Praktek Installasi & Persiapan DW dgn SQL Server

# Preparation

## Step 1: Install Software

The lessons in this tutorial assume that you have the following software installed. All of the following software is installed using SQL Server installation media. For simplicity of deployment, you can install all of the features on a single computer. To install these features, run SQL Server Setup and select them from the Feature Selection page.

For more information, see [Install SQL Server 2012 from the Installation Wizard (Setup)](http://msdn.microsoft.com/en-us/library/6ad23de1-2bab-4933-9122-c09f5565028d(SQL.110)).

* Database Engine
* Analysis Services
* SQL Server Data Tools (SSDT)
* SQL Server Management Studio

Optionally, consider installing Excel to browse your multidimensional data as you proceed through the tutorial. Installing Excel enables the Analyze in Excel feature that starts Excel using a PivotTable field list that is connected to the cube you are building. Using Excel to browse data is recommended because you can quickly build a pivot report that lets you interact with the data.

## Step 2: Install Databases

An Analysis Services multidimensional model uses transactional data that you import from a relational database management system. For the purposes of this tutorial, you will use the following relational database as your data source. • AdventureWorksDW2012 – This is a relational data warehouse that runs on a Database Engine instance. It provides the original data that will be used by the Analysis Services databases and projects that you build and deploy throughout the tutorial. To install this database, do the following:

1. Download the [AdventureWorkDW2012](http://msftdbprodsamples.codeplex.com/releases/view/55330) database from the product samples page on codeplex. The database file name is AdvntureWorksDW2012\_Data.mdf. The file should be in the Downloads folder on your computer.
2. Copy the AdventureWorksDW2012\_Data.mdf file to the data directory of the local SQL Server Database Engine instance. By default, it is located at C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL\Data.
3. **Start SQL Server Management Studio** and **connect to the Database Engine instance**.
4. Right-click Databases, click **Attach**.
5. Click **Add**.
6. **Select the AdventureWorksDW2012\_Data.mdf** database file and click OK. If the file is not listed, check the C:\Program Files\Microsoft SQL Server\MSSQL11.MSSQLSERVER\MSSQL\Data folder to be sure the file is there.
7. In database details, **remove the Log file entry**. The sample download does not include a log file. A new log file will be created automatically when you attach the database. Select the **log file** and **click Remove**, and then **click OK to attach** just the primary database file.

## Step 3: Grant Database Permissions

The sample projects use data source impersonation settings that specify the security context under which data is imported or processed. By default, the impersonation settings specify the Analysis Services service account for accessing the data. To use this default setting, you must ensure that the service account under which Analysis Services runs has data reader permissions on the **AdventureWorksDW2012** database.

**Note**: *For learning purposes, it is recommended that you use the service account impersonation option and grant data reader permissions to the service account in SQL Server. Although other impersonation options are available, not all of them are suitable for processing operations. Specifically, the option for using the credentials of the current user is not supported for processing.*

Determine the service account. You can use SQL Server Configuration Manager or the Services console application to view account information. If you installed Analysis Services as the default instance, using the default account, the service is running as **NT Service\MSSQLServerOLAPService**.

1. In Management Studio, connect to the database engine instance.
2. **Expand the Security folder**, **right-click Logins** and select **New Login**.
3. On the **General page**, in **Login name**, type NT Service\MSSQLServerOLAPService (or type the name you want or whatever account the service is running as).
4. **Click User Mapping**.
5. Select the **checkbox** next to the **AdventureWorksDW2012** database. Role membership should automatically include **db\_datareader** and **public**. Click **OK** to accept the defaults.

# Lesson 1: Defining a Data Source View within an Analysis Services Project

## Creating an Analysis Services Project

In the following task, you use **SQL Server Data Tools (SSDT)** to create a new Analysis Services project named Analysis Services Tutorial, based on the Analysis Services Project template. A project is a collection of related objects. Projects exist within a solution, which includes one or more projects.

