



# **Introduction to SQL Server 2005 Analysis Services**

Tutorial / Sample Courseware Chapter

provided by Accelebrate  
<http://www.accelebrate.com> • [info@accelebrate.com](mailto:info@accelebrate.com)  
877 849 1850 phone • 866 566 1228 fax

# Chapter 15: Analysis Services

## **In this chapter:**

- Understanding Analysis Services
- Introducing BIDS
- Creating a Data Cube
- Exploring a Data Cube

## **Files needed:**

- AdventureWorksCube1.zip
- AdventureWorksCube2.zip

## Analysis Services

So far in this course you've focused on getting data into a SQL Server database and then later getting the same data out of the database. You've seen how to create tables, insert data, and use SQL statements, views, and stored procedures to retrieve the data. This pattern of activity, where individual actions deal with small pieces of the database, is sometimes called *online transaction processing*, or OLTP.

But there's another use for databases, especially large databases. Suppose you run an online book store and have sales records for 50 million book sales. Maybe books on introductory biology show a strong spike in sales every September. That's a fact that you could use to your advantage in ordering stock, if only you knew about it.

Searching for patterns like this and summarizing them is called *online analytical processing*, or OLAP. Microsoft SQL Server 2005 includes a separate program called Microsoft SQL Server 2005 Analysis Services to perform OLAP analysis. In this chapter you'll learn the basics of setting up and using Analysis Services.

## Understanding Analysis Services

The basic idea of OLAP is fairly simple. Let's think about that book ordering data for a moment. Suppose you want to know how many people ordered a particular book during each month of the year. You could write a fairly simple query to get the information you want. The catch is that it might take a long time for SQL Server to churn through that many rows of data.

And what if the data was not all in a single SQL Server table, but scattered around in various databases throughout your organization? The customer info, for example, might be in an Oracle database, and supplier information in a legacy xBase database. SQL Server can handle distributed heterogeneous queries, but they're slower.

What if, after seeing the monthly numbers, you wanted to drill down to weekly or daily numbers? That would be even more time-consuming and require writing even more queries.

This is where OLAP comes in. The basic idea is to trade off increased storage space now for speed of querying later. OLAP does this by precalculating and storing aggregates. When you identify the data that you want to store in an OLAP database, Analysis Services analyzes it in advance and figures out those daily, weekly, and monthly numbers and stores them away (and stores many other aggregations at the same time). This takes up plenty of disk space, but it means that when you want to explore the data you can do so quickly.

## Understanding Analysis Services

Later in the chapter, you'll see how you can use Analysis Services to extract summary information from your data. First, though, you need to familiarize yourself with a new vocabulary. The basic concepts of OLAP include:

- Cube
- Dimension table
- Dimension
- Level
- Fact table
- Measure
- Schema

### Cube

The basic unit of storage and analysis in Analysis Services is the *cube*. A cube is a collection of data that's been aggregated to allow queries to return data quickly. For example, a cube of order data might be aggregated by time period and by title, making the cube fast when you ask questions concerning orders by week or orders by title.

Cubes are ordered into *dimensions* and *measures*. Dimensions come from *dimension tables*, while measures come from *fact tables*.

### Dimension table

A *dimension table* contains hierarchical data by which you'd like to summarize. Examples would be an Orders table, that you might group by year, month, week, and day of receipt, or a Books table that you might want to group by genre and title.

### Dimension

Each cube has one or more *dimensions*, each based on one or more dimension tables. A dimension represents a category for analyzing business data: time or category in the examples above. Typically, a dimension has a natural hierarchy so that lower results can be "rolled up" into higher results. For example, in a geographical level you might have city totals aggregated into state totals, or state totals into country totals.

### Level

Each type of summary that can be retrieved from a single dimension is called a *level*. For example, you can speak of a week level or a month level in a time dimension.

## Analysis Services

### Fact table

A *fact table* contains the basic information that you wish to summarize. This might be order detail information, payroll records, drug effectiveness information, or anything else that's amenable to summing and averaging. Any table that you've used with a Sum or Avg function in a totals query is a good bet to be a fact table.

### Measure

Every cube will contain one or more *measures*, each based on a column in a fact table that you'd like to analyze. In the cube of book order information, for example, the measures would be things such as unit sales and profit.

### Schema

Fact tables and dimension tables are related, which is hardly surprising, given that you use the dimension tables to group information from the fact table. The relations within a cube form a *schema*. There are two basic OLAP schemas: star and snowflake. In a *star schema*, every dimension table is related directly to the fact table. In a *snowflake schema*, some dimension tables are related indirectly to the fact table. For example, if your cube includes OrderDetails as a fact table, with Customers and Orders as dimension tables, and Customers is related to Orders, which in turn is related to OrderDetails, then you're dealing with a snowflake schema.



