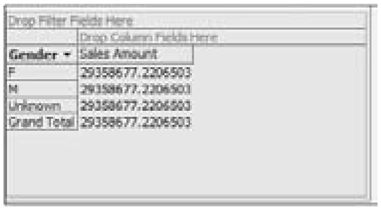
Лекция 11

Тема « Relationship Types »

Relationship Types

Six different types of relationships can exist between a dimension and a measure group: No Relationship, Regular, Fact, Referenced, Many - to - Many, and Data Mining. In Figure 6 - 10 you see three of the six relationship types: No Relationship, Referenced, and Regular. Cells corresponding to a specific dimension and measure group can have an attribute specified that indicates that the dimension type is Regular. Further, such attributes can be used in the join condition between the dimension and the measure group. Often this attribute is the key attribute of the dimension and is called the *granularity attribute* . The granularity attribute can be an attribute that is above the key attribute of the dimension. When you browse a dimension along with measures of a measure group where the dimension and measure group have a regular relationship, Analysis Services aggregates the data appropriately. If you have your granularity attribute above the key attribute of your dimension, it is critical that you define appropriate attribute relationships in your dimension to make sure the data that is getting aggregated is accurate. The relationship between Dim Customer and Fact Internet Sales measure group is a regular relationship. The granularity attribute is shown in the cell intersecting the dimension and measure group as shown in Figure 6 - 10 .

Cells that are shaded gray indicate there is no relationship between the dimension and measure group. Whenever there is no relationship between a dimension and measure group, the measure group property IgnoreUnrelatedDimension controls the results of queries involving any hierarchy of that dimension and any measure from the measure group. The measure values will either be null (IgnoreUnrelatedDimension=False) or the same value for each member of the dimension (IgnoreUnrelatedDimension=True). For example, there is no relationship between the dimension [Dim Employee] and the [Fact Internet Sales] measure group. If you browse the Gender hierarchy of [Dim Employee] and the measure [Sales Amount], you see that the measure values for each member of Gender hierarchy are the same value as the Grand Total as shown in Figure 6 - 11 . This is because the IgnoreUnrelatedDimension value is set to True by the Cube Wizard as a default. You learn more about properties of measure groups and measures later in this chapter.



Gender	Sales Amount
F	29358677.2206503
M	29358677.2206503
Unknown	29358677.2206503
Grand Total	29358677.2206503

When a table is used as both a fact and dimension table, it constitutes a unique relationship between the dimension and measure group called the *fact* relationship. The relationship is similar to that of the regular dimension, but specifying it as a fact dimension helps improve query performance for a certain class of MDX queries. You learn more about fact dimensions in Chapter 9 .

Typically there is a one - to - many relationship between a fact and a dimension member for regular relationships. When you have a one - to - one relationship between a fact and a dimension member, you typically have a fact relationship. When there is a many - to - many relationship between a fact and a dimension member, the dimension member has a one - to - many relationship with various facts and a single fact is associated with multiple dimension members. The definition for a many - to - many relationship can be well understood via an example. Assume you have a fact table (for sales of books) that is related to a dimension table containing author information. There is another fact table that contains authors ' salary information, which is related to the Authors dimension table as well as the geographical information of the publisher who is paying the authors. In this example you have a one - to - many relationship between authors and books. The salary fact data is related to the publisher ' s geographical information and the authors. If you want to analyze the book sales based on the geographical information of the publisher, the Geography dimension of publishers acts as a many - to - many relationship with the fact Book Sales. You learn about the usage of fact and many - to - many relationships in Chapter 9 .

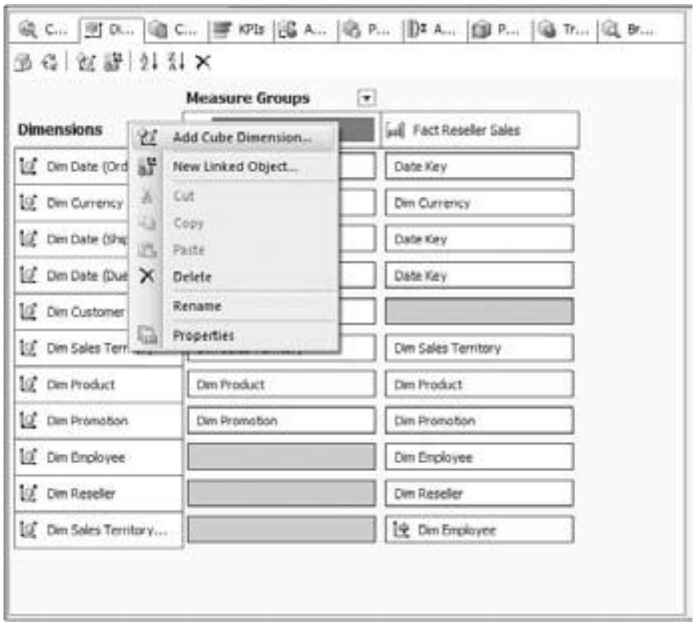
Data Mining dimensions are another item type in the list of relationships; these are used to establish linkage between a cube and a dimension created from a Data Mining model. You learn more about this in Chapters 9 and 16 .