### Procedure to Create new a new analysis services project:

1. Click Start, point to All Programs, point to Microsoft SQL Server 2012, and then click **SQL Server Data Tools**. The Microsoft Visual Studio development environment opens.
2. On the Start page of Visual Studio, click **New Project**.
3. In the New Project dialog box, in the Installed Templates pane, expand **Business Intelligence**, and then select **Analysis Services**. Choose the **Analysis Services Multidimensional and Data Mining** Project template. Notice the default project name, location, and the default solution name are generated in the bottom of the dialog box. By default, a new directory is created for the solution.
4. **Change** the **project Name** to Analysis Services Tutorial, which also changes the Solution name box, and then click **OK**.

You have successfully created the Analysis Services Tutorial project, based on the Analysis Services Multidimensional and Data Mining Project template, within a new solution that is also named Analysis Services Tutorial.

## Defining a Data Source

After you create an Analysis Services project, you generally start working with the project by defining one or more data sources that the project will use. When you define a data source, you are defining the connection string information that will be used to connect to the data source.

### Procedures to define a new data source:

1. In **Solution Explorer** (on the right of the Microsoft Visual Studio window), **right-click Data Sources**, and then click **New Data Source**.
2. On the Welcome to the Data Source Wizard page of the Data Source Wizard, **click** **Next** to open the Select how to define the connection page.
3. On the Select how to define the connection page, you can define a data source based on a new connection, based on an existing connection, or based on a previously defined data source object. In this tutorial, you define a data source based on a **new connection**. Verify that Create a data source based on an existing or new connection is selected, and then **click New**.
4. In the Connection Manager dialog box, you define connection properties for the data source. In the Provider list box, verify that **Native OLE DB\SQL Server Native Client 11.0** is **selected**. Analysis Services also supports other providers, which are displayed in the Provider list.
5. In the Server name text box, type **localhost**.

To connect to a named instance on your local computer, type **localhost\<instance name>**.

To connect to the specific computer instead of the local computer, type **the computer name** or **IP address**.

1. Verify that Use **Windows Authentication** is **selected**. In the Select or **enter a database name** list, select **AdventureWorksDW2012**.
2. Click **Test Connection** to test the connection to the database.
3. Click **OK**, and then click **Next**.
4. On the Impersonation Information page of the wizard, you define the security credentials for Analysis Services to use to connect to the data source. Impersonation affects the Windows account used to connect to the data source when Windows Authentication is selected. Analysis Services does not support impersonation for processing OLAP objects. Select Use the service account, and then click Next.
5. On the Completing the Wizard page, accept the default name, Adventure Works DW 2012, and then click **Finish** to create the new data source. Note

**Note**: *To modify the properties of the data source after it has been created, double-click the data source in the Data Sources folder to display the data source properties in Data Source Designer.*

## Defining a Data Source View

After you define the data sources that you will use in an Analysis Services project, the next step is generally to define a data source view for the project. A data source view is a single, unified view of the metadata from the specified tables and views that the data source defines in the project. Storing the metadata in the data source view enables you to work with the metadata during development without an open connection to any underlying data source.

In the following task, you define a data source view that includes five tables from the AdventureWorksDW2012 data source.

### Procedures to define a new data source view:

1. In Solution Explorer (on the right of the Microsoft Visual Studio window), **right-click Data Source Views**, and then click **New Data Source View**.
2. On the Welcome to the Data Source View Wizard page, click **Next**. The Select a Data Source page appears.
3. Under **Relational data sources**, the **Adventure Works DW 2012** data source is selected. Click **Next**.

**Note**: *To create a data source view that is based on multiple data sources, first define a data source view that is based on a single data source. This data source is then called the primary data source. You can then add tables and views from a secondary data source. When designing dimensions that contain attributes based on related tables in multiple data sources, you might need to define a Microsoft SQL Server data source as the primary data source to use its distributed query engine capabilities.*

1. On the **Select Tables and Views** page, select tables and views from the list of objects that are available from the selected data source. You can filter this list to help you select tables and views.

