

Irina Gorbach
Alexander Berger
Edward Melomed

Microsoft®
SQL Server™ 2008
Analysis Services

UNLEASHED



SAMS

Microsoft® SQL Server™ 2008 Analysis Services Unleashed

Copyright © 2009 by Pearson Education, Inc.

All rights reserved. No part of this book shall be reproduced, stored in a retrieval system, or transmitted by any means, electronic, mechanical, photocopying, recording, or otherwise, without written permission from the publisher. No patent liability is assumed with respect to the use of the information contained herein. Although every precaution has been taken in the preparation of this book, the publisher and author assume no responsibility for errors or omissions. Nor is any liability assumed for damages resulting from the use of the information contained herein.

ISBN-13: 978-0-672-33001-8

ISBN-10: 0-672-33001-6

Library of Congress Cataloging-in-Publication Data:

Melomed, Edward.

Microsoft SQL server 2008 analysis services unleashed / Edward

Melomed, Alexander Berger, Irina Gorbach.

p. cm.

ISBN 978-0-672-33001-8

1. SQL server. 2. Client/server computing. 3. Relational databases.

I. Berger, Alexander. II. Gorbach, Irina. III. Title.

QA76.9.C55M483 2008

005.75'65-dc22

2008049303

Printed in the United States of America

First Printing December 2008

Trademarks

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Sams Publishing cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

Warning and Disclaimer

Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied. The information provided is on an "as is" basis. The authors and the publisher shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book.

Bulk Sales

Pearson offers excellent discounts on this book when ordered in quantity for bulk purchases or special sales. For more information, please contact:

U.S. Corporate and Government Sales

1-800-382-3419

corpsales@pearsontechgroup.com

For sales outside of the U.S., please contact:

International Sales

+1-317-581-3793

international@pearsontechgroup.com

Editor-in-Chief

Karen Gettman

Executive Editor

Neil Rowe

Acquisitions Editor

Brook Farling

Development Editor

Mark Renfrow

Managing Editor

Patrick Kanouse

Project Editor

Jennifer Gallant

Copy Editor

Keith Cline

Indexer

Ken Johnson

Proofreader

Water Crest
Publishing, Inc.

Publishing

Coordinator

Cindy Teeters

Book Designer

Gary Adair

Composer

Mark Shirar

Graphics

Laura Robbins

Foreword

I am pleased to see this book being updated for a second edition, to cover the changes in Analysis Services 2008, and also to clarify some of the more difficult material in the first edition. This should make the book even more useful to its target users.

Now that Analysis Services is in its fourth major release, it has become a big, complex product, far removed from the relatively simple first release of a mere decade earlier. To make the most of it, model designers need much more knowledge than is available in the online documentation, which makes books like this all the more necessary. And, of course, now that the product is so widely used, sometimes for quite challenging applications, there is much more experience of the best practices to follow. Some of these are now baked into the product itself, but books like this can provide much more context for their use.

The authors are to be highly commended for putting in the effort to comprehensively update a substantial work like this; I know from my own experience how much extra motivation you need to update an existing publication after just two years, compared to the excitement of creating the first edition. All too often, publications like this remain frozen when new versions of the software they describe are released, leaving users to guess which parts remain true, and which have been superseded. In this case, this second edition actually follows more closely on the heels of Analysis Services 2008 than did the first edition on Analysis Services 2005.

Microsoft is also to be commended for continuing to permit or even encourage the disclosure of this level of detail about one of its major products; with the consolidation of the BI industry, some of the other major vendors have become much less willing to provide detailed information about the inner workings of their products. In any case, I have never known any other OLAP server vendor to be so open.

Users of Analysis Services are fortunate in the range of books available to them: more than for all the other OLAP servers combined. This is clearly the book for the most technical users who really need and want to understand exactly how Analysis Services works.

There are many other books for those just getting started with Analysis Services, or who want a clear 'how do I?' guide. The many application developers who just want to improve their Analysis Services skills will probably find this book overwhelming; there are at least a dozen simpler books to choose from. And, needless to say, this book is definitely not aimed at business users who want to understand what Analysis Services can do for them.

Nigel Pendse

Editor of *The OLAP Report*

Author of *The OLAP Survey*

Introduction

Analysis Services began as the project of a small Israeli firm named Panorama, which had responded to a request from a British publishing company to develop an application that would analyze the data stored in its relational database. By the end of 1994, Panorama developers began work on a more general application that would make it possible for business managers to analyze data with relative ease.

With its first release in 1995, Panorama deployed the application to several dozen customers. As the next release moved the application more deeply into the Israeli market, the Panorama team began to develop a new client/server analytical application. The server would process the data and store it in a proprietary format, and the client would also offer users an easy-to-use, rich graphical interface.

By 1996, the application had come to the attention of Microsoft, which acquired the technology by the end of that same year. In early 1997, a small Panorama team comprised of Alexander Berger, Amir and Ariel Netz, Edward Melomed, and Mosha Pasumansky moved from Tel Aviv to Redmond to start work on the first version of Microsoft OLAP Server. After the move to the United States, the team added new developers Irina Gorbach and Py Bateman.

To make the application attractive to enterprise customers, the team took on the challenge of formalizing and standardizing data exchange protocols, and they eliminated the client side of the application in favor of supporting a variety of third-party client applications. In early 1997, a small group including Alexander Berger retreated to a Puget Sound island to brainstorm the foundation of what would become SQL Server Analysis Services.

That retreat produced a plan for developing a standard protocol for client applications to access OLAP data: OLEDB for OLAP. More important, and more challenging, was the plan for developing a new query language that could access multidimensional data stored in the OLAP server—MDX (Multidimensional Expressions). MDX is a text language similar to SQL. MDX makes it possible to work with a multidimensional dataset returned from a multidimensional cube. From its inception, MDX has continued to change and improve, and now it is the de facto standard for the industry.

The original release plan was to include the OLAP server in the 1997 release of SQL Server 6.5. However, instead of rushing to market, Microsoft decided to give the development team more time to implement MDX and a new OLEDB for OLAP provider. Microsoft's first version of a multidimensional database was released in 1998 as part of SQL Server 7.0. That version was integrated with Microsoft Excel PivotTables, the first client for the new server.

Under the slogan, “multidimensionality for the masses,” this new multidimensional database from Microsoft opened the market for multidimensional applications to companies of all sizes. The new language and interface were greeted favorably. The simplicity (and, one could say, elegance) of the design made it possible for users to rapidly become proficient with the new product, including users who weren’t database experts. Technology that used to be available only to large corporations was now accessible to medium-sized and small businesses. As a result, the market for new applications that use multidimensional analysis has expanded and flourished in an environment rich with developers who write those applications.

But, of course, we were not satisfied to rest on our laurels. We took on a new goal—turn Analysis Services into a new platform for data warehousing. To achieve this, we introduced new types of dimensions, increased the volume of data the server can process, and extended the calculation model to be more robust and flexible. Even though no additional personnel joined the team for this effort, by the end of 1999 we brought the new and improved Analysis Services 2000 to market.

For the next five years, more and more companies adopted Analysis Services until it became a leader in the multidimensional database market, garnering a 27% market share. Now, multidimensional databases running on OLAP servers are integral to the IT infrastructures of companies of all sizes. In response to this wide adoption of multidimensional database technology, Microsoft has increased the size of the team devoted to OLAP technology in order to continue to develop the platform to meet the requirements of enterprise customers.

For the 2005 release of SQL Server Analysis Services, we started from ground up, rewriting the original (and now aging) code base. We built enterprise infrastructure into the core of the server.

SQL Server 2008 release continues to improve architecture and functionality of Analysis Services. While improving the performance of query execution, it also introduces query language extensions and new management capabilities.

Who Is This Book’s Intended Audience?

In this book, we bring you the tools you need to fully exploit Analysis Services and explain the architecture of the system. You’ll find all of the coverage of our previous book (just in case you were wondering if you needed to go back and read that one first), including the basic architecture established in Analysis Services 2005, as well as all the improvements introduced in Analysis Services 2008. *Analysis Services Unleashed* gives you a full understanding of multidimensional analysis and the MDX query language. It also exposes all the aspects of designing multidimensional applications and management of the system.

How This Book Is Organized

The book is divided into the following nine parts:

Parts I and II are devoted to a formalized description of the multidimensional model implemented in the new version of the OLAP server. We give you the vocabulary and concepts you'll need to work with this model.

In Part III, we present a detailed discussion of MDX and explanation of the way we use it to query multidimensional data. You'll need a practical grasp of the data model and MDX to take advantage of all the functionality of Analysis Services.

We devote the middle section of the book in Parts IV–VII to the practical aspects of loading and storing data in Analysis Services, as well as methods of optimizing data preparation and data access. In addition, we examine server architecture.

In the last section of the book, Parts VIII–IX, we discuss data access, the architecture of client components, and data protection. In addition, we examine the practical aspects of administering the server and monitoring its activities.

We wish you great success in your work with Analysis Services 2008, and we hope that our humbly offered book is of service to you.

Conventions Used in This Book

Commands, scripts, and anything related to code are presented in a special monospace computer typeface. Bold indicates key terms being defined, and italic is used to indicate variables or for emphasis. Great care has been taken to be consistent in letter case, naming, and structure, with the goal of making command and script examples more readable. In addition, you might find instances in which commands or scripts haven't been fully optimized. This lack of optimization is for your benefit, as it makes those code samples more intelligible and follows the practice of writing code for others to read.

Other standards used throughout this book are as follows:

CAUTION

Cautions alert you to actions that should be avoided.

NOTE

Notes give you additional background information about a topic being discussed.

CHAPTER 1

Introduction to OLAP and Its Role in Business Intelligence

In the past decade, Microsoft SQL Server Analysis Services established itself as one of the leaders in the Business Intelligences systems market. Analysis Services helps managers, employees, customers, and partners to make more informed business decisions by enabling them to analyze information accumulated during a company's day-to-day operations.