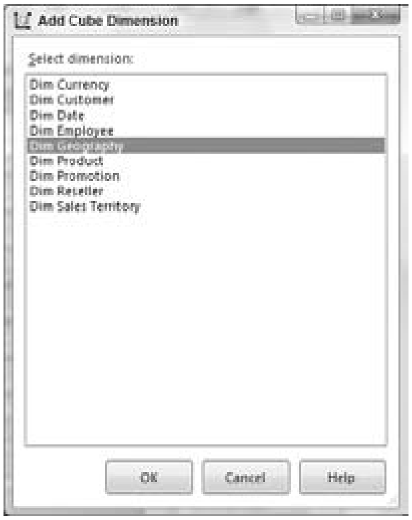
When a dimension is related to the fact data through another dimension, you define this indirect relationship between the measure group and the dimension as a *reference relationship* . In Figure 6 - 10 the Dim Sales Territory (Dim Employee – Sales Territory) dimension is related to the Fact Reseller Sales measure group through the Employee dimension. The icon at the intersection of the dimension and measure group indicates this reference relationship. You might also recall that you added the Dim Geography dimension in the Select Existing Dimensions page of the Cube Wizard. However, the Cube Wizard was not smart enough to figure out there is a relationship between the Fact tables and the Dim Geography dimension table through other dimension tables. Hence the Cube Wizard did not add the Dim Geography dimension as a cube dimension. Because the relationship between the Dim Geography dimension and the measure groups in the Adventure Works DW cube is through another dimension, you can say that there is an indirect relationship between the Dim Geography dimension and the measure groups.

Follow these steps to add the Dim Geography dimension to the cube and establish the reference relationship:

1. To add the Dim Geography database dimension to the cube, right - click in the Dimension pane of the Dimension Usage tab and select Add Cube Dimension, as shown in Figure 6 - 12 .

2. A dialog showing all the shared dimensions within the project launches, as shown in Figure 6 - 13 . Select the Dim Geography dimension and click OK.



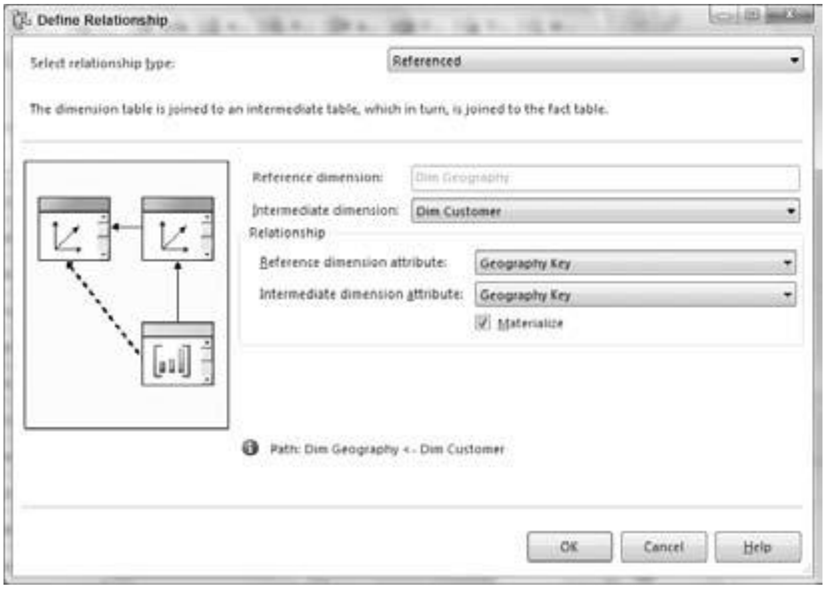


The cube editor is unable to identify a relationship through an attribute between the existing measure groups and the Geography dimension and as a result leaves the relationship definition up to you. There exists an indirect relationship between the [Dim Geography] dimension and [Fact Internet Sales] measure group through the [Dim Customer] dimension. There is an indirect relationship between the [Dim Geography] dimension and the Fact Reseller measure group through the [Dim Reseller] dimension. You need to define these reference relationships.

3. To define the relationship between [Dim Geography] and the [Fact Internet Sales] measure group, select the corresponding cell in the matrix and you will see an ellipsis in that cell. Click the ellipsis (“ . . . ”).

4. This opens the Define Relationship dialog shown in Figure 6 - 14 . Select Referenced from the Select relationship type drop - down list box. The [Dim Geography] dimension forms an indirect or reference relationship with the [Fact Internet Sales] measure group through the [Dim Customer] dimension. You define the intermediate dimension through the Intermediate Dimension field in the wizard. Once you have defined the intermediate dimension, you need to select the attributes that are involved in the join of the relationship. The “ Reference dimension attribute ” is the attribute in the reference dimension that is used in the join between the intermediate dimension ([Dim Customer]) and the reference dimension ([Dim Geography]). The “ Intermediate dimension attribute ” is the attribute of the intermediate dimension that is involved in the join between the reference dimension and the intermediate dimension. Define the Intermediate dimension as [Dim Customer], Reference dimension attribute as [Geography Key], and Intermediate dimension attribute as [Geography Key] as shown in Figure 6 - 14 and click OK. In Figure 6 - 14 you see a checkbox with the text Materialize. This checkbox is enabled by default in SQL Server Analysis Services 2008. By enabling this checkbox you are ensuring Analysis Services will build appropriate indexes to get improved query performance while querying fact data along with reference dimension hierarchies.

5. Similar to step 4, establish a referenced relationship between the [Dim Geography] dimension and the [Fact Reseller Sales] measure group through the [Dim Reseller] dimension. Once you have completed specifying the relationship between [Dim Geography] and he two measure groups of the cube, your Dimension Usage tab will resemble Figure 6 - 15 .