There are additional schema types besides the star and snowflake schemas, including parent-child schemas and data-mining schemas. However, the star and snowflake schemas are the most common types in normal cubes.

## Introducing Business Intelligence Development Studio

Business Intelligence Development Studio (BIDS) is a new tool in SQL Server 2005 that you can use for analyzing SQL Server data in various ways. You can build three different types of solutions with BIDS:

- Analysis Services projects

## Creating a Data Cube

- Integration Services projects (you'll learn about SQL Server Integration Services in Chapter 16)
- Reporting Services projects (you'll learn about SQL Server Reporting Services in Chapter 18)

To launch Business Intelligence Development Studio, select Microsoft SQL Server 2005 ► SQL Server Business Intelligence Development Studio from the Programs menu. BIDS shares the Visual Studio shell, so if you have Visual Studio installed on your computer, this menu item will launch Visual Studio complete with all of the Visual Studio project types (such as Visual Basic and C# projects).

## Creating a Data Cube

To build a new data cube using BIDS, you need to perform these steps:

- Create a new Analysis Services project
- Define a data source
- Define a data source view
- Invoke the Cube Wizard

We'll look at each of these steps in turn.



You'll need to have the AdventureWorksDW sample database installed to complete the examples in this chapter. This database is one of the samples that's available with SQL Server.

## Creating a New Analysis Services Project

To create a new Analysis Services project, you use the New Project dialog box in BIDS. This is very similar to creating any other type of new project in Visual Studio.

## Try It!

To create a new Analysis Services project, follow these steps:

1. Select Microsoft SQL Server 2005 ► SQL Server Business Intelligence Development Studio from the Programs menu to launch Business Intelligence Development Studio.
2. Select File ► New ► Project.

## Analysis Services

3. In the New Project dialog box, select the Business Intelligence Projects project type.
4. Select the Analysis Services Project template.
5. Name the new project AdventureWorksCube1 and select a convenient location to save it.
6. Click OK to create the new project.

Figure 15-1 shows the Solution Explorer window of the new project, ready to be populated with objects.

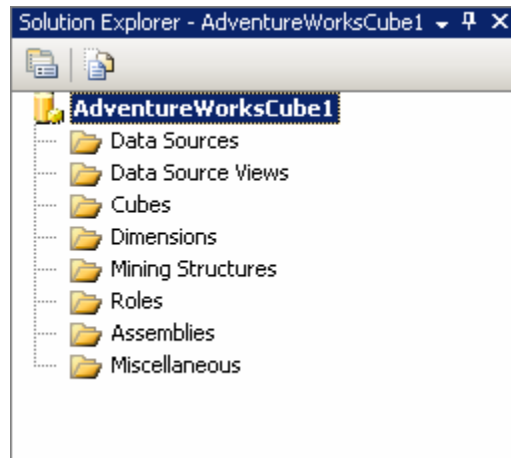


Figure 15-1: New Analysis Services project

## Defining a Data Source

To define a data source, you'll use the Data Source Wizard. You can launch this wizard by right-clicking on the Data Sources folder in your new Analysis Services project. The wizard will walk you through the process of defining a data source for your cube, including choosing a connection and specifying security credentials to be used to connect to the data source.

## Try It!

To define a data source for the new cube, follow these steps:

1. Right-click on the Data Sources folder in Solution Explorer and select New Data Source.
2. Read the first page of the Data Source Wizard and click Next.
3. You can base a data source on a new or an existing connection. Because you don't have any existing connections, click New.

## Try It!

4. In the Connection Manager dialog box, select the server containing your analysis services sample database from the Server Name combo box.
5. Fill in your authentication information.
6. Select the Native OLE DB\SQL Native Client provider (this is the default provider).
7. Select the AdventureWorksDW database. Figure 15-2 shows the filled-in Connection Manager dialog box.