Success of Analysis Services and the entire Business Intelligence market was predefined by incredible growth of amounts of data accumulated as a result of everyday functioning of a large number of companies. Today it's hard to imagine a business or an organization that doesn't use an online transaction processing (OLTP) system. OLTP systems provide means to highly efficient execution of a large number of small transactions and reliable access to data stored in the result of the transactions.

The volume of the data stored and processed for one day by an OLTP system could be several gigabytes per day; after a period of time, the total volume of data can reach to the tens and even hundreds of terabytes. Such a large volume of data can be hard to store, but it is a valuable source of information for understanding the way the enterprise functions. This data can prove very helpful for making projections that lead to successful strategic decisions, and for improving everyday decision making.

It's easy to see why analysis of data has become so important to the management of modern enterprises. However, OLTP systems are not well suited to analyzing data. In the past decades, an entire new market has emerged for systems that can provide reliable and fast access for analyzing very large amounts of data: online analytical processing (OLAP).

IN THIS CHAPTER

- The Multidimensional Data Model

OLAP enables managers, executives, and analysts to gain insight into data using fast, interactive, and consistent interfaces to a wide variety of possible views of information. For example, with OLAP solution, you can request information about company sales in Europe over the year, then drill down to the sales of computers in September, calculate year-to-date sales or compare revenue figures with those for the same products sold in January, and then see a comparison of TV sets sales in Europe in the same time period.

Because OLAP systems are designed specifically for analysis, they typically don't need to both read and write data. All that is necessary for analysis is reading data. With this emphasis on reading only, OLAP systems enjoy a speed advantage over their OLTP cousins. However, a read-only approach to the database architecture is not the only distinction of the OLAP solution. The following rules distinguish OLAP systems from relational databases:

► **Multidimensional data structures**

OLAP solutions typically use multidimensional data structures that allow analysts and managers to analyze numeric values from different perspectives, such as time, customers, products, and others.

► **Consistently fast data access**

Architecture of the system allows constantly fast access to the data. To ensure fast, predictable query times, OLAP solutions typically pre-aggregate data.

► **Intuitive interface**

Skilled analysts and nontechnical users alike can manipulate and analyze data; they can generate reports without involving their organization's IT department.

► **Complex calculations**

With multiple dimensions come more complex, cross-dimensional calculations. You might need to calculate the subtotal of sales for the state Washington, for example, to be expressed as a percentage of the whole U.S. sales. Further, this result may be presented as part of a time-series analysis (for instance, current month versus last month, versus a year ago).

The Multidimensional Data Model

The design and development of the multidimensional database—especially Microsoft SQL Server Analysis Services, the system designed and developed by the authors of this book—was inspired by the success of relational databases. If you're already familiar with relational databases, you'll recognize some of the terminology and architecture. But, to understand Analysis Services, you must first understand multidimensional data models, how this model defines the data and processes it, and how the system interacts with other data storing systems, primarily with the relational data model.

The multidimensional data model for Analysis Services consists of three more specific models:

- ▶ The conceptual data model
- ▶ The application data model
- ▶ The physical data model

The Conceptual Data Model

The conceptual data model contains information about how the data is represented and the methods for defining that data. It defines data in terms of the tasks that the business wants to accomplish using the multidimensional database. To define conceptual data model, you use the user specifications for the structure and organization of the data, rules about accessing the data (that is, security rules), and calculation and transformation methods.

In a sense, the conceptual data model serves as a bridge between a business model and the multidimensional data model. The solutions architect is the primary user for the conceptual data model. We use Data Definition Language (DDL) and MDX (Multidimensional Extensions) script for the creation of the conceptual model. You can also use Business Intelligence Development Studio to develop the conceptual data model.

The Application Data Model

The application model defines the data in a format that can be used by the analytical applications that will present data to a user in a way that he can understand and use. The primary user for the application data model is the client application, which exposes the model to the user. The application model is built with the MDX language and XML for Analysis protocol. The chapters of Part 3, “Using MDX to Analyze Data,” contain detailed information about MDX and a few of most commonly used client applications. The chapters of Part 7, “Accessing Data in Analysis Services,” contain information about protocol used by Analysis Services to communicate with client applications.

The Physical Data Model

As in the arena of relational databases, the physical model defines how the data is stored in physical media:

- ▶ **Where it is stored**—What drive (or maybe on the network), what types of files the data is stored in, and so on
- ▶ **How it is stored**—Compressed or not, how it's indexed, and so on
- ▶ **How the data can be accessed**—Whether it can be cached, where it can be cached, how it is moved into memory, and so on

The database administrator is the primary user for the physical data model. We use XML-based commands for manipulation of data on the physical layer.

Figure 1.1 shows relationships between three parts of multidimensional model.

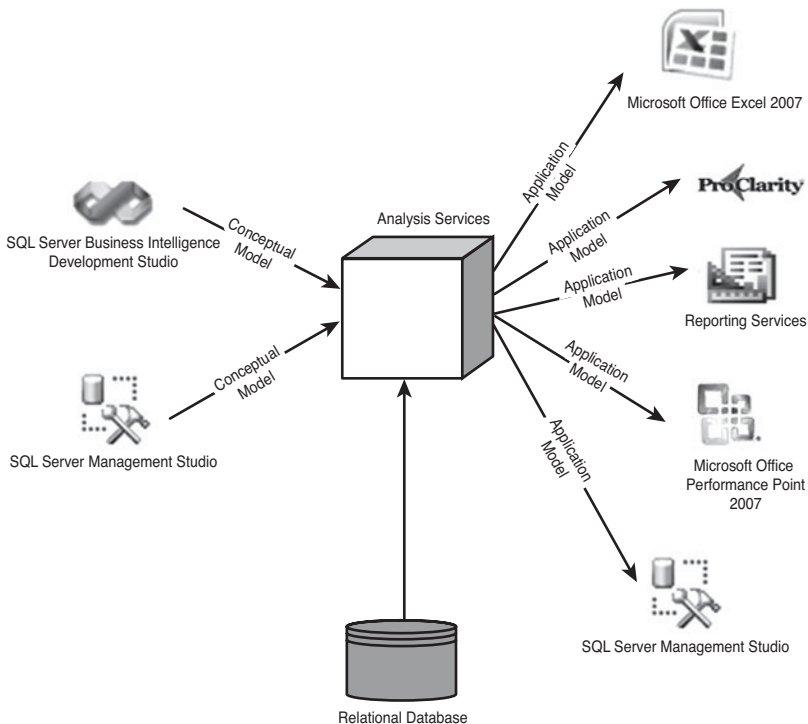


FIGURE 1.1 Submodels of the multidimensional model.

You use SQL Server Business Intelligence Development Studio or SQL Server Management Studio to define a conceptual data model, also known as a Unified Dimensional Model (UDM) or cube. After the conceptual model is defined, you populate it with data by loading/processing the data from the relational database. At this time, you define the physical data model—partitioning scheme of the data, indexing scheme, and so on. The application model of Analysis Services consists of standard data access interfaces. Client applications use those interfaces: XML for Analysis and MDX to communicate with Analysis Services. More than hundred applications available today support the application model of Analysis Services and can work with any Analysis Services cubes.

Unified Dimensional Model

The UDM of Microsoft SQL Server Analysis Services makes it possible for you to set up your system so that different types of client applications can access data from both the relational and the multidimensional databases in your data warehouse, without using separate models for each.

It's been a common industry practice for some time now to build data warehouses that include a relational database for storing data and a multidimensional database for analyzing data. This practice developed because the large volumes of data that multidimensional databases were developed to analyze are typically stored in relational databases. The data would be moved to the multidimensional database for analysis, but relational database would continue to serve as primary storage.

Therefore, it makes sense that the interaction between the stored data and the multidimensional database where it can be analyzed has been an important component of multidimensional database architecture. Our goal for Analysis Services, put simply, is speedy analysis of the most up-to-date data possible.

The speedy and up-to-date parts are what present the challenge. The data in OLTP systems is constantly being updated. But we wouldn't want to pour data directly from an OLTP system into a multidimensional database, because OLTP data is easily polluted by incomplete transactions or incomplete data entered in a transaction. In addition, you don't want your analysis engine to access the OLTP data directly, because that could disrupt work and reduce productivity.

In a data warehouse, OLTP data is typically transformed and stored in a relational database and then loaded into a multidimensional database for analysis. To connect the two databases, you can choose from three methods, each one using a different kind of interaction:

- ▶ Relational OLAP (ROLAP), in which no data is stored directly in the multidimensional database. It is loaded from the relational database when it is needed.
- ▶ Multidimensional OLAP (MOLAP), in which data is loaded into the multidimensional database and cached there. Future queries are run against the cached data.
- ▶ Hybrid OLAP (HOLAP), in which the aggregated data is cached in the multidimensional database. When the need arises for more detailed information, that data is loaded from the relational database.

In earlier versions of Analysis Services, the multidimensional part of the data warehouse was a passive consumer of data from the relational database. The functions of storing data and analyzing data were not only separate, but you had to understand two models—one for accessing a relational database and one for accessing a multidimensional database.

Some client applications would use one model, and others would use the other model. For example, reporting applications traditionally would access the data in a relational database. On the other hand, an analysis application that has to look at the data in many

different ways would probably access the data in the multidimensional database, which is designed specifically for that sort of use.

Now, the UDM offers a substantially redefined structure and architecture so that the one model (UDM) serves the purposes of any client application. You no longer have to understand two models; we're providing a unified model. Figure 1.2 shows how many different client applications can use UDM to access data in a variety of different data stores.