Define Relationship

Select relationship type: Referenced

The dimension table is joined to an intermediate table, which in turn, is joined to the fact table.

Reference dimension: Dim Geography

Intermediate dimension: Dim Customer

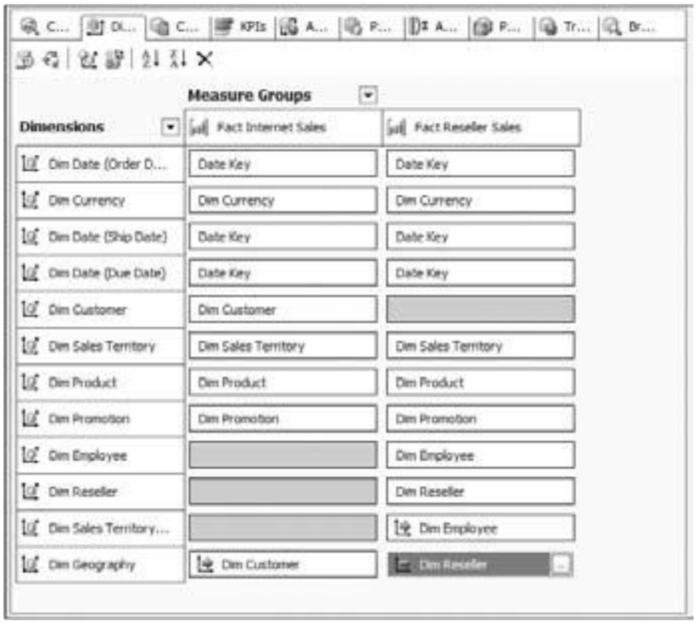
Relationship

Reference dimension attribute: Geography Key

Intermediate dimension attribute: Geography Key

☑ Materialize

Path: Dim Geography <- Dim Customer

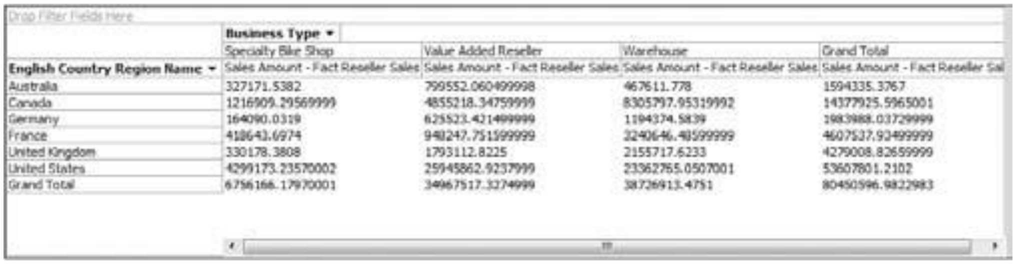


Dimensions	Measure Groups Fact Internet Sales	Fact Reseller Sales
Dim Date (Order D...	Date Key	Date Key
Dim Currency	Dim Currency	Dim Currency
Dim Date (Ship Date)	Date Key	Date Key
Dim Date (Due Date)	Date Key	Date Key
Dim Customer	Dim Customer	
Dim Sales Territory	Dim Sales Territory	Dim Sales Territory
Dim Product	Dim Product	Dim Product
Dim Promotion	Dim Promotion	Dim Promotion
Dim Employee		Dim Employee
Dim Reseller		Dim Reseller
Dim Sales Territory...		Dim Employee
Dim Geography	Dim Customer	Dim Reseller

The reference relationship between a dimension and a measure group is indicated by an arrow pointing to the intermediate dimension as shown in Figure 6 - 15 . This graphical view of the reference relationship helps you identify the type of relationship between a dimension and measure group when you are looking at the Dimension Usage tab of the cube editor. Similar graphical representations are available for fact, many - to - many, and Data Mining dimensions, and you learn about these relationships later in Chapters 9 and 16 .

Browsing Reference Dimensions

Having added the [Dim Geography] dimension as the reference dimension to the cube, assume you want to analyze the Reseller Sales based on different business types in various countries. To do so you need redeploy and process the cube with the changes. Then go to the Cube Browser and drag and drop the [English Country Region Name] hierarchy from the [Dim Geography] dimension to the rows, the [Business Type] hierarchy of the [Dim Reseller] dimension to the columns, and the measure [Sales Amount – Fact Reseller Sales] of the [Fact Reseller Sales] measure group to the details area. You can now analyze the Sales data based on the business type in each country, as shown in Figure 6 - 16 . Based on this sales knowledge, the costs associated with the products, and your business goals, you can strategically promote the business type yielding the maximum profit for your company. Reference dimensions help you to analyze fact data even though they are not directly related to the facts.