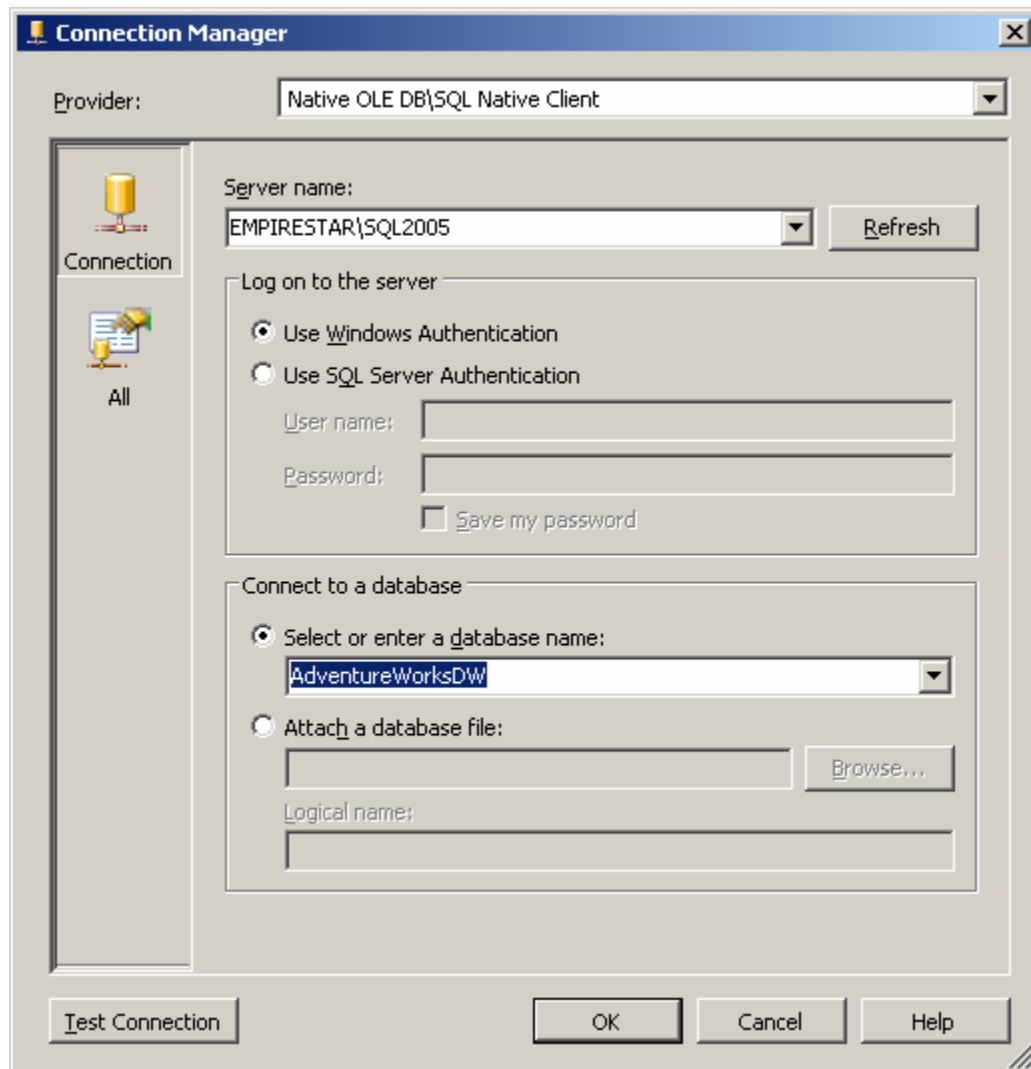


Figure 15-2: Setting up a connection

8. Click OK to dismiss the Connection Manager dialog box.
9. Click Next.
10. Select Default impersonation information to use the credentials you just supplied for the connection and click Next.
11. Accept the default data source name and click Finish.