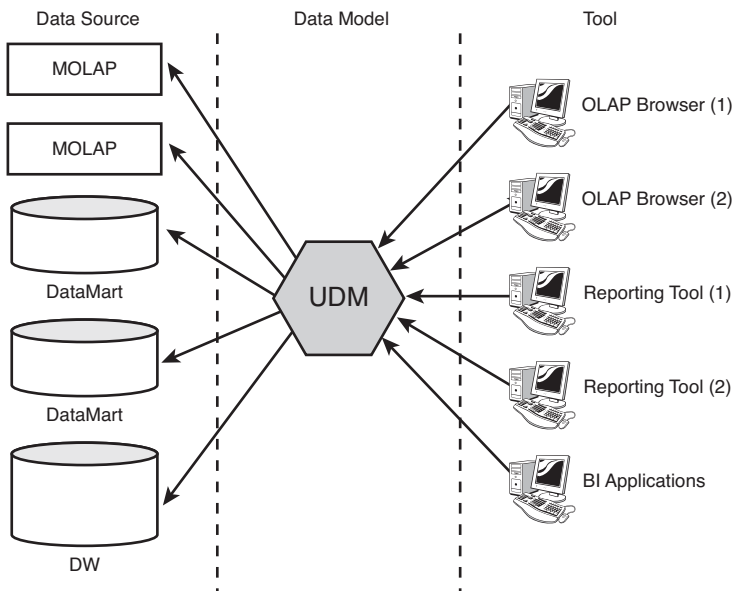


FIGURE 1.2 The UDM provides a unified model for accessing and loading data from varied data sources.

Analysis Services uses *proactive caching* to ensure that the user of the client application is always working with predictable data latency. In essence, proactive caching is a mechanism by which the user can schedule switching from one connection mode (ROLAP, MOLAP, or HOLAP) to another. For example, the user might set his system to switch from MOLAP to ROLAP if the data in the MOLAP system is older than, say, four hours.

With UDM at the center of the multidimensional model, you no longer need to have different methods of data access for different data sources. Before UDM, every system had a number of specialized data stores, each one containing data that was stored there for a limited number of users. Each of these data sources would likely require specific methods of data access for loading data into the multidimensional model. With Analysis Services, all the data of the enterprise is available through the UDM, even if those data sources are located on different types of hardware running different operating systems or different

database systems. OLAP now serves as an intermediate system to guarantee effective access to the data.

Basic Concepts

When you start to build a multidimensional model, you think about business entities your organization operates with and about values that you need to analyze. For example, in our fictional organization—a chain of grocery stores known as Food Mart—we operate with warehouses, stores, products, customers, and different currencies, as shown in Figure 1.3. Those business entities became *dimensions* of our multidimensional model. Typically, you want to analyze data in a context of a time periods, and therefore the Time dimension is present in almost all multidimensional models. Actual values or facts that you are analyzing, such as sales, costs, and units, are called *measures*.

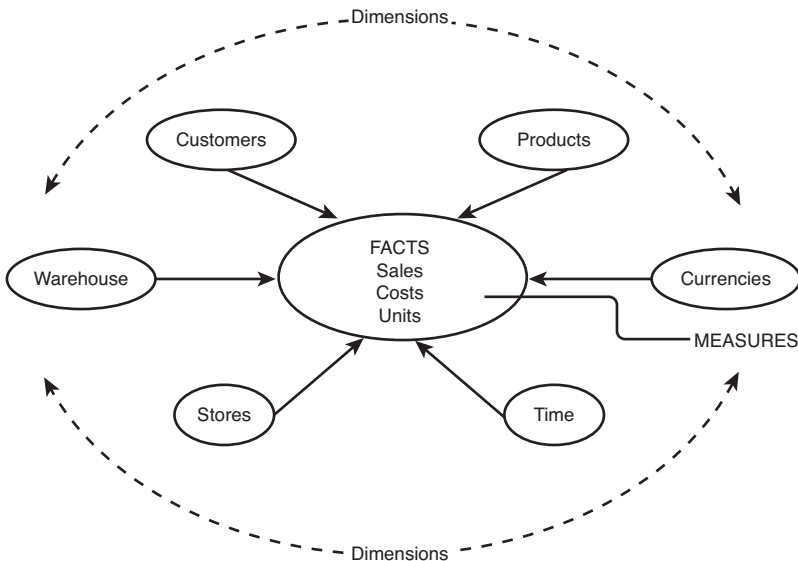


FIGURE 1.3 A multidimensional model consists of dimensions and measures.

Each individual element of the dimension is called a *member*. For example, “Club 1% Milk” is a member of the Products dimension, Irina Gorbach is a member of the Customers dimension, and January 1997 is a member of the Time dimension.

Each business entity usually has multiple characteristics. For instance, a customer can have the following properties: name, gender, city, state, and country. You might look at the products by name, Stock Keeping Unit (SKU), brand, product family, product category, and so on. We call these characteristics of the business entity *dimension attributes*. Figure 1.4 shows dimension attributes.

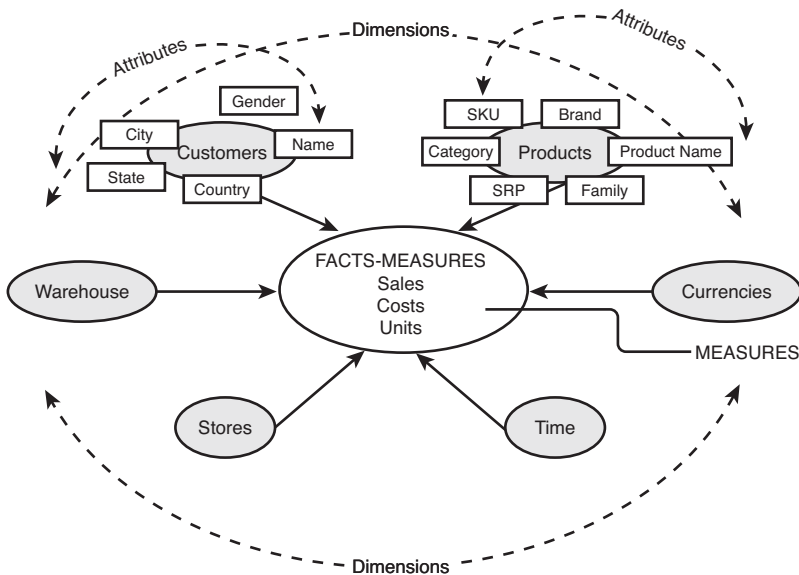


FIGURE 1.4 Each dimension is defined by its attributes.

Dimension attributes are not completely independent from each other. For example, Year contains Quarter, and Quarter contains Month. We can say that Year, Quarter, and Month attributes are related to each other.

If members of different attributes have a hierarchical structure, attributes can be organized in a *hierarchy*. For example, you can create the hierarchy Calendar—Year > Quarter > Month within the Time dimension, because the year contains quarters and quarters contains months.

After data is loaded in the cube, you can access it with many client applications. Microsoft Excel is one of the most frequently used application. Figure 1.5 shows Excel 2007 exposing data stored in Analysis Services cube.

This Excel spreadsheet demonstrates sales and cost for products in different time periods based on the data stored in the FoodMart 2008 database.

In Chapter 2, “Multidimensional Space,” we explain the terms that we use to describe multidimensional space.

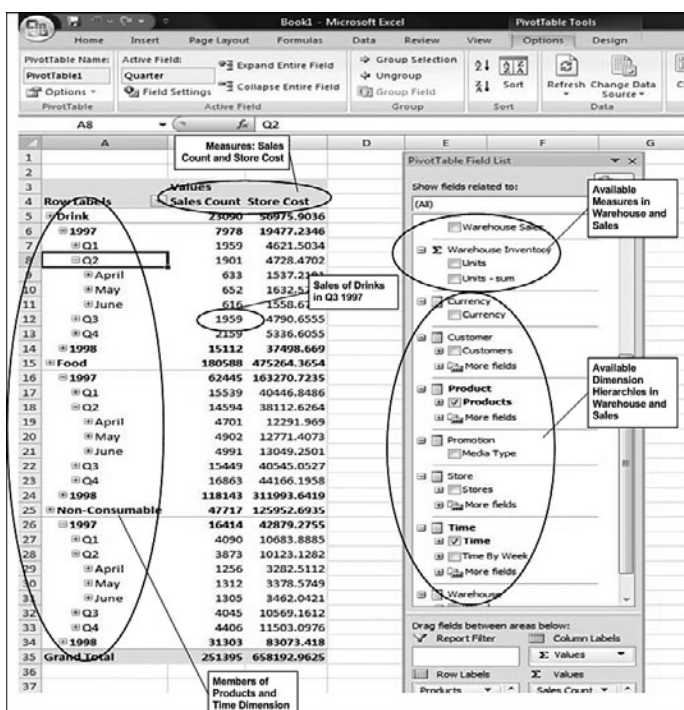


FIGURE 1.5 Accessing data in FoodMart 2008 sample using Excel 2007.