**Note**: *Click the maximize button in the upper-right corner so that the window covers the full screen. This makes it easier to see the complete list of available objects.*

In the **Available objects list**, select the following objects. You can select multiple tables by clicking each while holding down the CTRL key:

* + **DimCustomer (dbo)**
  + **DimDate (dbo)**
  + **DimGeography (dbo)**
  + **DimProduct (dbo)**
  + **FactInternetSales (dbo)**

1. Click **>** to add the selected tables to the **Included** **objects** list.
2. Click **Next**.
3. In the Name field, make sure **Adventure Works DW 2012** displays, and then click **Finish**.

The Adventure Works DW 2012 data source view appears in the Data Source Views folder in Solution Explorer. The content of the data source view is also displayed in Data Source View Designer in SQL Server Data Tools (SSDT). This designer contains the following elements:

* + A Diagram pane in which the tables and their relationships are represented graphically.
  + A Tables pane in which the tables and their schema elements are displayed in a tree view.
  + A Diagram Organizer pane in which you can create subdiagrams so that you can view subsets of the data source view.
  + A toolbar that is specific to Data Source View Designer.

1. **To maximize** the Microsoft Visual Studio development environment, click the **Maximize** button.
2. To view the tables in the Diagram pane at 50 percent, click the Zoom icon on the Data Source View Designer toolbar. This will hide the column details of each table.
3. **To hide Solution Explorer**, click the **Auto Hide** button, which is the pushpin icon on the title bar. To view Solution Explorer again, position your pointer over the Solution Explorer tab along the right side of the development environment. To unhide Solution Explorer, click the Auto Hide button again.
4. If the windows are not hidden by default, **click** **Auto Hide** on the title bar of the Properties and Solution Explorer windows.

You can now view all the tables and their relationships in the **Diagram** pane.

Notice that there are three relationships between the FactInternetSales table and the DimDate table. Each sale has three dates associated with the sale: an order date, a due date, and a ship date. To view the details of any relationship, double-click the relationship arrow in the Diagram pane.

## Modifying Default Table Names

You can change the value of the FriendlyName property for objects in the data source view to make them easier to notice and use. In the following task, you will change the friendly name of each table in the data source view by removing the "Dim" and "Fact" prefixes from these tables. This will make the cube and dimension objects (that you will define in the next lesson) easier to notice and use.

You can also change the friendly names of columns, define calculated columns, and join tables or views in the data source view to make them easier to use.

### Procedures to modify default table names

1. In the **Tables** pane of **Data Source View Designer**, **right-click** **the FactInternetSales** table, and then **click** **Properties**.
2. If the Properties window on the right side of the Microsoft Visual Studio window is not displayed, **click** the **Auto Hide** button on the title bar of the Properties window so that this window remains visible.

It is easier to change the properties for each table in the data source view when the Properties window remains open. If you do not pin the window open by using the Auto Hide button, the window will close when you click a different object in the Diagram pane.

1. Change the **FriendlyName** property for the **FactInternetSales** object to **InternetSales**.

When you click away from the cell for the FriendlyName property, the change is applied. In the next lesson, you will define a measure group that is based on this fact table. The name of the fact table will be InternetSales instead of FactInternetSales because of the change you made in this lesson.

1. **Click** **DimProduct** in the Tables pane. In the Properties window, change the **FriendlyName** property to **Product**.
2. **Change the FriendlyName** property of each remaining table in the data source view in the same way, to remove the "Dim" prefix.
3. When you have finished, click the **Auto Hide** button to hide the Properties window again.
4. On the **File** **menu**, or on the toolbar of SQL Server Data Tools, **click Save All** to save the changes you have made to this point in the Analysis Services Tutorial project. You can stop the tutorial here if you want and resume it later.