## Analysis Services

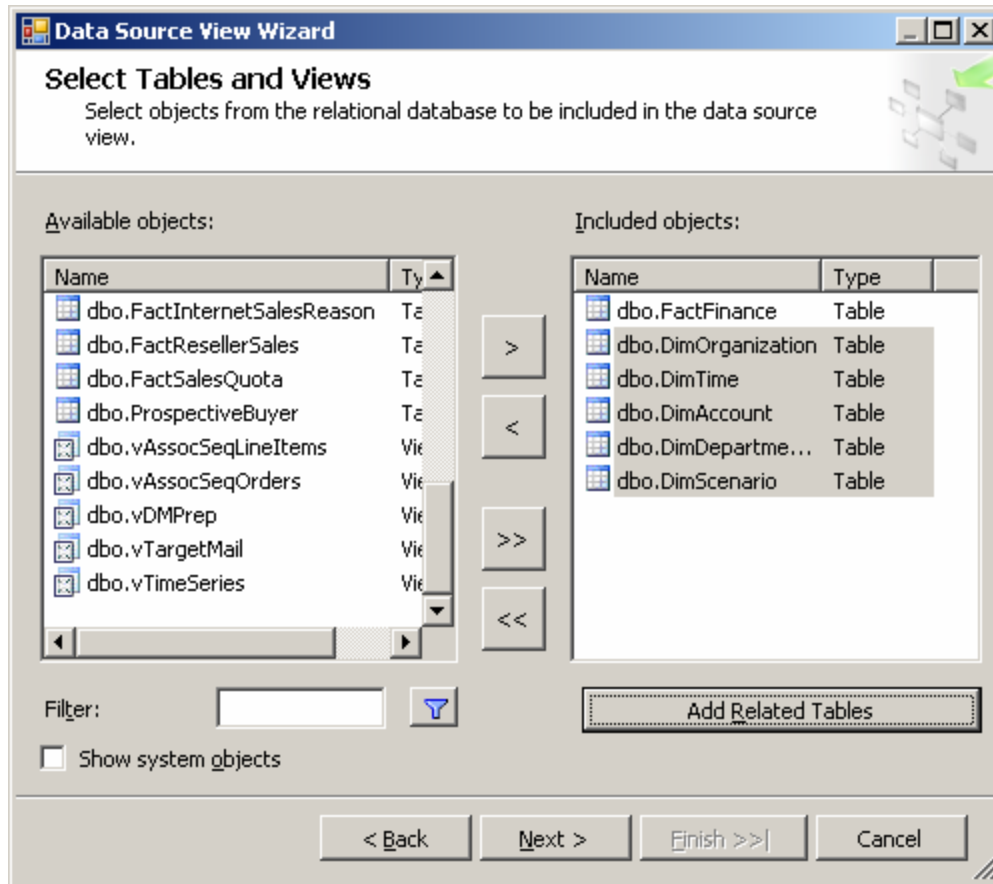
### Defining a Data Source View

A data source view is a persistent set of tables from a data source that supply the data for a particular cube. BIDS also includes a wizard for creating data source views, which you can invoke by right-clicking on the Data Source Views folder in Solution Explorer.

### Try It!

To create a new data source view, follow these steps:

1. Right-click on the Data Source Views folder in Solution Explorer and select New Data Source View.
2. Read the first page of the Data Source View Wizard and click Next.
3. Select the Adventure Works DW data source and click Next. Note that you could also launch the Data Source Wizard from here by clicking New Data Source.
4. Select the dbo.FactFinance table in the Available Objects list and click the > button to move it to the Included Object list. This will be the fact table in the new cube.
5. Click the Add Related Tables button to automatically add all of the tables that are directly related to the dbo.FactFinance table. These will be the dimension tables for the new cube. Figure 15-3 shows the wizard with all of the tables selected.



**Figure 15-3: Selecting tables for the data source view**

6. Click Next.
7. Name the new view Finance and click Finish. BIDS will automatically display the schema of the new data source view, as shown in Figure 15-4.