Index

A

Access property, **CellPermission** object, 752

accessing data

ADO.NET, 576

ADOMD.NET, 576

AMO, 577

Analysis Services Libraries, 575

Binary XML format, 570

client applications, building, 574

compressed data format, 571

DSO, 577

external data access security, 714

 changing service logon accounts, 720

 choosing service logon accounts, 718

 configuring data source access, 719-720

 failover cluster operations, 721

 named instances (SQL Server Browser),
 721

HTTP, 571-573

HTTPS, 571-573

offline, 573

OLE DB, 576

SOAP, 569-570

TCP/IP, 569-571

text format, 570

XML/A, 570, 574-575

Account dimension, 222-223

Account property, **ImpersonationInfo** object, 313

Account Time Intelligence Wizard, 231

account type mapping, 230

actions, 272

 creating, 276- 279

 defining, 273-276

 function of, 274-275

 scope of, 274

discovering, 279-283

report actions, 276

Actions collections, defining perspectives, 72

Actions parameter, cubes, 64

ACTION_TYPE property, 165

Add() method

named object collections, 674

unnamed object collections, 673

Add(string name) method, 675

Add(string name, string id) method, 675

administration

security, 713

traces, 765

Administrators role

defining, 722-723

server administrator security, 723

ADO.NET data access, 576

ADOMD.NET, 29

AdomdCommand class, 630

CommandStream property, 631

CommandText property, 631

Connection property, 630-631

Execute method, 632-636

ExecuteCellSet method, 632

ExecuteDataReader method, 632-633

ExecuteNonQuery method, 632, 635

ExecuteXmlReader method, 632-635

AdomdCommand.Cancel method, 658-662

AdomdDataParameter object, 655

AdomdParametersCollection object, 655

AdomdParametricQuery object, 655-657

AMO, sharing server sessions with, 686-688

analytical applications, writing with, 602

authentication, 606

CellSet object, 304, 636, 654

axis objects, 639

displaying multidimensional data in grid format, 640-641

object symmetry, 644-647

OlapInfo object, 637-639

retrieving Hierarchy object properties, 641-644

commands, asynchronous execution/
cancellation of, 658-662

connections

AdomdConnection class, 603

ConnectionString property, 604

Datasource property, 604

local cubes, 605

multi-instancing computers, 604

multiple connections using one session, 607-608

opening/closing, 603

passwords, 606

security, 606

server connections without specified databases, 608-610

session creation, 606

supported properties, 603

data access, 576

error handling, 662-663

AdomdCacheExpiredException class, 666-667

AdomdConnectionException class, 666

AdomdErrorResponseException class, 663-665

AdomdUnknownResponseException class, 666

MDX

queries, executing, 632

requests, parameters in, 655-657

metadata

caching, 615-617

collection operations, 612-615

Count method, 612

Find method, 612

GetEnumerator method, 613

GetSchemaDataSet method, 625-630

handling metadata not in object form, 625-630

Item method, 612

iteration of, 613-614

MemberCollection class, 618-624

Properties collection, 615

retrieving dimension ordinals, 615

schema rowset requests, 614-615

- multidimensional data in, 636-644
 - grid displays, 640-641
 - object symmetry, 644-647
- OlapInfo object, 637, 639
- project creation, 599-600
- tabular-formatted data
 - AdomdDataReader object, 649-654
 - DataAdapter class, 647
 - DataSet object, 647-649
 - IDataReader interface, 647
 - iterating query results, 650-652
 - populating datasets with query results, 647
 - Visual Studio operations with OLAP data, 652
- text data format, 571
- AdomdCacheExpiredException class, 666-667**
- AdomdCommand class, 630**
 - CommandStream property, 631
 - CommandText property, 631
 - Connection property, 630-631
 - Execute method, 632-636
 - ExecuteCellSet method, 632
 - ExecuteDataReader method, 632-633
 - ExecuteNonQuery method, 632, 635
 - ExecuteXmlReader method, 632-635
- AdomdCommand.Cancel method, 658-662**
- AdomdConnection class, 603**
- AdomdConnectionException class, 666**
- AdomdDataAdapter object, 647**
- AdomdDataParameter object, 655**
- AdomdDataReader object**
 - ADOMD.NET, 649-654
 - query results, iterating, 650-652
 - restrictions on, 652
 - uses for, 654
- AdomdErrorResponseException class, 663-665**
- AdomdParametersCollection object, 655**
- AdomdParametricQuery object, 655-657**
- ADOMDServer object model, 251-255**
- AdomdUnknownResponseException class, 666**
- AggregateFunction property, measures, 77-78**
- Aggregation Design Algorithm Wizard, 426**
- AggregationDesign object, 424-426**
- AggregationDesignID property, 364, 424-426**
- AggregationMemoryLimitMax server configuration property, 517**
- AggregationMemoryLimitMin server configuration property, 517**
- AggregationPrefix property, partitions, 364**
- aggregations, 419**
 - Account Time Intelligence Wizard, 231
 - Aggregation Design Algorithm Wizard, 426
 - attribute properties, 427
 - AverageOfChildren function, 229
 - building memory model of, 517-518
 - ByAccount function, 229-232
 - collections, 425
 - DDL, defining via, 424-425
 - defining, 417
 - design
 - algorithms, 427
 - objects, 423-426
 - dimension properties, 427
 - DISTINCT_COUNT function, 395, 400
 - FirstChild function, 228
 - FirstNonEmpty function, 228
 - flexible aggregations, 422-423
 - functions, 24
 - granularity of, 418
 - indexes, 373
 - LastChild function, 228
 - LastNonEmpty function, 228
 - managing, 431-432
 - manually designing, 431-432
 - monitoring usage of, 433-434
 - nonaggregatable attributes, 427
 - None function, 228
 - objects, 423-426
 - partitions
 - building in, 393-395
 - cube processing, 397
 - HOLAP data storage mode, 397
 - queries, 555, 559
 - relational reporting-style dimensions, 420-422

- rigid aggregations, 422-423
- ROLAP aggregations, 440
- AggregationStorage property, proactive caching, 439**
- AggregationUsage property, values of, 427**
- algebra, 142, 149**
 - CrossJoin sets, 151-152
 - Except sets, 150
 - Extract sets, 152
 - Intersect sets, 150
 - Union sets, 149
- algorithms, aggregation design, 427**
- All level (dimension hierarchies), 60**
- allocation methods (Analysis Service), 295**
- allocators (memory), 505, 511**
 - levels of, 512
 - types of, 511
- AllowCreate parameter, Alter command, 485**
- AllowedSet property, AttributePermission object, 735-737, 740, 747, 750**
- AllowOverwrite parameter, Create command, 484**
- Alter command, 245**
 - AllowCreate parameter, 485
 - ObjectExpansion parameter, 485
 - Scope parameter, 485
 - syntax of, 484
- AMO (Analysis Management Objects), 29, 239**
 - Clone method, 673
 - collections
 - major object collections, 677-678
 - named object collections, 674-676
 - unnamed object collections, 673-674
 - data access, 577
 - dependency in, 678-681
 - DependenciesCalculator class, 683-684
 - impact analysis, 681-682
 - disconnected mode, 693-694
 - error handling, 706
 - ConnectionException class, 708
 - OperationException class, 706-707
 - OutOfSyncException class, 708-709
 - ResponseFormatException class, 707

- hierarchy of objects, 670
- IClonable interface, 672
- major objects, 670
 - collections of, 677-678
 - IMajorObject interface, 676
 - IProcessable interface, 678
 - processable objects, 678
- minor objects, 672
 - ID property, 674
 - named objects, 674-676
 - unnamed objects, 672-674
- Parent method, 673
- referenced objects, 678
- Server.CancelCommand method, 688, 692
- text data format, 571
- uses of, 669
- Validate method, 673
- Visual Studio project
- Visual Studio projects, 685
 - asynchronous cube processing, 689-692
 - canceling long-running operations, 688-692
 - disconnected mode, 693-694
 - error handling, 706-709
 - object loading, 692-693
 - registration, 685
 - Scripter object, 694-696
 - server connections, 685-686
 - sharing ADOMD.NET sessions, 686-688
 - traces, 697-705

Analysis Services

- allocation methods, 295
- connecting, 111
- data source objects, 310
 - connection timeouts, 314
 - defining DDL, 310
 - pooling connections, 314
 - properties of, 310
 - security, 312-314
- distributed storage
 - linked objects, 32
 - remote partitions, 32
 - thick clients, 33
 - thin clients, 33

- DSV, 308
- MDX query execution context, 147-148
- Multidimensional Data Model, 19
 - aggregation functions, 24
 - application data model, 9
 - cells, 22
 - conceptual data model, 9, 37-41
 - dimensions, 13-22
 - four-tier architectures, 28-31
 - measures, 13
 - one-tier architectures, 27-29
 - physical data model, 9
 - slices, 19
 - subcubes, 24
 - three-tier architectures, 27, 30
 - tuples, 19
 - two-tier architectures, 27-28
 - UDM, 11-12, 32
- relational data, loading, 310
 - Binding object, 307-308
 - DataSource object, 307-311
 - DataSourceView object, 307
 - DSV, 308
- writeback
 - enabling, 301-303
 - IRowsetChange OLE DB interface, 304
 - permanent writeback, 291, 299-301
 - temporary writeback, 291-292, 299-301
 - UPDATE_CUBE statements, 292-301, 304
 - writeback partitions, 300-303
- Analysis Services Libraries, 575**
- analytical data providers, XML/A and, 29**
- Annotation property, 38**
 - named minor objects, 674
- anonymous access, connection security, 716**
- APIs (application programming interfaces), 237**
- Application property, 276**
- applications**
 - data models, 9
 - domains, sending CLR assemblies, 244
- applying**
 - hierarchies, 122
 - translations, 123
- arbitrary sets, 179**
- ArenaAllocator memory allocators, 511**
- assemblies**
 - CLR
 - creating, 239-242
 - sending, 244
 - COM, implementing, 245
 - deploying, 239
 - server object models, metadata, 252
- Assembly object, 239**
- assignments, 190, 198**
 - assignment operator
 - creating, 201
 - syntax of, 199-200
 - calculation
 - cells, 208-209
 - properties, specifying, 202
 - Leaves function, 207
 - Root function, 206
 - Scope statements, 203-206
- AssociatedMeasureGroupID property, KPI object, 264**
- Associated_Measure_Group property, CREATE KPI statements, 269**
- Attach command, database scalability, 472**
- AttributeHierarchyDisplayFolder property, 61**
- AttributeHierarchyEnable parameter, cube dimensions, 69**
- AttributeHierarchyEnabled property, 61**
- AttributeHierarchyOptimizedState parameter, cube dimensions, 69**
- AttributeHierarchyOptimizedState property, 61**
- AttributeHierarchyOrdered property, 61**
- AttributeHierarchyVisible parameter, cube dimensions, 69**
- AttributeHierarchyVisible property, 61**
- AttributeID parameter, cube dimensions, 69**
- AttributeID property, AttributePermission object, 735**
- AttributePermission object**
 - AllowedSet property, 735-737, 740, 747, 750
 - AttributeID property, 735

