

Creating Hierarchies:

- In dimensions, double "**Dim Date.dim**" it will open a new tab like this:

![Attribute Relationships](image2)

- In the right most column "**Data Source View**", select dates that are required. (I have selected everything in that except “French and
Spanish” related and FullDateAlternateKey) After selecting the data, drag and drop them in “Attributes”. 
• Now create hierarchies using attributes, select data from attributes and drop in hierarchy column. This will create a new table like this.

In that rename the hierarchy, how it is required.

Now, rename English month name to month, so it is easier to understand.

Attention symbol next to calendar is seen because it does not have any relationship within the hierarchy created.

To create it, go to “Attribute Relationships” you will see like this.

In this, we can see that data key is hierarchy for everything that is the reason for us to get the attention symbol. Now what we will do is,
select the first key and point it to the next.

Point it like this.
Once this is done Click save and a dialog box will pop us, click proceed.
Follow this procedure for everything and create hierarchies.

**Deploy the Cube:**
After clicking on save all, go to “Solution Explorer”, right click the project name (AdventureWorksDW2014) and select build first and see if you get any error or else it will say it is succeeded. Then, right click the project name again and deploy it now.
If you are editing in Dimensions after the deployment, you should not deploy it again, without doing process… (in case you forget to do so, while deploying even it will ask: if you want to process, click proceed for it process.). To process..., right click on the project name and click process.

While deployment, we will get two types of error:

1. **Security:** Enter correct windows name and password, if it is correct then it will work. If it was not given at the time of creating follow then follow this: **Double-click on data source**
In this, click on Impersonation Information: In that select windows authentication and then enter the details.

2. **SQL SERVER Server Agent** is turned off. To “Turn it” on by going to the database in SQL Server Management Studio and connect it, then in the **Object Explorer** open click the + then, it will show up like this.
Right click on, SQL Server Agent and click start. Then a pop like this will come, click **YES** in it.

If you deploy it now, then it will get deployed.

Now open the SQL Server Management Studio and before connecting the database to database engine, click on database engine a drop-down list will come in that select Analysis service and then click connect.
4. MDX Querying:

After having connected the server type with Analysis Services. Go to Object Explorer, in that click plus symbol on Databases and select the deployed database. Now right click on the deployed database and go to New Query -> MDX.

This is how it will look:

![MDX Query Example](image1)

Now we will do the prediction using MDX Queries:

1. Show Product wise sales by both internet sales amount and Reseller Sales amount for the fiscal quarter and month/ year.
   
   **Query:**
   
   SELECT NON EMPTY {[Measures].[Internet Sales Amount],[Measures].[Reseller Sales Amount],[Measures].[Sales Amount]} on columns, non empty crossjoin ([Product].[Category].[Category].[Date].[Fiscal Quarter of Year].[Fiscal Quarter of Year].[Date].[Calendar].[Month]) on rows from [Adventure Works];

   **Output:**
Prediction: We can see the highest Promotional sales were reseller sales, assuming this we can add more products for Sale in reseller sales than on internet sales.

(Hint: normally we can not add more tables with different dimension, so if we can use crossjoin to add multiple tables)
2. Show the cities in United States where the total sales amount is more than 5000000 (5 Million) for all the product. (Using VisualTotals and without VisualTotals)

Query:

select [Geography].[City].[City] on columns,
order(visualtotals (filter([Product].[Category].members, [Measures].[Sales Amount] > 5000000)),[Measures].[Sales Amount] , basc) on rows
from [Adventure Works]
where [Geography].[Country] & [United States];

Output:

Without VisualTotals:
Using VisualTotals:
**Prediction:** Houston had the highest sales and will also have the highest sales for the next year.

*(Hint: Without visualTotals-> When we filter the data and list hets filtered and displays list that meets our requirements and the final result/value will not match the value that because the total is not calculated as it is listed, it fetches the value from cube where all the details are pre-aggregated. Using VisualTotals-> In this It calculates the final result and displays.)*

3. **What is the total reseller order count and reseller amount for the products (Chains, Breaks, Crank set, Handle Bars, Road Bikes, Pumps, Locks, Cleaners, Bottles ad Cages, Bike Stand and Racks), with promotional offers such (i). Seasonal Discount, (ii). Excess Inventory, , where the customer is Male, Year (2012,2013,2014) in all State-Province.**

**Query:**
SELECT NON EMPTY {
[Measures].[Reseller Order Count],[Measures].[Reseller Sales Amount]} ON COLUMNS,
NON EMPTY {{[Date].[Calendar Year].[Calendar Year].ALLMEMBERS,[Geography].[Geography].[State-Province].ALLMEMBERS}} ON ROWS
FROM
(SELECT
{([Promotion].[Promotion Type].&[Seasonal Discount],[Promotion].[Promotion Type].&[Excess Inventory])} ON COLUMNS
FROM
(SELECT
FROM
(SELECT
{([Customer].[Gender].&[M])} ON COLUMNS
FROM
(SELECT
)
Output:
Prediction: California has the highest reseller sales along with promotions, where men has bought bike along with multiple accessories and components, based on this we can that California only will have highest reseller sales, so we can provide them more stock during that offer.
References:

- Create tables and followed procedure from this website:
  - https://docs.microsoft.com/en-us/sql/analysis-services/tabular-modeling-adventure-works-tutorial
- Code Project:
  - http://www.codeproject.com/Articles/710387/Learn-to-Write-Custom-MDX-Query-First-Time
  - http://www.codeproject.com/Articles/658912/Create-First-OLAP-Cube-in-SQL-ServerAnalysis-Serv