	Business Type			
	Specialty Bike Shop	Value Added Reseller	Warehouse	Grand Total
English Country Region Name	Sales Amount - Fact Reseller Sales	Sales Amount - Fact Reseller Sales	Sales Amount - Fact Reseller Sales	Sales Amount - Fact Reseller Sal
Australia	327171.5382	799552.060499998	467611.778	1594335.3767
Canada	1216909.29569999	4855218.34759999	8305797.95319992	14377925.5965001
Germany	164090.0319	625523.421499999	1194374.5839	1983988.03729999
France	418643.6974	948247.751599999	3240646.48599999	4607537.92499999
United Kingdom	330178.3808	1793112.8225	2155717.6233	4279008.82659999
United States	4299173.23570002	25945862.9237999	23362765.0507001	53607801.2102
Grand Total	6756166.17970001	34967517.3274999	38726913.4751	80450596.9822983

The OWC sends the following statements and queries to retrieve data for analyzing the reseller sales fact of various business types across various countries of the resellers. The OWC creates sets for the members on the columns and rows of OWC and then queries the facts added to the detail data along with the sets.

```
Drop visual totals for [Adventure Works DW]
CREATE SESSION
SET [Adventure Works DW].[{76D2D6C7-D50B-4C12-8DBF-DA53595646F5}
Pivot24Axis0Set0] AS
‘
{
{ [Dim Reseller].[Business Type].[All] },
AddCalculatedMembers([Dim Reseller].[Business Type].[Business
Type].MEMBERS)
}
‘
SET [Adventure Works DW].[{76D2D6C7-D50B-4C12-8DBF-DA53595646F5}
Pivot24Axis1Set0] AS
‘
{
{ [Dim Geography].[English Country Region Name].[All] },
AddCalculatedMembers([Dim Geography].[English Country Region Name].
[English Country Region Name].MEMBERS)
}
‘
SELECT
NON EMPTY [{76D2D6C7-D50B-4C12-8DBF-DA53595646F5}Pivot24Axis0Set0]
DIMENSION PROPERTIES MEMBER_NAME, PARENT_UNIQUE_NAME ON COLUMNS,
NON EMPTY [{76D2D6C7-D50B-4C12-8DBF-DA53595646F5}Pivot24Axis1Set0]
DIMENSION PROPERTIES MEMBER_NAME, PARENT_UNIQUE_NAME ON ROWS,
{
[Measures].[Sales Amount - Fact Reseller Sales]
}
ON PAGES
FROM [Adventure Works DW]
CELL PROPERTIES VALUE, FORMATTED_VALUE, FORE_COLOR, BACK_COLOR
DROP SET [Adventure Works DW].[{76D2D6C7-D50B-4C12-8DBFDA53595646F5}
Pivot23Axis0Set0]
DROP SET [Adventure Works DW].[{76D2D6C7-D50B-4C12-8DBFDA53595646F5}
Pivot23Axis1Set0]
```

The OWC provides you with the option of slicing the data you are analyzing. Therefore OWC creates the MDX statements to create sets within the specified session. It then queries the multidimensional data on three different axes and displays them on the Rows, Columns, and Fields area. Because the query used by the OWC control retrieves data on three - dimensional axes, you cannot execute the same query in SQL Server Management Studio (SSMS). SSMS will only be able to display two - dimensional results. Therefore if you need to see the exact same results in SSMS, you need an MDX query that will retrieve results in a two - dimensional format. The MDX query generated by OWC can be re - written using the CrossJoin function or the cross join operator (*) so that the results can be retrieved on two axes. The simplified MDX query that will return the same results as the OWC is:

```
SELECT
{
[Measures].[Sales Amount - Fact Reseller Sales]
}
ON COLUMNS,
NON EMPTY {[Dim Reseller].[Business Type].members *
[Dim Geography].[English Country Region Name].members}
DIMENSION PROPERTIES MEMBER_NAME ON ROWS
FROM [Adventure Works DW]
CELL PROPERTIES VALUE, FORMATTED_VALUE, FORE_COLOR, BACK_COLOR
```

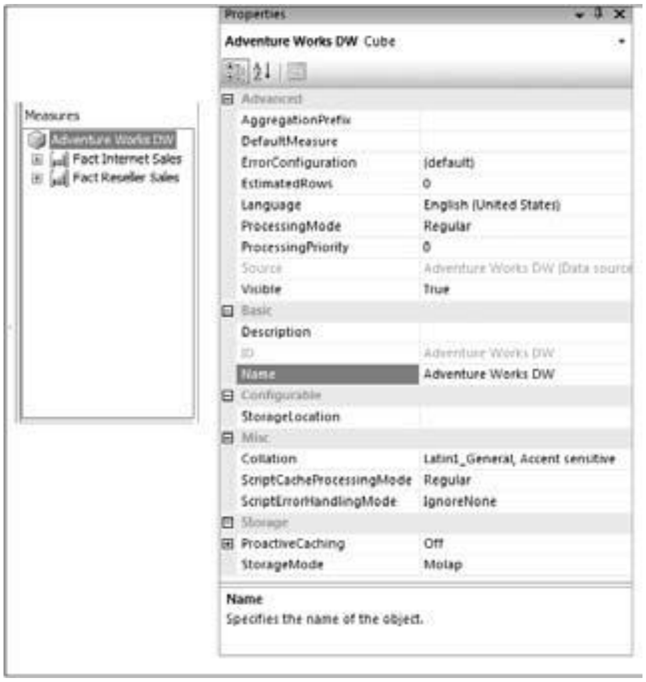
So far you have learned about cube dimensions, how to add them to a cube, how to define relationships, and then how to query data along with the dimensions. Cube dimensions and their attributes and hierarchies contain several properties. Some properties such as AttributeHierarchyEnabled, AttributeHierarchyVisible, and the AttributeHierarchyOptimizedState reflect the state of the cube dimension hierarchies or attributes in the shared dimension by default. You can override these properties so that appropriate settings are applied for the cube dimensions within the cube. The properties AggregationUsage for attributes and AllMemberAggregationUsage for cube dimensions control the behavior of aggregations designed on the cube. You learn more about these properties in Chapters 9 and 14 .