DefaultMember property, 735
 DeniedSet property, 735- 737, 740, 747, 750
 VisualTotals property, 735, 740

attributes

data structures, 351
 BLOB stores, 355
 deleted member stores, 354-355
 key stores, 351, 353
 overview of, 358-359
 property stores, 353-354
 unary member stores, 355
 Dataltem object, 51-53
 dimensions, 44, 120-121
 discretization, 95-96
 hierarchies, 60-61, 122
 implicit overwrites, 176
 IsAggregatable property, attributes, 176
 key values of, 45
 members
 keys, 50-53
 names, 53
 populating, 47
 processing, 377
 plan development, 378
 plan execution, 379-383
 process data jobs, 382-383
 read data jobs, 381
 write data jobs, 383
 properties of, 45-47
 relating attributes, 47-56, 176
 flexible relationships, 55-56
 mandatory relationships, 55-56
 One-to-Many relationships, 55-56
 One-to-One relationships, 55-56
 optional relationships, 55-56
 rigid relationships, 54, 56
 relationships, 355
 indexes, 358
 map stores, 356-358
 tree of, 48-50
 Unknown Members, 51

Attributes collection, cubes, 68

Audit property, Create command, 767

Audit_Login event, 772

authentication

ADOMD.NET, 606
 connection security, 713-717
 constrained delegation, 716

authorization, connection security, 713

auto-exist tuples, 167-168

automatic MOLAP proactive caching scenario, 441

automation libraries, creating COM assemblies, 245

AutoRestart property, Create command, 767

AverageOfChildren aggregation function, 229

axes (queries)

defining, 141
 listing, 140
 multidimensional space, defining coordinates in, 142
 naming, 141
 slicer axis, 144

Axes section (MDDataset-formatted results), 590-593

axis objects, ADOMD.NET, 639

AxisInfo section (MDDataset-formatted results), 590

B

BackColor property, measures, 77

BACK_COLOR property, 164

Batch command

nontransactional Batch commands, 498
 ProcessAffectedObjects parameter, 499-501
 syntax of, 496
 Transaction parameter, 497

BEGIN TRANSACTION statements, 299

BeginTransaction command, 489-493

BI Dev Studio, 468

Cube Wizard
 building, 336-339
 relational schemas via, 336-339

DSV, 319

ETL packages, testing, 409

- linked dimensions, creating, 467
- Schema Generation Wizard
 - building, 337-341
 - relational schemas via, 337-341

Binary XML

- data access, 570
- enabling, 410

binding objects, 321

- Column bindings, 321-322
- loading relational data via, 307-308
- measure bindings, defining via DDL, 322
- query bindings, 326-327
- Row bindings, 323
- tabular bindings, 324, 326

bit stores, structure of, 348

bitmap indexes

- measure groups, 559
- queries, 555

BLOB stores, attribute data structures (dimensions), 355

block commands, 502

- maxParallel parameter, 501
- parallel execution blocks, syntax of, 501

Browser tab (Dimension Designer), 124

BudgetCubeUsers role, 724, 728

buffer size, 410

BufferMemoryLimit server configuration property, 515

BufferRecordLimit server configuration property, 515

Build Decoding Table jobs, 384

Build Index jobs, 384-385

Build option, 133

building cube perspectives, 130

ByAccount aggregation function, 229-232

C

cache system memory model, 509

CacheRowsetRows server configuration property, 517

CacheRowsetToDisk server configuration property, 517

caches

- AdomdCacheExpiredException class, 666-667

- data caches, 548-550

- dimension caches, 548-550

- flat caches, 550

- formula caches, 550

- global scope caches, 535

- levels of, 548

- measure group caches, 548-550

- metadata, ADOMD.NET, 615-617

- proactive, 436

- considerations for, 448

- data latency, 437

- HOLAPenario, 442

- long-running MOLAP processing, 439

- MOLAP, 440-442

- monitoring activity, 448

- notifications, 445-446

- ROLAP, real-time scenario, 443

- scheduling object processing, 443

- updates, 438, 443, 447-448

- session scope caches, 536

Calculate command, cube-based MDX calculations, 215

Calculate Non Empty Begin event, 784

Calculate statement, 232, 234, 236

calculations

- cells, 190, 208-209

- logical plans, 542-545

- physical plans, 546-547

- values (MDX queries), 542

- measures, 193

- members, 190-192

- calculated measures, 193

- CREATE_MEMBER statements, 193-196

- creating, 194

- defining, 193

- DROP_MEMBER statements, 196

- MDX scripts, 193

- NON_EMPTY_BEHAVIOR property, 197-198

- queries, 196

- SELECT clauses, 196
- WITH clauses, 193, 196
- scope, 209, 532
 - global, 535-538
 - sessions, 117, 536-538
- Calculations collections, 72**
- Call statements, 246**
- callbacks, stored procedures, 257-260**
- calling stored procedures from MDX, 246**
- CanAdd method, unnamed object collections, 673**
- Cancel command**
 - session deletion, 494
 - syntax of, 495
- canceled command execution, 494-496**
- CanProcess method, 678**
- Caption property, 276**
- CaptionIsMDX property, 276**
- cardinality of dimensions, 18**
- Cardinality property, 55-56**
- Catalog object, 38**
- CellInfo section (MDDataset-formatted results), 590-592**
- CellPermission object, 730, 751**
 - Access property, 752
 - Expression property, 752
- cells, 190, 208-209**
 - assignments, 190, 198
 - assignment operator, 199-201
 - calculated cells, 208-209
 - calculation properties, 202
 - Leaves function, 207
 - Root function, 206
 - Scope statements, 203-206
 - calculation plans (MDX queries), 542
 - logical plans, 542-545
 - physical plans, 546-547
 - CREATE_CELL_CALCULATION statements, 199
 - empty cells, nulls, 170-172
 - measures of, 22
 - properties
 - MDX queries, 162-164
 - SELECT statements, 162
 - queries, calculating in, 147-148
 - security, 731-733
 - CellPermission object, 751-752
 - contingent cell security, 756
 - defining, 751-754
 - dynamic cell security, 758-760
 - MDX scripts, 748, 760
 - testing, 754
 - WITH_CELL_CALCULATION clauses, 199
 - writing to cubes
 - nonupdatable cells, 298-299
 - updatable cells, 298-299
 - UPDATE_CUBE statements, 292-298
- Cells section (MDDataset-formatted results), 590-593**
- CellSet object**
 - ADOMD.NET, 304, 636-639, 654
 - axis objects, 639
 - displaying multidimensional data in grid format, 640-641
 - Hierarchy object properties, retrieving, 641-644
 - object symmetry, 644-647
 - OlapInfo object, 637-639
 - uses for, 654
- CELL_PROPERTES clause, SELECT statements, 163-164**
- .Children function, 154**
- Clear method, unnamed object collections, 673**
- client/server architecture, data access**
 - HTTP, 571-573
 - HTTPS, 571-573
 - offline, 573
 - TCP/IP, 569-571
- Clone method, AMO, 673**
- "closest wins" rule, dimension-based calculations, 233-236**
- CLR (common language runtime) assemblies**
 - creating, 239-242
 - sending, 244
- ClrAssembly object properties, 239**
- clusters, failover clusters**
 - external data access security, 721
 - synchronizing, 814

code access security, 248, 714

Collation property, 52

dimension attribute member names, 53

rule of ordering, 41

collections

aggregations, 425

empty collections, DDL rules for, 41

major object collections

Drop method, 677, 680

LastSchemaUpdate method, 677

processable major objects, 678

Refresh method, 677

Update method, 677, 683

UpdateMode parameter, 677

XmlaWarningCollections collection, 677

named object collections

Add() method, 674

Add(string name) method, 675

Add(string name, string id) method, 675

Contains(string id) method, 675

ContainsName(string name) method,
675

Find(string id) method, 675

FindByName method, 675

GetByName(string name) method, 675

GetNewID method, 675

GetNewName method, 675

IndexOf(string id) method, 676

IndexOfName(string name) method, 676

IsValidID method, 676

IsValidName method, 676

Item method, 676

Remove method, 676

unnamed object collections

Add method, 673

CanAdd method, 673

Clear method, 673

Contains method, 673

Count method, 673

IndexOf method, 673

Insert method, 674

Item method, 674

Item property, 674

Move method, 674

properties of, 673

Remove method, 674

RemoveAt method, 674

Column bindings, 321-322

Column property, Drillthrough Action object, 288

ColumnBinding object, 321-322

COM (Component Object Model), 237, 245

ComAssembly object properties, 245

Command parameter (Execute method), 588

commands

asynchronous execution/cancellation of,
658-662

block commands, 502

maxParallel parameter, 501

parallel execution block syntax, 501

canceling execution of, 494-496

execution of, 477, 480

grouping, 496

monitoring, 818-819

objects

creating, 484

deleting, 486

editing, 484-485

locking, 491-494

processing, 486, 489

transactional commands, 489-490

**CommandStream property, AdomdCommand
class, 631**

**CommandText property, AdomdCommand class,
631**

Command_Begin event, 770

commit locks, 492-494

COMMIT TRANSACTION statements, 299-301

CommitTransaction command, 489-490

composite keys, 50

defining, 52-53

mapping, 415-416

compression

data format, 571

features (OLE DB provider), enabling, 410

stores, structure of, 349-350

conceptual data models, 9, 37**DDL**

- Customer dimension attribute definitions, 45
- Database dimension definitions, 43
- dimension attribute composite key definitions, 52-53
- dimension attribute tree definitions, 48-50
- major objects, 38-39
- minor objects, 38
- rules for empty collections, 41
- rules of inheritance, 42
- rules of ordering, 41
- specifying default properties, 41
- XML and, 37

multilanguage support, 39, 41

Condition property, defining scope of actions, 274**configuring**

data source access, 719-720

DSVs, 114-115

flight recorder trace, 775-776

MDX

- calling stored procedures, 246-247
- CLR assemblies, 239-242
- implementing COM assemblies, 245
- sending CLR assemblies, 244
- stored procedures, 239

multidimensional models

cubes, 124-136

data sources, 110-111

DDL files, 112-113

dimensions, 118-123

DSVs, 114-116

modifying data sources, 111-112

query logs, 428-430

Connection property, AdomdCommand class, 630-631**ConnectionException class, 708****ConnectionID property, Progress Report event, 449****connections**

data source objects, 314

MdxMissingMemberMode property, 167

monitoring, 818-819

security, 713-714

anonymous access, 716

authentication, 715-717

constrained delegation, 716

DISCOVER_CONNECTIONS requests, 715

HTTP, 715-717

TCP/IP, 714-715

ConnectionString property, 311, 314, 604**constrained delegation, 716****Contains method, unnamed object collections, 673****Contains(string id) method, named object collections, 675****ContainsName(string name) method, named object collections, 675****Context object, 252****contingent cell security, 756****converting**

currencies example (measure expressions), 105, 107

writeback partitions to regular partitions, 303

coordinates

current, 174

defining in multidimensional space, 141-142

CoordinatorQueryBalancingFactor property, 558**COUNT aggregations, measures, 79****Count method**