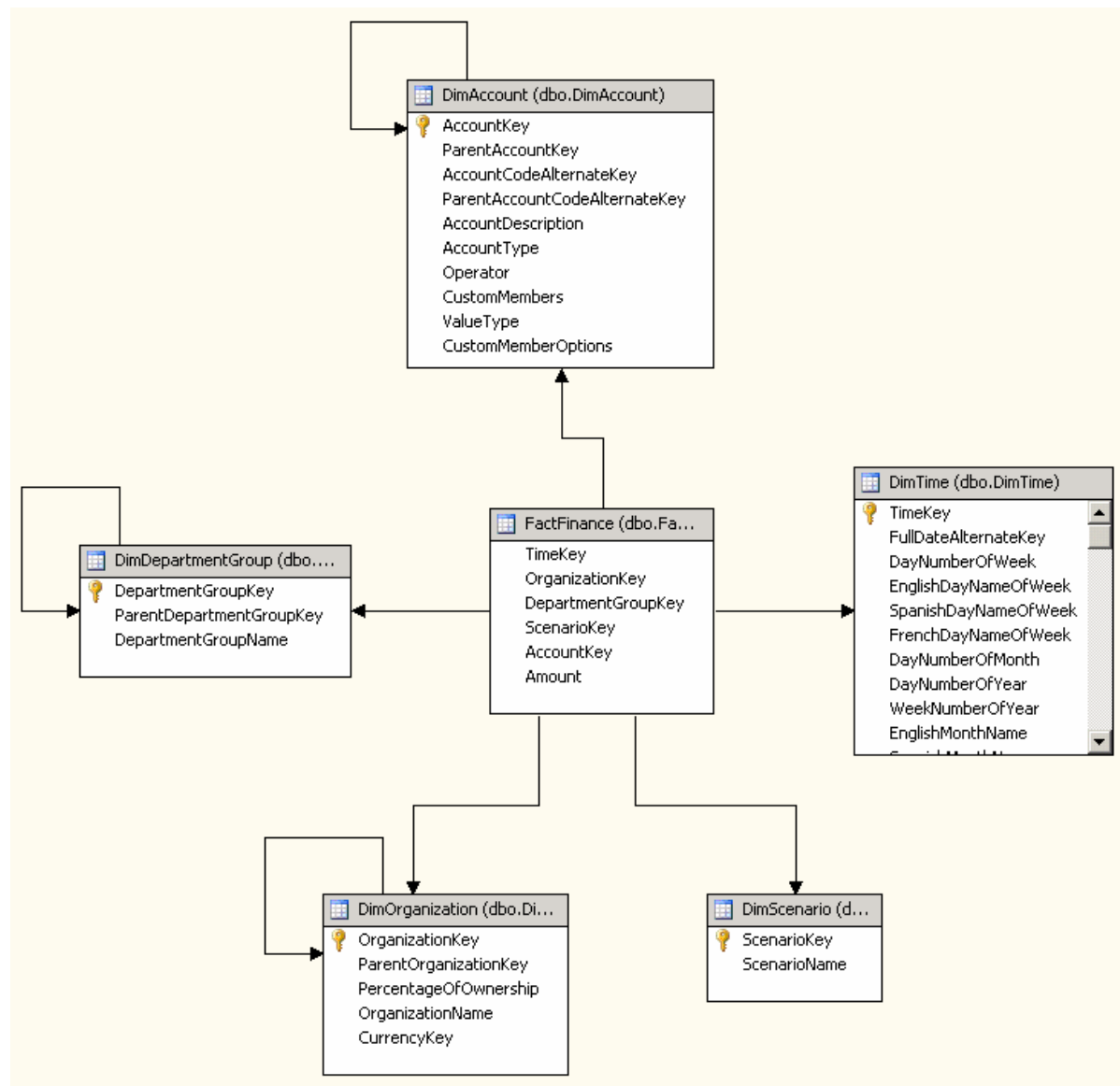


Figure 15-4: The Finance data source view

## Invoking the Cube Wizard

As you can probably guess at this point, you invoke the Cube Wizard by right-clicking on the Cubes folder in Solution Explorer. The Cube Wizard interactively explores the structure of your data source view to identify the dimensions, levels, and measures in your cube.

## Try It!

To create the new cube, follow these steps:

## Try It!

1. Right-click on the Cubes folder in Solution Explorer and select New Cube.
2. Read the first page of the Cube Wizard and click Next.
3. Select the option to build the cube using a data source.
4. Check the Auto Build checkbox.
5. Select the option to create attributes and hierarchies.
6. Click Next.
7. Select the Finance data source view and click Next.
8. Wait for the Cube Wizard to analyze the data and then click Next.
9. The Wizard will get most of the analysis right, but you can fine-tune it a bit. Select DimTime in the Time Dimension combo box. Uncheck the Fact checkbox on the line for the dbo.DimTime table. This will allow you to analyze this dimension using standard time periods.
10. Click Next.
11. On the Select Time Periods page, use the combo boxes to match time property names to time columns according to Table 15-1.

Time Property Name	Time Column
Year	CalendarYear
Quarter	CalendarQuarter
Month	MonthNumberOfYear
Day of Week	DayNumberOfWeek
Day of Month	DayNumberOfMonth
Day of Year	DayNumberOfYear
Week of Year	WeekNumberOfYear
Fiscal Quarter	FiscalQuarter
Fiscal Year	FiscalYear

Table 15-1: Time columns for Finance cube

12. Click Next.
13. Accept the default measures and click Next.
14. Wait for the Cube Wizard to detect hierarchies and then click Next.
15. Accept the default dimension structure and click Next.
16. Name the new cube FinanceCube and click Finish.

## Deploying and Processing a Cube

At this point, you've defined the structure of the new cube - but there's still more work to be done. You still need to deploy this structure to an Analysis Services server and then process the cube to create the aggregates that make querying fast and easy.

To deploy the cube you just created, select **Build ► Deploy AdventureWorksCube1**. This will deploy the cube to your local Analysis Server, and also process the cube,

## Analysis Services

building the aggregates for you. BIDS will open the Deployment Progress window, as shown in Figure 15-5, to keep you informed during deployment and processing.