Measures and Measure Groups

You learned about editing cube dimensions and establishing the right relationships between dimensions and measure groups in a cube. Similarly, you can add or delete measures and measure groups in a cube. Measures are the focus point for data analysis and therefore they are the core objects of a cube. Measures are columns from the fact table that contain meaningful information for data analysis. Usually measures are of type numeric and can be aggregated or summarized along hierarchies of a dimension.

You can specify the type of aggregation to be applied for each measure. The most widely used aggregate functions are Sum, Count, and Distinct Count. A collection of measures forms an object called a measure group, and a collection of measure groups forms the dimension called *Measures* in the cube. “ Measures ” is a keyword in Analysis Services that refers to a special dimension that only contains the fact data.

If you click the Cube Structure tab in the cube editor you will see the Measures pane on the top - left corner. Click the cube named Adventure Works DW within the Measures pane to see the associated properties in the Properties window located on the bottom - right corner of the BIDS. Figure 6 - 17 shows the Measures and Properties panes. The Measures pane shows the cube name and the measure groups within the cube. You can see the two measure groups Fact Reseller Sales and Fact Internet Sales that correspond to the two fact tables. Fact table columns that become measures of the cube are contained within the corresponding measure group. There is typically a one - to - one relationship between a fact table and measure group in the cube.



Advanced	
AggregationPrefix	
DefaultMeasure	
ErrorConfiguration	(default)
EstimatedRows	0
Language	English (United States)
ProcessingMode	Regular
ProcessingPriority	0
Source	Adventure Works DW (Data source
Visible	True
Basic	
Description	
ID	Adventure Works DW
Name	Adventure Works DW
Configurable	
StorageLocation	
Misc	
Collation	Latin1_General, Accent sensitive
ScriptCacheProcessingMode	Regular
ScriptErrorHandlingMode	IgnoreNone
Storage	
ProactiveCaching	Off
StorageMode	Molap

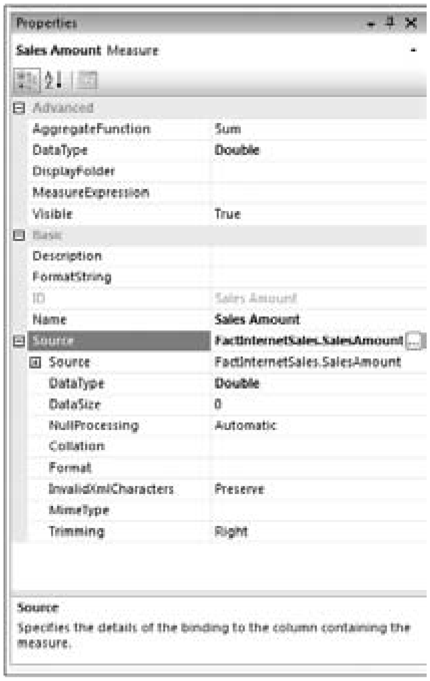
Name
Specifies the name of the object.

dimension, it needs to be handled differently when designing the cube. For example, if you have Fact Internet Sales data stored in separate fact tables for each quarter (fact data has been partitioned into multiple fact tables across the Time dimension), then with respect to the cube all these are considered a single fact table because they have the same schema. You typically partition your relational fact data into multiple fact tables due to design or scalability considerations, but when you want to analyze the data you will be looking to aggregate the data appropriately across various dimensions, especially the Time dimension. You can either merge the data from all the fact tables within the DSV with a named query or you can utilize the partitioning feature in Analysis Services so that Analysis Services aggregates the data correctly during browsing. You learn more about partitions in Chapters 7 and 14 .

You can see several properties of the cube in Figure 6 - 17 . The most important property is DefaultMeasure. As the name indicates, this property is used to define the measure used by default whenever queries are sent to the cube. The reason why the default measure is important is that whenever your MDX query does not explicitly contain a member from the measures dimension, the default measure is returned. In addition to that, the default measure is used whenever restrictions are applied in the query with the WHERE clause, and based on the default measure your results can be different. If you select the DefaultMeasure property you can see a drop - down list box that shows all the measures of the cube. You can choose the measure you want to define as the default measure of the cube. If the default measure is not specified, the first measure of the first measure group of the cube (as seen in the Measures pane) will be the default measure of the cube.

The next most important property is the StorageMode property. This defines whether your fact data will be stored in Analysis Services, your relational data source, or both. The StorageMode property has three options: Multidimensional OLAP (MOLAP), Relational OLAP (ROLAP), and Hybrid OLAP (HOLAP). The default value is MOLAP, which means that when the cube is processed, Analysis Services reads the relational data and stores it in a proprietary format for fast retrieval. You learn more about the defining storage modes in Chapter 9 . In Analysis Services 2008, you have the option to instruct the Analysis