ADOMD.NET, 612

unnamed object collections, 673

Count option, metadata objects, 252**Create command, 245**

AllowOverwrite parameter, 484

linked objects, 456

properties of, 767

Scope parameter, 484

syntax of, 484

trace, 766-767

CREATE KPI statements, 268-270**CreatedTimestamp property, partitions, 365****CREATE_CELL_CALCULATION statements, 199**

CREATE_MEMBER statements, calculated members, 193-196

CREATE_SET statements, named sets, 210, 213

CREATE_SUBCUBE statements, 180-184, 200

CrossJoin sets, 151-152

Cube objects, 39

Cube Wizard, 124

cube-based KPI (Key Performance Indicators), 262

KPI objects, creating, 266-268

MDX expressions for

goal, 265

status, 265

trend, 265

value, 264

CubeAttributeSecurity data structure, 748

CubeInfo section (MDDataset-formatted results), 589

CubePermission objects, 729, 754

defining, 726

security, 726

CubePermissions parameter, cubes, 64

cubes, 63, 251, 364

Actions collections, 72

Actions parameter, 64

asynchronous processing, 689-692

Attributes collection, 68

Calculations collections, 72

cells

nonupdatable cells, 298-299

updatable cells, 298-299

writing data to via **UPDATE_CUBE** statements, 292-298

collections of, 63

creating, 124-125

CubePermissions parameter, 64

data structure overview, 375

defining, 64, 66

dimension cubes, creating, 71

Dimensions collection, 65-66, 72

AttributeHierarchyEnable parameter, 69

AttributeHierarchyOptimizedState parameter, 69

AttributeHierarchyVisible parameter, 69

AttributeID parameter, 69

attributes of, 68-69

DimensionID parameter, 67

Enabled parameter, 70

hierarchies of, 69

HierarchyID parameter, 70

HierarchyUniqueNameStyle parameter, 67

ID parameter, 67

MemberUniqueNameStyle parameter, 67

multiple roles in, 70-71

Name parameter, 67

OptimizedState parameter, 70

role-playing dimensions, 70-71

Visible parameter, 70

Dimensions parameter, 64

DRILLTHROUGH columns, defining, 287-290

Hierarchies collection, 68

KPIs

collections, 72

parameter, 64

local cubes, **ADOMD.NET** connections, 605

MDX calculations, 189

assignments, 190, 198-209

calculated members, 190-198

dimension-based calculations, 190

FREEZE statements, 218

named sets, 190, 209-214

order of execution in, 215-220

scripts, 191-193

semi-additive measures, 190

MDXScripts parameter, 64

measure groups, 75, 81, 375

defining, 82-84

dimensions, 84-89

Dimensions property, 82

EstimatedRows property, 83

granularity, 81-89

IgnoreUnrelatedDimensions property, 82

Measures property, 82

properties of, 82-83

Type property, 82

MeasureGroups

- collections, 72
- parameters, 64
- measures, 75
 - AggregateFunction property, 77-78
 - BackColor property, 77
 - COUNT aggregations, 79
 - DataType property, 77-78
 - defining, 76-78
 - Description property, 76
 - DisplayFolder property, 77
 - DISTINCT COUNT aggregations, 79-80
 - FontFlags property, 77
 - FontName property, 77
 - FontSize property, 77
 - ForeColor property, 77
 - FormatString property, 77
 - ID property, 76
 - MAX aggregations, 79
 - MeasureExpression property, 77
 - MIN aggregations, 79
 - Name property, 76
 - properties of, 76
 - Source property, 76
 - SUM aggregations, 78
 - Translations property, 76
 - Visible property, 76
- Measures dimension, 75
- modifying, 125-130
- optimizing logical space in, 541
- parameters of, 63-64
- partitions
 - aggregation indexes, 373
 - building indexes, 370-371
 - data, 368
 - decoding attributes, 369
 - defining, 364, 367
 - metadata files, 374
 - properties of, 364-366
 - remote partitions, 374-375
 - slices, 368
- perspectives, 72
 - building, 130
 - defining, 72-74
 - Perspectives parameter, 64
- processing, 395
 - aggregations, 393-395
 - building aggregations, 397
 - building indexes, 397
 - data processing, 391-393
 - incremental partition updates, 398-399
 - indexes, 393
 - lazy processing, 397
 - loading data into partitions, 396-397
 - merging partitions, 399-400
 - ProcessAdd option, 396
 - ProcessClear option, 396
 - ProcessClearIndexes option, 396
 - ProcessData option, 396-397
 - ProcessFull option, 396
 - ProcessIndex option, 396
 - storing data in partitions, 390
- translations, 131-132
- Translations collection, 67
- viewing, 133-136
- virtual cubes, creating, 468
- Visible parameter, 63
- Cube_Dimension_Attribute object, 427**
- currency conversion example (measure expressions), 105, 107**
- current coordinates, 174**
- CurrentMember function, 149, 185**
- CurrentTimeMember property, KPI object, 264**
- Current_Time_Member property, CREATE KPI statements, 269**
- custom members**
 - formulas, 225-227
 - MDX queries, 162
- Customer dimension**
 - attributes of, 56
 - key attributes, defining via DDL, 45
 - many-to-many dimensions, 103
- CustomRollupColumn property, 225**
- CustomRollupPropertiesColumn property, 226**

D

data access

- ADO.NET, 576
- ADOMD.NET, 576
- AMO, 577
- Analysis Services Libraries, 575
- Binary XML format, 570
- client applications, building, 574
- compressed data format, 571
- DSO, 577
- external data access security, 714
 - changing service logon accounts, 720
 - choosing service logon accounts, 718
 - configuring data source access, 719-720
 - failover cluster operations, 721
 - named instances (SQL Server Browser), 721
- HTTP, 571-573
- HTTPS, 571-573
- offline, 573
- OLE DB, 576
- SOAP, 569-570
- TCP/IP, 569-571
- text format, 570
- XML/A, 570, 574-575

data caches, 548-550

data decoder

- measure groups, 559
- queries, 555

data flow components (SSIS), 410

- dimension-loading packages, 410-411
- partition-loading packages, 414

data flow task (ETL process), 408-410

Data Format property, 51

Data jobs, queries, 565

data latency

- MOLAP to MOLAP transition mode, 437
- MOLAP-ROLAP-MOLAP transition mode, 437, 439

data ordering functions, 158

data retrieval

- Data jobs, 565
- KPI data, 270

- Lookup jobs, 565

- managing, 576

- measure groups, 556, 558

- DISTINCT_COUNT measures, 560-563

- linked measure groups, 563

- measure groups with indirect dimensions, 564-566

- query execution process, 554-555

- remote partitions, 563

- ROLAP partitions, 559, 562

data security, 713

Data Size property, 51

data source objects (Analysis Services), 310

- DDL, defining via, 310

- properties of, 310

- security, 312-313

- connection timeouts, 314

- pooling connections, 314

- storing private information, 314

Data Source View. See DSV

Data Source Wizard, 110-111

data sources

- creating, 110-111

- modifying, 111-112

data storage

- bit stores, structure of, 348

- compressed stores, structure of, 349-350

- data stores, structure of, 346

- file stores, structure of, 346, 348

- hash stores, 350

- OLAP farms, 453-455

- partitions

- HOLAP storage mode, 390

- MOLAP storage mode, 390

- ROLAP storage mode, 390

- queries, execution process, 554-555

- string stores, structure of, 348

data stores, structure of, 346

data warehouses, 11

- data's life cycle in, 407

- ETL process, 407-408

- relational database schemas, 329-331

data, loading, 407-408

- direct-load ETL packages, 409

- SSIS, 408-414

- testing packages, 409

DataAdapter class, DataSet object, 647**Database dimension, 43****Database Engine Tuning Advisor, 331, 334****Database object, 38****DatabaseConnectionPoolConnectTimeout property, 314****DatabaseConnectionPoolMax property, 315****DatabasePermission object, 728-729****databases**

- assemblies, 243

- Attach command, 472

- DbStorageLocation property, 473

- deploying

- DDL command, 809-811

- Deployment Wizard, 805-807

- Synchronize command, 805-808

- Synchronize Database Wizard, 809

- Detach command, 470-471

- HOLAP connections, 11

- MOLAP connections, 11

- multidimensional models

- cubes, 124-136

- data sources, 110-111

- DDL files, 112-113

- dimensions, 118-123

- DSVs, 114-116

- modifying data sources, 111-112

- read-only databases, 473

- ROLAP connections, 11

- roles

- managing, 730

- security, 723

- shared scalable databases, 470

- synchronizing

- DDL command, 809-811

- Synchronize command, 805-808

- Synchronize Database Wizard, 809

Datacache objects, 563-565**Dataltem objects**

- composite key definitions, 52-53

- dimensions

- attribute member names, 53

- attribute membership keys, 51-53

- properties of, 51

DataMember property, 93**DataSet object**

- ADOMD.NET, 647-649

- AdomdDataAdapter object, 647

- DataAdapter class, 647

datasets, 647**DataSource objects, 317-319, 580**

- binding objects, 321

- Column bindings, 321-322

- query bindings, 326-327

- Row bindings, 323

- tabular bindings, 324-326

- DDL, defining via, 310

- loading relational data via, 307-309

- named calculations, 320-321

- named queries, 319-321

- properties of, 310-311

Datasource property, ADOMD.NET, 604**DataSourceImpersonationInfo property, 313, 720****DataSourcePermission object, 729****DataSourceView object, 307****DataType property, 51**