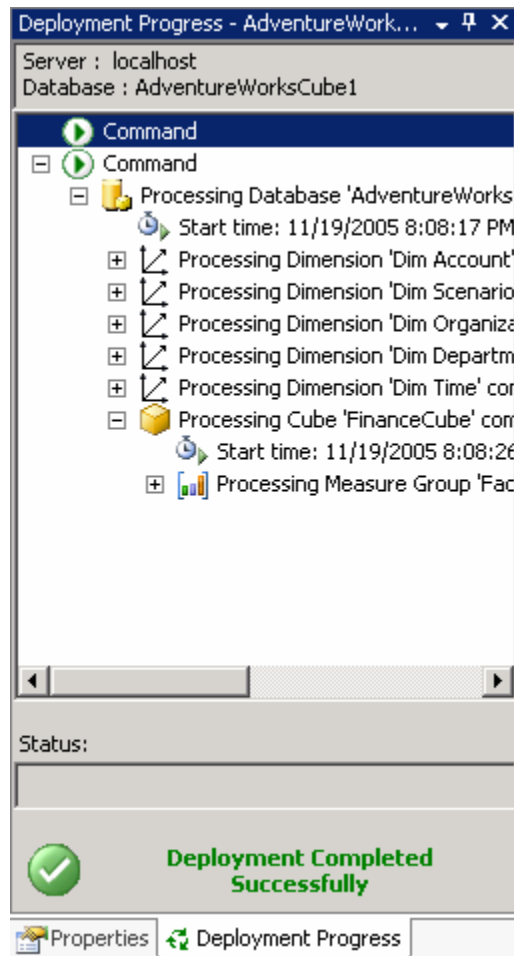


Figure 15-5: Deploying a cube



One of the tradeoffs of cubes is that SQL Server does not attempt to keep your OLAP cube data synchronized with the OLTP data that serves as its source. As you add, remove, and update rows in the underlying OLTP database, the cube will get out of date. To update the cube, you can select Cube ► Process in BIDS. You can also automate cube updates using SQL Server Integration Services, which you'll learn about in Chapter 16.

## Exploring a Data Cube

At last you're ready to see what all the work was for. BIDS includes a built-in Cube Browser that lets you interactively explore the data in any cube that has been deployed and processed. To open the Cube Browser, right-click on the cube in Solution Explorer and select Browse. Figure 15-6 shows the default state of the Cube Browser after it's just been opened.

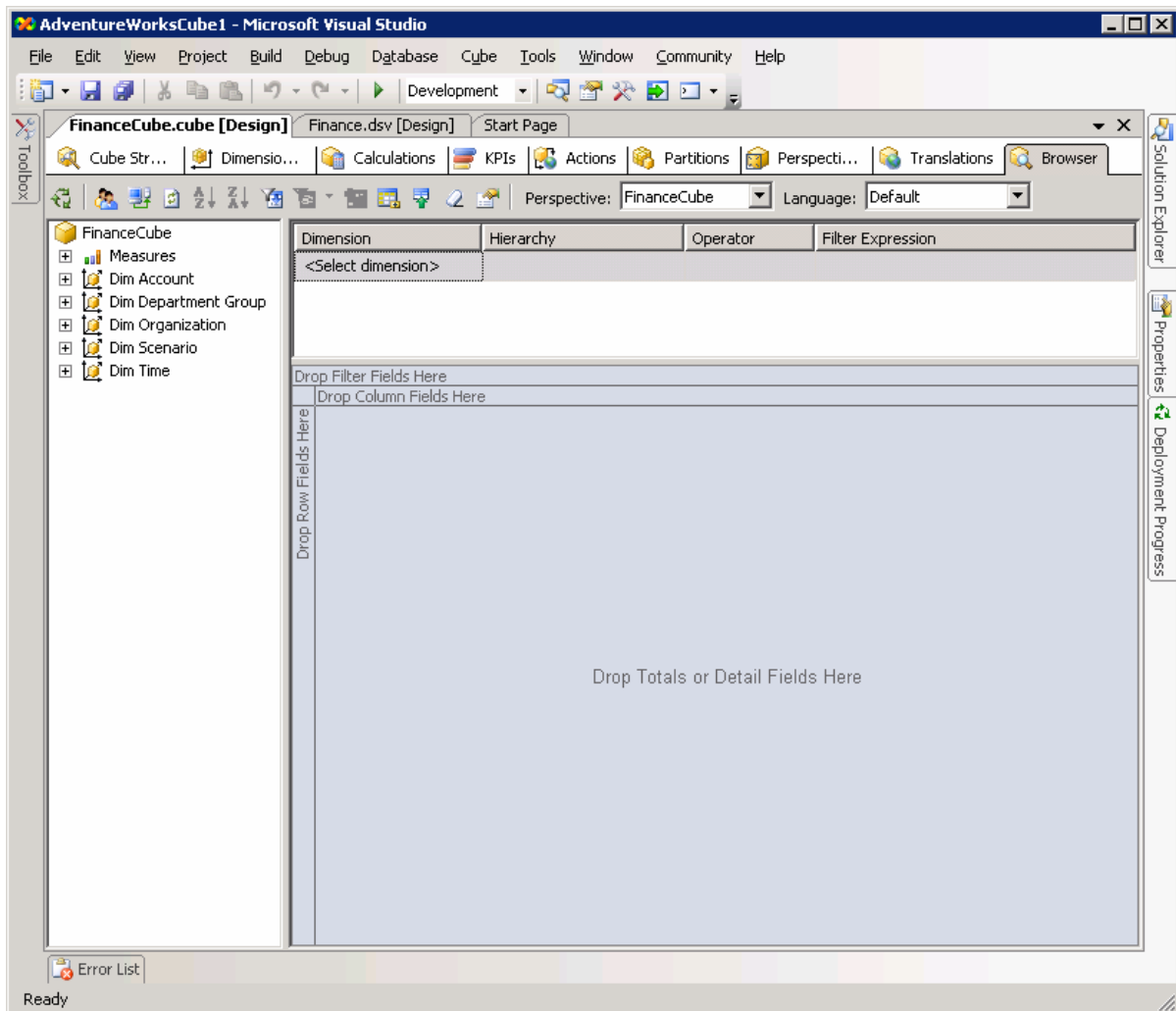
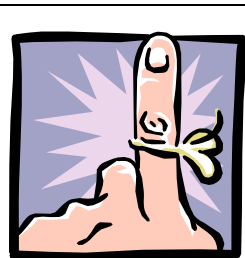