relational data. The ProactiveCaching property lets you specify the frequency of the update of the cube data based on changes in the relational data. You learn about the benefits of proactive caching with the help of a complete scenario in Chapter 21 . The ErrorConfiguration property helps in handling the various errors that can occur while processing the fact data and defining what actions should be taken under such error circumstances such as ignoring the error, converting to a specific value, or stopping processing when errors are encountered. One of the main features of an OLAP database is the ability to create aggregations that facilitate fast query response times. The AggregationPrefix property is used to prefix the name of the aggregations that are created for the cube. The remaining properties are self - explanatory and you can find detailed information for each property in Analysis Services 2008 product documentation.
If you click one of the measure groups, you will see the properties associated with that measure group. Most of the properties at the cube level are also applicable to the measure group. If you specify a value for a property at the cube level as well as the measure group level for a common property, the value specified at the measure group level will be honored by the Analysis Services instance. Expand the measure group Fact Internet Sales and select the measure Sales Amount. The Properties pane now shows the properties of the measure, as shown in Figure 6 - 18 . Next, you learn the important properties of a measure in detail.
The AggregateFunction property defines how the measure value is to be aggregated from one level to another level of a hierarchy in a dimension. For example, assume the Product dimension contains a hierarchy called Products that contains two levels, Model Name and Product Name. Each model contains one or more products. If you want the sales amount of a specific product to be aggregated to the model, you need to specify the aggregate function to be Sum. Whenever you browse the cube along the Products hierarchy, you will see that the sales of each product are aggregated to the corresponding model. However, sometimes you might not want the measure value to be aggregated while browsing a hierarchy. Therefore, Analysis Services 2008 provides you with several aggregate functions. Aggregation functions supported in the properties can also be done using MDX scripts. (You learn about MDX scripts in Chapter 9 .) However, we recommended that you use the built - in aggregation functions supported in the Properties window to get optimal performance from your Analysis Services instance.



Sales Amount Measure

⊟ Advanced	
AggregateFunction	Sum
DataType	Double
DisplayFolder	
MeasureExpression	
Visible	True
⊟ Basic	
Description	
FormatString	
ID	Sales Amount
Name	Sales Amount
⊟ Source	FactInternetSales.SalesAmount
⊞ Source	FactInternetSales.SalesAmount
DataType	Double
DataSize	0
NullProcessing	Automatic
Collation	
Format	
InvalidXmlCharacters	Preserve
MimeType	
Trimming	Right

Source
Specifies the details of the binding to the column containing the measure.

Other than the Sum aggregate function, the most commonly used aggregate functions are Count and Distinct Count. The Count aggregate function, as the name indicates, is used whenever you want to count each occurrence of the measure value rather than add the measure values. For example, if you want to find the number of transactions in a day or number of customers in a day, you would use a Count aggregate function on a fact table column that indicates the customers who came to the store on a specific day. The Distinct Count aggregate function, on the other hand, can be used to identify the unique number of occurrences of a specific measure. For example, a customer can buy a specific product every month. If you want to find the unique number of customers who purchase a specific product, you use the Distinct Count aggregate function. You will see examples of Count and Distinct Count aggregate functions in this section. The None aggregate function is used when you do not want to aggregate the values of a specific measure across a dimension. An example of where the None aggregate function would be used is for the price of a specific product or discount provided on a unit product.

When you build and browse a cube you will see all the measures occurring under the dimension called [Measures] in the Measure Groups pane of the Cube Browser. If you want to organize the related measures in a logical structure that is more meaningful for users, you use the property called DisplayFolder. You can specify a measure to be part of one or more display folders by editing the DisplayFolder property. If you enter a name in the DisplayFolder property, that specific measure will become part of the newly entered display folder. You can make a specific measure part of multiple display folders by specifying the display folders separated by a semicolon. When display folders are specified, while browsing the cube you will see the display folders under the appropriate measure group name in the metadata pane of the Browser. Therefore you cannot have measures from different measure groups under a single display folder.

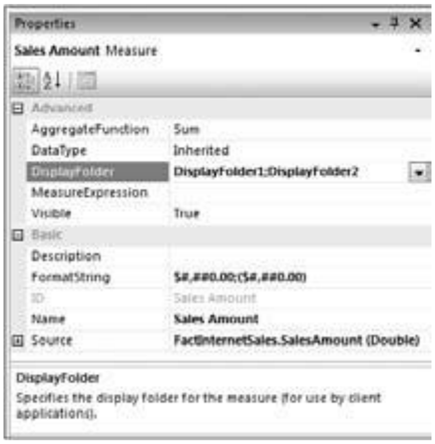
In some business applications you only allow access to the aggregated results of a measure. For such applications you need a way to aggregate the results of a measure but do not want to show the base measure. You can aggregate the results of a measure by specifying a measure called the calculated measure (you learn more about calculations a little later in this chapter) and hide the base measure.

The measure property Visible allows you to hide the base measure from viewing for such applications.

The FormatString property allows you to show the measure value in a format of your choice.

If you select the FormatString property you will see the various format options available. The MeasureExpression property is used for specifying the expression that evaluates the value for the measure. For example, if you have Sales information you might want to ensure the Sales information is presented in the local currency based on the currency conversion rates. In such a case you will define an appropriate expression for the MeasureExpression property for the measure Sales. You learn about the MeasureExpression property in Chapter 9 .

The easiest way to see the effect of some of the properties mentioned is to try them yourself in your own project. Specify two display folders named DisplayFolder1 and DisplayFolder2 for the measure Sales Amount. Because the measure Sales Amount is a currency data type, you can select the currency format. Select the $#,##0.00;($#,##0.00) format from the FormatString property drop - down list. The Properties window for the measure Sales Amount should resemble the Figure 6 - 19 .