- measures, 77-78

DbStorageLocation property, 473**DDL (Data Definition Language)**

- Actions, creating, 277, 279

- Administrators role, defining, 722-723

- aggregations, defining, 424-425

- Attach functionality, 472

- BudgetCubeUsers role, defining, 724, 728

- cell security, defining, 753

- Column bindings, defining, 321-322

- CubePermission objects, defining, 726, 754

- Customer dimension, defining key attribute of, 45

- Database dimension, defining, 43

- Databases, synchronizing, 809-811

- DataSource objects, defining, 310

- default properties, specifying, 41

- Detach functionality, 471
- dimension attributes
 - composite key definitions, 52-53
 - trees, 48-50
- DimensionPermission object, defining, 742-744
- Drillthrough Action objects, defining, 289-290
- DSV, defining via, 318-319
- empty collections, rules for, 41
- ErrorConfiguration object, defining, 401
- files, modifying, 112-113
- ImpersonationInfo objects, defining, 312
- ImpersonationInfo property, defining, 719-720
- KPI objects, creating via, 266-268
- linking
 - dimensions, 458
 - measure groups, 462
- major objects, 38-39
- MDX scripts, creating, 191
- measures
 - bindings, defining, 322
 - defining, 76
 - expressions, 106
 - groups, 82-89
- minor objects, 38
- partitions
 - defining, 364
 - defining via query bindings, 326-327
 - defining via table bindings, 326
- ProactiveCaching object, defining, 437
- remote partitions, 464-465
- requests, creating assemblies, 241-242
- Restore command, Synchronize command's similarities to, 811
- rules
 - of inheritance, 42
 - of ordering, 41
- Source objects, 458
- Synchronize command
 - defining, 810
 - options for, 810
 - Restore command's similarities to, 811
 - synchronizing, 812-814
 - trace, creating, 766-767
 - XML and, 37
- deadlocks, troubleshooting, 491**
- debugging ETL packages**
 - dimension-loading packages, 413
 - partition-loading packages, 415
- decoding tables**
 - building, 384
 - hierarchy data structures (dimensions), 362-363
- default libraries, MDX, 260**
- default members, 144-146**
- default properties, specifying in DDL, 41**
- DEFAULT property, Drillthrough Action object, 288**
- Default property, ImpersonationInfo object, 313**
- DefaultBufferMaxRows property (data flow task), 410**
- DefaultBufferSize property (data flow task), 410**
- DefaultMember property, AttributePermission object, 735**
- defining cubes, 128-132**
- delegation, constrained, 716**
- Delete command**
 - IgnoreFailures parameter, 486
 - syntax of, 486
- deleting**
 - member stores, attribute data structures (dimensions), 354-355
 - objects, 486
 - sessions, 494
 - writeback partitions, 303
- DeniedSet property, 735-737, 740, 747, 750**
- DependenciesCalculator class, 683-684**
- dependent objects (AMO), 678-681**
 - DependenciesCalculator class, 683-684
 - impact analysis, 681-682
- deploying projects, 135**
- Deployment option, 134**
- Deployment Wizard, 805-807**

Descendants function, 155**Description property, 38**

- measures, 76
- named minor objects, 674

Detach command, database scalability, 470-471**dimension attribute keys, mapping composite keys, 415-416****dimension cubes, creating, 71****Dimension Designer**

- Browser tab, 124
- Translations tab, 123

Dimension objects, 39**Dimension Processing Destination editor, 411****dimension security, 731-732**

- architecture of, 748
- AttributePermission object
 - AllowedSet property, 735-737, 740, 747, 750
 - AttributeID property, 735
 - DefaultMember property, 735
 - DeniedSet property, 735-737, 740, 747, 750
 - VisualTotals property, 735, 740
- defining, 734
- dynamic dimension security, 746-747
- MDX scripts, 748
- testing, 744
- user interface, defining via, 742-744

Dimension Wizard, 118-119**dimension-based calculations**

- custom member formulas, 225, 227
- order of execution, 232-236
- semi-additive measures, 227-232
- unary operators, 221-224

dimension-loading ETL packages

- debugging, 413
- SSIS, creating in, 410-411
- testing, 413

Dimension1 object, proactive caching, 437**dimensionality (tuples), 142****DimensionID parameter, cube dimensions, 67****DimensionPermission object, 729, 742-744****dimensions, 251**

- attributes
 - data structures, 351-359
 - processing, 377-383
 - relationships, 355-356
- caches, 548-550
- calculations in, 190
- creating, 118-119
- cubes, 128
- Customer dimension
 - attribute relationships, 56
 - defining key attributes via DDL, 45
- data structures
 - attribute data structures, 351-359
 - attribute relationships, 355-356
 - hierarchy data structures, 360-363
- Database dimension, 43
- decoding tables, building, 384
- direct dimensions, 97
- full dimension processing, 387
- group dimensions
 - direct dimensions, 97
 - indirect dimensions, 97-105
- hierarchies
 - data structures, 360-363
 - processing, 383-384
- incremental dimension processing, 387
- indexes, building, 384-385
- indirect dimensions, 97
 - many-to-many dimensions, 102-105
 - referenced dimensions, 98-102
- linked dimensions, 455-456
 - creating, 467
 - OLAP farms, 457-460
- many-to-many dimensions, 102
 - defining, 103-104
 - measure groups in, 105
 - queries, 104
- modifying, 119-123
- ordinals, retrieving via ADOMD.NET, 615
- parent-child dimensions, 389
- processing
 - activity, tracing, 776-779

- schemas, 385
- updates, 387-388
- referenced dimensions
 - defining, 100-102
 - Geography dimension, 98-99
 - Materialization property, 99
- relational reporting-style dimensions, aggregations, 420-422
- ROLAP processing, 388
- dimensions (Multidimensional Data Model), 13**
 - attributes of, 13, 20, 44
 - DataItem object, 51-53
 - hierarchies of, 60-61
 - key values of, 45
 - member keys, 50-53
 - member names, 53
 - populating, 47
 - properties of, 45-47
 - related attributes, 47-56
 - tree of, 48-50
 - Unknown Members, 51
 - cardinality of, 18
 - hierarchies of, 21-22
 - All level, 60
 - attribute hierarchies, 60-61
 - coordinates in multidimensional space, 141
 - defining, 57
 - Key level, 60
 - member references, 59
 - natural hierarchies, 57-58
 - members of, 17
 - attributes of, 18
 - values of, 17
 - size of, 18
- Dimensions collection, cubes, 65-66**
 - AttributeHierarchyEnable parameter, 69
 - AttributeHierarchyOptimizedState parameter, 69
 - AttributeHierarchyVisible parameter, 69
 - AttributeID parameter, 69
 - attributes of, 68-69
 - cubes, 72
 - DimensionID parameter, 67
 - Enabled parameter, 70
 - hierarchies of, 69
 - HierarchyID parameter, 70
 - HierarchyUniqueNameStyle parameter, 67
 - ID parameter, 67
 - MemberUniqueNameStyle parameter, 67
 - multiple roles in, 70-71
 - Name parameter, 67
 - OptimizedState parameter, 70
 - role-playing dimensions, 70-71
 - Visible parameter, 70
- Dimensions parameter, 64**
- Dimensions property, 82**
- DIMENSION_PROPERTIES clause**
 - MDX queries, 162
 - SELECT statements, 163
- direct dimensions, 97**
- direct relationships (dimension attributes), 48**
- direct-load ETL packages, 409**
- disconnected mode (AMO), 693-694**
- Discover Begin event, 772**
- Discover method**
 - Properties parameter, 584-587
 - RequestType parameter, 583
 - Restrictions parameter, 583
 - signature of, 583
- discovering**
 - actions, 279-283
 - KPI, 270-271
 - server-state, 776
- Discover_Commands DMV, 818-819**
- Discover_Connections DMV, 818-819**
- DISCOVER_CONNECTIONS requests, 715**
- Discover_Object_Activity DMV, 820**
- DISCOVER_PROPERTIES requests, 772**
- Discover_Sessions DMV, 818-819**
- discretization (attributes), 95-96**
- DisplayFolder property, measures, 77**
- DISTINCT_COUNT aggregate function, 395, 400**
 - measures, 79-80, 560-563
 - processing partitions in measure groups, 392

distributed storage

- linked objects, 32
- remote partitions, 32
- thick clients, 33
- thin clients, 33

distributing memory, 512**DLL (dynamic link library), 238****DMV (Dynamic Management Views), 816**

- Discover_Commands, 818-819
- Discover_Connections, 818-819
- Discover_Object_Activity, 820
- Discover_Sessions, 818-819
- queries, 817
- server-state discover requests, 776

domains

- applications, 244
- defining, 44

Drillthrough Action object

- defining, 289-290
- properties of, 288

DRILLTHROUGH statements, 284

- cube columns, defining in, 287-290
- CubePermission object, 730
- requests for, 285-287
- syntax of, 285

DROP KPI statements, 268**Drop method, 677, 680****DROP_MEMBER statements, 196****DROP_SET statements, 210****DROP_SUBCUBE statements, 180****DSO (Decision Support Objects), 29, 577****DSV (Data Source View), 308, 317-319**

- creating, 114-115
- DDL, defining via, 318-319
- modifying, 115-116
- objects, 321
 - Column bindings, 321-322
 - query bindings, 326-327
 - Row bindings, 323
 - tabular bindings, 324-326
- named calculations, 320-321
- named queries, 319-321

