Introduction

Business Intelligence has become a buzzword in recent years as a support to decision making. Today we can find several database systems which include data warehousing, online analytical processing (OLAP), and data mining technologies. Data warehousing provides an efficient storage, maintenance, and retrieval of data. OLAP is a service that provides a way to create ad hoc queries against the data warehouse in order to answer important business questions. Data mining is a disciple comprising of several algorithms for discovering knowledge in a large bulk of data.

In order to build a data warehouse solution, we need to model a consistent architecture where the operational data will fit well in an integrated and enterprise-wide view as well as to take into consideration a handful implementation strategies to provide a high quality application. The design and implementation of a data warehouse solution sometimes is a very complex challenge in theory and practice. In this article, I will cover the main principles and techniques to design and implement a data warehouse providing my own experience in such an overwhelming challenge.

This is the fourth article of a series of articles that I want to write in order to share my knowledge and experience in this subject matter. In this part and the previous one, we’ve seen how to implement the technical solution for a data warehouse.

Implementing the technical solution

In this section, we’ll continue working on the technical solution.

After the steps on the previous article, you must click on Next button, and the Select Measures page will appear which displays the measures associated to the fact table. Click on Next button, and the Detecting Hierarchies page process each dimension in order to find potential hierarchies between the attributes inside the dimension. Click on Next button, and you can review the found hierarchies for each dimension. In this case, you can see that we have found two hierarchies for the TimePeriodDim dimension (see Figure 1).
Click on Finish button in order to create the cube. Then the Cube editor will be displayed (see Figure 2).
You can add new measures to the cube by clicking on the corresponding measure group on the Measures windows and selecting the New Measure option (see Figure 3).

Then the New Measure dialog box will enable selecting the attribute and the type of measure operation to apply it. In this case, we add a new measure to get the maximum values of the SalesVolumes attribute in the SalesFact fact table.

As well, you can edit the dimension by right-clicking on the particular dimension on the Dimensions windows in the Cube editor or by navigating the tree on the Solution Explorer until the particular dimension node on the Dimensions parent node. Then the Dimension editor will be displayed.

Let's add a Subcategory-Color hierarchy to the Product dimension by opening the Dimension editor on the Product dimension. First, drag and drop the Subcategory attribute from the Attributes windows into the Hierarchies and Level pane in the Dimension editor (see Figure 4).
Then add drag and drop the Color attribute inside the new hierarchy and below the Subcategory attribute (see Figure 5).
Now it's the time to deploy the solution from the development environment to the production environment in order for the end users to consume reports from the cube. First, we need to configure the deployment scenario by going into the project properties in the *Project* *ProjectName* Properties menu. Then select the Deployment node and set the Server property to the name of your production server (see Figure 6).
Once you have deployed your solution, you can create custom reports for displaying the information to your users. You have several options for the data warehouse presentation layer. You can create the reports using Microsoft technologies such as Excel pivot tables for Windows applications and Office Web Components for Web applications. You can also find several third-party tools from vendors such as Cognos, ProClarity, Microstrategy, and Business Objects.

You can also write your own queries against the cubes using a language named Multidimensional Expressions (MDX). MDX is for OLAP databases like SQL is for relational databases. Using this language, you can build your own data warehouse front-end components and customize views produced by third-party tools.

Now we're going to discuss how to use Pivot tables in Microsoft Excel 2007 to produce OLAP custom reports as the main client tool. You can think of a pivot table as a way to create summary reports for analyzing your data. The main structure of a pivot table is:

1. **PivotTable Field List.** This section in the top right displays the fields in your spreadsheet. You may check a field or drag it to a quadrant into the lower section.
2. The lower right quadrants. This area defines where and how the data shows on the pivot table. You can have a field show in either a column or a row. You may indicate if the data should be counted, summed, averaged, filtered and so on.
3. The red outlined area to the left is the result of your selection from (1) and (2).

Let's create an Excel Pivot Table to display the Test DW cube created before. The first step is to create a connection to the Analysis Service multidimensional database. From the Data ribbon, choose From Analysis Services option from the From Other Sources drop-down box (see Figure 7). Excel will display the Data Connection Wizard in order to configure your connection string and establish a link to the server.
In the first page of the Wizard, enter the name of the server as well as the authentication information (see Figure 8).

Figure 7

Connect to Database Server
Enter the information required to connect to the database server.

1. Server name: [my_analysis_server]

2. Log on credentials
   - Use Windows authentication
   - Use the following User Name and Password

   User Name: [ ]
   Password: [ ]

[Cancel] [Back] [Next] [Finish]

Figure 8

Click Next button, and select the target database from the drop-down box. When you select the TestAS database, then a list of the underlying tables and cubes is shown below in the drop-down menu. Then choose the TestDW cube (see Figure 9).
Click on the Next button in order to add some descriptive information related to the data connection (see Figure 10).

And finally, click on the Finish button. After that, you immediately see the Import Data dialog box, and then select the PivotTable report option, and the PivotTable object will be display in the current worksheet (see Figure 11).
As you can see that the measures and its attributes are listed under the Sigma icon node. You can see the dimensions and its attributes which are listed under the table icon nodes. In the case of the Product dimension, you can see the attributes such as Color, Product Business ID, Product Dim and Subcategory Name as well as the defined Subcategory Name – Color hierarchy (see Figure 12).

Now it's time to analyze the Sales Volumes and ShippedUnits by Customer, Product, and Region. From the PivotTable Field List windows, click on the Sales Volumes and ShippedUnits measures on the Sales Fact node.
You can see in the lower portion of the window on the Values pane, that these measures are added. Now click on the CustomerType attribute of the Customer dimension and this attribute is added to the Row Labels pane on the lower portion of the window. The worksheet also displays the measures group by the CustomerType attribute (see Figure 13).

![Figure 13](image)

Now let’s enhance the report by comparing by the Product Subcategory Name-Color hierarchy and Region by clicking on the SubcategoryName-Color Hierarchy folder on the Product dimension and clicking on the Region Name attribute on the Region dimension. The hierarchy and attribute are added to the Row Labels pane on the lower region of the window. Now you can drill-down or rollup to see the summaries of the measures (see Figure 14).

![Figure 14](image)
Conclusion

In this series of articles, I've covered the principles, methodology, techniques and the main technologies for the development of a data warehouse solution using Microsoft related technologies such as Microsoft SQL 2005 Analysis Services and Microsoft Excel 2007.

Figure 14

FILED UNDER DATA WAREHOUSE

3 Responses to Designing and implementing a Data Warehouse. Part 4

Pingback: MVP Factor

elvincheng says:
October 27, 2010 at 2:03 pm

Great posts.

Reply
Thanks, Elvin.

Reply