⊟ Advanced	
AggregateFunction	Sum
DataType	Inherited
DisplayFolder	DisplayFolder1;DisplayFolder2
MeasureExpression	
Visible	True
⊟ Basic	
Description	
FormatString	$#,##0.00;($#,##0.00)
ID	Sales Amount
Name	Sales Amount
⊞ Source	FactInternetSales.SalesAmount (Double)

DisplayFolder
Specifies the display folder for the measure (for use by client applications).

You learned examples of where the aggregate functions Count and Distinct Count can be useful. In your Adventure Works DW cube, if you want to count the number of customers and distinct customers who have bought specific products, you need to use these aggregate functions. Customer Key identifies the customer who has bought a specific product in the fact table. Therefore, in order to see the counts of customers you need to create two new measures using the fact table column Customer Key. To create the two new measures follow these steps:

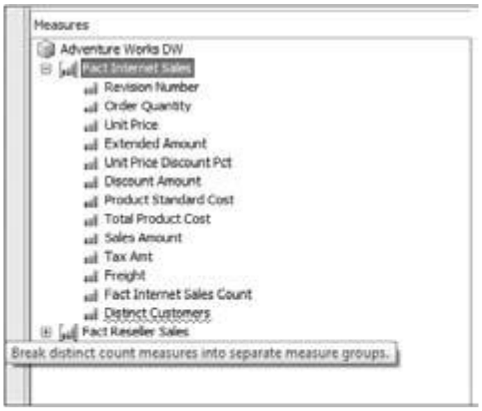
1. Right - click the measure group Fact Internet Sales and select New Measure.
2. In the New Measure dialog, select the checkbox “ Show all columns. ”
3. Select the column Customer Key and click OK. A measure called Customer Key is now created.
4. In the Measures pane, change the name for this measure from [Customer Key] to [Distinct Customers] by right - clicking the measure and selecting Rename.
5. Change the aggregate function for this measure to be Distinct Count.

At this point you will notice that a blue squiggly line shows up under the Distinct Customers measure. Analysis Services 2008 has built - in checks for many of the best practices for dimension and cube design. These best practices are implemented as warnings in the Analysis Management Objects (AMO) API. BIDS surfaces these AMO warnings through warning icons and blue squiggly lines for objects that don ' t meet these design best practices. To discover the rules that generate these warnings, you simply have to move the mouse over the blue squiggly lines or warning icon to see a tooltip with the best practice design rule. In this case the rule suggests that distinct count measures be broken out into separate measure groups (see Figure 6 - 20).

The reason for this warning is that distinct count measures are semi - additive and require storing records at a finer level of detail than measures utilizing other aggregate functions. For a distinct count of customers to be accessible, the individual customer IDs must be available. Therefore, Analysis Services must use the Customer Key attribute for any and all aggregations. Consider the following example. If you are interested in the total Sales (sum) for a product in a given year, it is not necessary to retain the individual customer IDs that purchased the product within the desired year. The Analysis Services engine should be able to pre - aggregate data at the Product ID and Year attribute levels as part of the normal processing of the cube. However, if there is a requirement that you know the distinct number of customers who purchased the product within the year, you must retain the customer keys throughout any aggregated data.

If the product in question was purchased by 50,000 customers over the year, the aggregate goes from one row of data per product per year to 50,000 rows per product for the year. Separating Distinct Count measures into different measure groups allows maximum pre - aggregation of other additive measures with minimal storage and on - the - fly aggregation requirements.

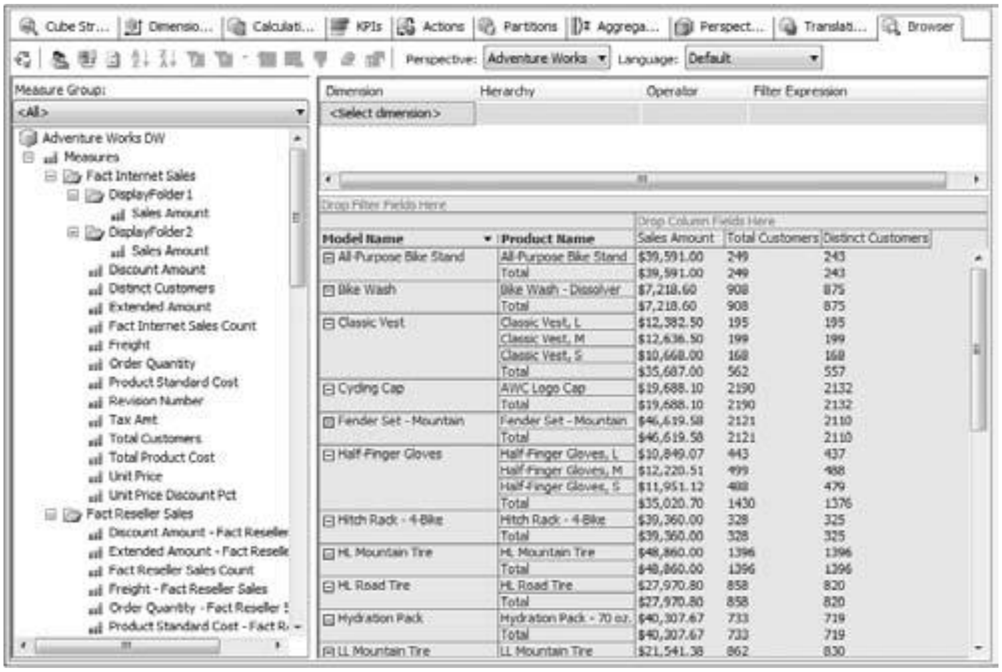
If you drag and drop the Customer Key column into the Measures pane to create a new measure, you will notice that the Cube Designer automatically creates a new measure group and adds the measure based on the assumption that you are probably defining a distinct count aggregation function. For this illustration we will ignore this best practice warning and move on to the next steps.