Figure 15-6: The cube browser in BIDS

The Cube Browser is a drag-and-drop environment. If you've worked with pivot tables in Microsoft Excel, you should have no trouble using the Cube browser. The pane to the left includes all of the measures and dimensions in your cube, and the pane to the right gives you drop targets for these measures and dimensions. Among other operations, you can:

## Analysis Services

- Drop a measure in the Totals/Detail area to see the aggregated data for that measure.
- Drop a dimension or level in the Row Fields area to summarize by that level or dimension on rows.
- Drop a dimension or level in the Column Fields area to summarize by that level or dimension on columns
- Drop a dimension or level in the Filter Fields area to enable filtering by members of that dimension or level.
- Use the controls at the top of the report area to select additional filtering expressions.



In fact, if you've worked with pivot tables in Excel, you'll find that the Cube Browser works exactly the same, because it uses the Microsoft Office PivotTable 11.0 control as its basis.

## Try It!

To see the data in the cube you just created, follow these steps:

1. Right-click on the cube in Solution Explorer and select Browse.
2. Expand the Measures node in the metadata panel (the area at the left of the user interface).
3. Expand the Fact Finance node.
4. Drag the Amount measure and drop it on the Totals/Detail area.
5. Expand the Dim Account node in the metadata panel.
6. Drag the Account Description property and drop it on the Row Fields area.
7. Expand the Dim Time node in the metadata panel.
8. Drag the Calendar Year-Calendar Quarter-Month Number of Year hierarchy and drop it on the Column Fields area.
9. Click the + sign next to year 2001 and then the + sign next to quarter 3.
10. Expand the Dim Scenario node in the metadata panel.
11. Drag the Scenario Name property and drop it on the Filter Fields area.
12. Click the dropdown arrow next to scenario name. Uncheck all of the checkboxes except for the one next to the Budget name.

Figure 15-7 shows the result. The Cube Browser displays month-by-month budgets by account for the third quarter of 2001. Although you could have written queries to

## Exploring a Data Cube

extract this information from the original source data, it's much easier to let Analysis Services do the heavy lifting for you.

The screenshot shows the Microsoft Visual Studio interface for AdventureWorksCube1. The main window displays a pivot table of financial data. The table is filtered by Scenario Name 'Budget' and CalendarYear '2001'. The columns represent different time periods (7, 8, 9, Total, 4, Total) and the rows represent various account descriptions. The data is presented in a grid format with numerical values.