Break distinct count measures into separate measure groups.

1. Right - click the Fact Internet Sales measure group and select New Measure.
2. In the New Measure dialog select the checkbox " Show all columns. "
3. Select the column [Customer Key] and click OK. A measure called [Customer Key] is now created.
4. In the Measures pane, change the name for this measure from [Customer Key] to Total Customers by right - clicking the measure and selecting Rename.
5. In the Properties window change the AggregateFunction property for the [Total Customers] measure from Sum to Count.
6. The Unit Price of a product is the same value. Therefore this value should not be aggregated. In order to see the same value for a specific product you need to choose the aggregate function FirstNonEmpty.
7. Create a user hierarchy called Products in the Dim Product dimension with two levels, Model Name and English Product Name, where Model Name is the parent level of English Product Name. Rename the level English Product Name as Product Name.
8. Deploy the project to the Analysis Services instance.
9. Once the deployment is complete, switch to the Cube Browser tab. Click the Reconnect link to

get the updated metadata and data from the Analysis Services instance. When you expand the [Fact Internet Sales] folder under the Measures node, you will see two folders called DisplayFolder1 and DisplayFolder2 that contain the measure [Sales Amount] as shown in Figure 6 - 21 .



Model Name	Product Name	Sales Amount	Total Customers	Distinct Customers
All-Purpose Bike Stand	All-Purpose Bike Stand	$39,591.00	249	243
	Total	$39,591.00	249	243
Bike Wash	Bike Wash - Dissolver	$7,218.60	908	875
	Total	$7,218.60	908	875
Classic Vest	Classic Vest, L	$12,382.50	195	195
	Classic Vest, M	$12,636.50	199	199
	Classic Vest, S	$10,668.00	168	168
	Total	$35,687.00	562	557
Cycling Cap	AWC Logo Cap	$19,688.10	2190	2132
	Total	$19,688.10	2190	2132
Fender Set - Mountain	Fender Set - Mountain	$46,619.58	2121	2110
	Total	$46,619.58	2121	2110
Half-Finger Gloves	Half-Finger Gloves, L	$10,849.07	443	437
	Half-Finger Gloves, M	$12,220.51	499	488
	Half-Finger Gloves, S	$11,951.12	488	479
	Total	$35,020.70	1430	1376
Hitch Rack - 4-Bike	Hitch Rack - 4-Bike	$39,360.00	328	325
	Total	$39,360.00	328	325
HL Mountain Tire	HL Mountain Tire	$48,860.00	1396	1396
	Total	$48,860.00	1396	1396
HL Road Tire	HL Road Tire	$27,970.80	858	820
	Total	$27,970.80	858	820
Hydration Pack	Hydration Pack - 70 oz.	$40,307.67	733	719
	Total	$40,307.67	733	719
LL Mountain Tire	LL Mountain Tire	$21,541.38	862	830

10. Clear any data in your OWC browser by right - clicking within the OWC area and selecting Clear Results or by dragging and dropping each measure or hierarchy from rows/columns to outside of the OWC area.

11. Drag and drop the measures [Sales Amount], [Total Customers], and [Distinct Customers] for the [Fact Internet Sales] measure group to the data area of the OWC.

12. Then drag and drop the hierarchy Products from the Dim Product dimension to the rows and expand the member Classic Vest.

You can now see that the values for the measures are aggregated for the hierarchy Products that contains two levels, Model Name and Product Name, based on the aggregate function chosen. Choosing the aggregate functions Count and Distinct Count will not only count the values for the members of a hierarchy, but will also aggregate the counts to the next level. Notice that the values of [Sales Amount] are formatted based on the format string you specified earlier.

You should be able to add the measure [Unit Price] in the same browser and see the results of the FirstNonEmpty aggregate function. However we believe there is a bug in the product that causes the values for [Sales Amount] and [Total Customers] to be cleared in the Cube Browser. We recommend you clear the fields [Sales Amount] and [Total Customers] from the data area and drag and drop [Unit Price]. You can see that the Unit Price is aggregated from the members in the Product Name level to Model Name level based on the FirstNonEmpty aggregate function. You see the Total value for the measure Unit Price for the Classic Vest model as 63.5. The Unit Price value shown for the Mountain - 100 model name is the Unit Price Mountain - 100 Silver 42, one of the members of the Mountain - 100. In the example shown in Figure 6 - 21 , all the products under the model name Classic Vest have the same unit price. If you expand the Model Name member Mountain - 100 you will see that the Products under the model Mountain - 100 have different values.

You have now successfully enhanced the cube created by the Cube Wizard by adding cube dimensions and measures to the cube. In the process you have also learned about the properties of cube dimensions, measures, and measure groups. Most often businesses need complex logic to analyze the relational data. Analysis Services 2008 provides you with the ability to embed the complex calculations required for solving business problems in several ways. The most basic operation that every business will need is creating simple arithmetic operations on the base measures or dimension members. Objects created via such operations are called calculated members.