Account Description	7	8	9	Total	4	Total	2002	Grand
Amortization of Goodwill	1760	1760	1760	5280	5310	10590	10350	20940
Building Leasehold	9750	9750	9750	29250	30120	59370	59970	119340
Commissions	45000	45000	45000	135000	185400	320400	331200	651600
Conferences	1140	1140	1140	3420	3360	6780	7200	13980
Discounts		2300	600	2900	4500	7400	519500	526900
Employee Benefits	28110	28110	28110	84330	78300	162630	184380	347010
Entertainment	2360	2360	2360	7080	6510	13590	13800	27390
Equipment	660	660	660	1980	1950	3930	3990	7920
Furniture and Fixtures	2190	2190	2190	6570	7230	13800	13440	27240
Intercompany Sales	20700	116400	54200	191300	349700	541000	490500	1031500
Marketing Collateral	2540	2540	2540	7620	10980	18600	19710	38310
Meals	2930	2930	2930	8790	8280	17070	17010	34080
Office Supplies	4620	4620	4620	13860	12870	26730	27180	53910
Other Assets	1950	1950	1950	5850	5850	11700	11580	23280
Other Expenses	2720	2720	2720	8160	8010	16170	16170	32340
Payroll Taxes	35380	35380	35380	106140	116250	222390	248340	470730
Professional Services	2980	2980	2980	8940	8100	17040	17220	34260
Rent	8490	8490	8490	25470	26220	51690	51660	103350
Returns and Adjustments	10750	52400	59200	122350	168900	291250	375000	666400
Salaries	380900	380900	380900	1142700	1207200	2349900	2643300	4992200
Standard Cost of Sales								

Figure 15-7: Exploring cube data in the cube browser



### Exercises

Create a data cube, based on the data in the AdventureWorksDW sample database, to answer the following question: what were the internet sales by country and product name for married customers only?

## Solutions to Exercises

To create the cube, follow these steps:

1. Select Microsoft SQL Server 2005 ► SQL Server Business Intelligence Development Studio from the Programs menu to launch Business Intelligence Development Studio.
2. Select File ► New ► Project.
3. In the New Project dialog box, select the Business Intelligence Projects project type.
4. Select the Analysis Services Project template.
5. Name the new project AdventureWorksCube2 and select a convenient location to save it.
6. Click OK to create the new project.
7. Right-click on the Data Sources folder in Solution Explorer and select New Data Source.
8. Read the first page of the Data Source Wizard and click Next.
9. Select the existing connection to the AdventureWorksDW database and click Next.
10. Select Default and click Next.
11. Accept the default data source name and click Finish.
12. Right-click on the Data Source Views folder in Solution Explorer and select New Data Source View.
13. Read the first page of the Data Source View Wizard and click Next.
14. Select the Adventure Works DW data source and click Next.
15. Select the dbo.FactInternetSales table in the Available Objects list and click the > button to move it to the Included Object list.
16. Click the Add Related Tables button to automatically add all of the tables that are directly related to the dbo.FactInternetSales table.
17. Click Next.
18. Name the new view InternetSales and click Finish.
19. Right-click on the Cubes folder in Solution Explorer and select New Cube.
20. Read the first page of the Cube Wizard and click Next.
21. Select the option to build the cube using a data source.
22. Check the Auto Build checkbox.
23. Select the option to create attributes and hierarchies.
24. Click Next.
25. Select the InternetSales data source view and click Next.
26. Wait for the Cube Wizard to analyze the data and then click Next.
27. Click Next.
28. Accept the default measures and click Next.

## Analysis Services

29. Wait for the Cube Wizard to detect hierarchies and then click Next.
30. Accept the default dimension structure and click Next.
31. Name the new cube InternetSalesCube and click Finish.
32. Select Build ► Deploy AdventureWorksCube2.
33. Right-click on the cube in Solution Explorer and select Browse.
34. Expand the Measures node in the metadata panel.
35. Drag the Order Quantity and Sales Amount measures and drop it on the Totals/Detail area.
36. Expand the Dim Sales Territory node in the metadata panel.
37. Drag the Sales Territory Country property and drop it on the Row Fields area.
38. Expand the Dim Product node in the metadata panel.
39. Drag the English Product Name property and drop it on the Column Fields area.
40. Expand the Dim Customer node in the metadata panel.
41. Drag the Marital Status property and drop it on the Filter Fields area.
42. Click the dropdown arrow next to Marital Status. Uncheck the S checkbox.

Figure 15-8 shows the finished cube.

## Solutions to Exercises

The screenshot shows the AdventureWorksCube2 cube in Microsoft Visual Studio. The main window displays a data table with the following columns: Sales Territory Country, English Product Name, Order Quantity, Sales Amount, Order Quantity, Sales Amount, Order Quantity, and Sales Amount. The table is filtered by Marital Status = M. The left pane shows the cube's dimensions and measures.

Sales Territory Country	Order Quantity	Sales Amount	Order Quantity	Sales Amount	Order Quantity	Sales Amount
Australia	42	6678	223	2004.77	122	969.90
Canada	18	2862	144	1294.56	94	747.30
France	5	795	115	1033.85	17	135.15
Germany	5	795	131	1177.69	33	262.35
United Kingdom	12	1908	196	1762.04	47	373.65
United States	55	8745	377	3389.22999999997	210	1669.5
Grand Total	137	21783	1186	10662.14	523	4157.8

Figure 15-8: The AdventureWorksCube2 cube