DsvTableBinding object, 321, 324**dynamic cell security, 758-760****dynamic dimension security, 746-747****dynamic inheritance, 42****dynamic name sets, 213-214****E****economic memory management model, 504**

- Income parameter, 508
- InitialBonus parameter, 508
- MaximumBalance parameter, 508
- MinimumBalance parameter, 508
- parameters of, 508
- Tax parameter, 508

editing

- named calculations, 117
- objects, 484-485

Employee attribute, 93-94**Employee dimension, 92-94****empty cells, nulls and, 170-172****empty collections, DDL rules for, 41****empty sets, 165****Enabled parameter, cube dimensions, 70****error handling**

- ADOMD.NET, 662-663
 - AdomdCacheExpiredException class, 666-667
 - AdomdConnectionException class, 666
 - AdomdErrorResponseException class, 663-665
 - AdomdUnknownResponseException class, 666

AMO, 706

- ConnectionException class, 708
- OperationException class, 706-707
- OutOfSyncException class, 708-709
- ResponseFormatException class, 707

ErrorConfiguration object

- defining, 401
- properties of, 402-403

XML/A, 593

- cell calculation errors, 597
- errors occurring after start of response serialization, 596

- MDX errors, 595-596
 - whole method failure errors, 594
- error messages, 472**
- ErrorConfiguration object**
 - defining, 401
 - KeyDuplicate property, 403
 - KeyErrorAction property, 402
 - KeyErrorLimit property, 402
 - KeyErrorLimitAction property, 402
 - KeyErrorLogFile property, 402
 - KeyNotFound property, 403
 - NullKeyConvertedToUnknown property, 403
 - NullKeyNotAllowed property, 403
 - properties of, 402-403
- ErrorConfiguraton property, partitions, 365**
- EstimatedCount property, 47**
- EstimatedRows property**
 - measure groups, 83
 - partitions, 365
- EstimatedSize property, partitions, 365**
- ETL (extraction, transformation, and loading) process, 407-408**
 - data flow task, 408-410
 - direct-load ETL packages, 409
 - SSIS
 - building in, 408
 - data processing in, 410-414
 - dimension-loading packages, 410-413
 - partition-loading packages, 414-415
 - testing packages, 409
- Event Selections Tab (Trace Properties dialog)**
 - EventSubclass column, 769
 - IntegerData column, 769
 - TextData column, 769
- events**
 - proactive caching events, 448
 - trace, viewing in, 770-772
- EventSubclass column (Events Selection Tab), 769**
- Except sets, 150**
- Execute method, 587**
 - AdomdCommand class, 632-636
 - Command parameter, 588
 - Properties parameter, 588-593

- ExecuteCellSet method, 632**
- ExecuteDataReader method, 632-633**
- ExecuteNonQuery method, 632, 635**
- ExecuteXmlReader method, 632-635**
- existing dimensions, modifying, 119-123**
- Exists function, 183**
- Expression property**
 - actions, defining function of, 274
 - CellPermission object, 752
- Expressions, 153**
 - CurrentMember function, 185
 - sub_cube_expressions, 199-200
- external data access security, 714**
 - data source access, configuring, 719-720
 - failover cluster operations, 721
 - named instances (SQL Server Browser), 721
 - service logon accounts
 - changing, 720
 - choosing, 718
- ExternalCommandTimeout property, 314**
- Extract sets, 152**

F

- failover clusters**
 - external data access security, 721
 - synchronizing, 814
- file stores**
 - memory management, 510
 - structure of, 346-348
- FileSize property, 775**
- Filter function, 155-157**
 - MDX queries, 171
 - static name sets, 211
- FilterByRegEx function, 256**
- filtering sets, 155-157, 256**
- Find method, ADOMD.NET, 612**
- Find option, metadata objects, 255**
- Find(string id) method, named object collections, 675**
- FindByName method, named object collections, 675**

FirstChild aggregation function, 228**.FirstChild function, 155****FirstNonEmpty aggregation function, 228****flat caches, 550****flexible aggregations, 422-423****flexible relationships (dimension attributes), 55-56****flight recorder trace, 765, 773**

configuring, 775-776

properties of, 775

FontFlags property, 77**FontName property, 77****FontSize property, 77****ForceRebuildInterval property, 440-441, 449****ForeColor property, 77****FORE_COLOR property, 164****FormatString property, 77****formula caches, 550****four-tier architectures (Multidimensional Data Model), 28, 31****fragmentation, preventing, 511****fragments (MDX), 153****FREEZE statements, 218****FROM clauses**

SELECT statements, 140

subcubes, 182

full dimension processing, 387**full updates, 447-448****functions**

aggregation functions, 24

.Children function, 154**Descendants function, 155****Filter function, 155-157****.FirstChild function, 155****hierarchy navigation, 153-155****.LastChild function, 155**

methods, syntax of, 153

Order function, 158

properties, syntax of, 153

G**GeneralAllocator memory allocators, 511****Geography dimensions, 98-99****Get Data from Cache event, 784****GetByName(string name) method, 675****GetEnumerator method, 613****GetEnumerator option, metadata objects, 255****GetMembers() overload method, 623-624****GetNewID method, 675****GetNewName method, 675****GetSchemaDataSet method, 625-630****global calculation scopes, 532, 535**

caches, 535

lifetimes of, 536-538

Global level (memory allocators), 512**Goal property, 263-265****granularity (measure groups), 81-89****grids, displaying multidimensional data in via ADOMD.NET, 640-641****group dimensions**

direct dimensions, 97

indirect dimensions, 97

many-to-many dimensions, 102-105

referenced dimensions, 98-102

H**hash stores, 350****helper objects, 252****HideMemberIf property, 57****hiding attribute hierarchies, 122****hierarchies**

attributes, applying, 122

dimensions, 21-22

All level, 60

attribute hierarchies, 60-61

data structures, 360-363

defining, 57

Key level, 60

member references, 59

multidimensional space, defining coordinates in, 141

natural hierarchies, 57-58

processing, 383-384

MDX functions, navigating via, 153-155

user-defined hierarchies, 141

Hierarchies collection, 68
HierarchyID parameter, 70
HierarchyUniqueNameStyle parameter, 67
highest pass wins rule, 216
HighMemoryPrice (economic memory management model), 504
HOLAP (hybrid online analytical processing), 11
 partitions, 390
 aggregations in, 397
 indexes in, 397
 proactive caching, 442
HTTP (Hypertext Transfer Protocol)
 connection security, 715-717
 data access via, 571-573
HTTPS (HTTP Secure), 571-573
hypercubes, 63

I

ICloneable interface, 672
ICollection interfaces, 252
 ADOMD.NET collections, 612
ID parameters, 67
ID property, 40
 Create command, 767
 major objects, 38
 measures, 76
 minor objects, 672-674
 permission objects, 727
IDataReader interface, 647
IDispatch interface, 245
IEnumerable interfaces, 252
 ADOMD.NET collections, 612
IFormattable interface, 674
Ignore Case property, 41
IgnoreFailures parameter, Delete command, 486
IgnoreUnrelatedDimensions property, 82
IMajorObject interface, 676
impact analysis, 681-682
ImpactAnalysis property
 requests, example of, 499
 responses, example of, 500-501
ImpersonateAccount property, 313
ImpersonateCurrentUser property, 313
ImpersonateServiceAccount property, 313
ImpersonationInfo object
 DDL, defining via, 312
 properties of, 312-313
ImpersonationInfo property, 250, 719-720
ImpersonationLevel property, 715
implementing assemblies, COM, 245
implicit overwrites, 176
INamedComponent interface, 674
Income parameter, 508
incremental dimension processing, 387
incremental updates
 full updates versus, 447-448
 partitions, 398-399
indexes
 aggregation indexes, 373
 attribute relationships (dimensions), 358
 bitmap indexes
 measure groups, 559
 queries, 555, 559
 building, 384-385, 518-519
 partitions
 building, 370-371, 393
 cube processing, 397
 HOLAP data storage mode, 397
 relational databases, 331-332
IndexOf method
 named object collections, 676
 unnamed object collections, 673
IndexOfName(string name) method, 676
indirect dimensions, 97
 many-to-many dimensions, 102
 defining, 103-104
 measure groups in, 105
 queries, 104
 referenced dimensions
 defining, 100-102
 Geography dimension, 98-99
 Materialization property, 99
indirect relationships (dimension attributes), 48

inheritance

- permission objects, 727
- rules of (DDL), 42

InitialBonus parameter, 508

Insert method, 674

IntegerData column (Events Selection Tab), 769

Integrated Windows authentication, 715

Intersect sets, 150

intrinsic member properties, 161

InvalidXmlCharacters property, 52

Invocation property, 276

IP (Internet Protocol) data access, 569

IP property, major objects, 670

IProcessable interface, 678

IRowsetChange OLE DB interface, 304

IsAggregatable property, 61, 427

Isolation property, 311

IsValidID method, 676

IsValidName method, 676

Item method

- ADOMD.NET, 612
- named object collections, 676
- unnamed object collections, 674

Item option, metadata objects, 252

Item property, 674

J-K

Job Coordinator, 565

Key level (dimension hierarchies), 60

key performance indicators, 251

key stores, 351-353

KeyColumn property, 46

KeyDuplicate property, 403

KeyErrorAction property, 402

KeyErrorLimit property, 402

KeyErrorLimitAction property, 402

KeyErrorLogFile property, 402

KeyNotFound property, 403

KPI (Key Performance Indicators), 262

- cubes, 64, 72, 262
- creating KPI objects, 266-268
- MDX expressions for, 264-265
- data retrieval, 270
- defining, 262

discovering, 270-271

objects

- AssociatedMeasureGroupID** property, 264
- creating, 266-268
- CurrentTimeMember** property, 264
- Goal** property, 263-265
- ParentKpiID** property, 264
- Status** property, 263-265
- StatusGraphic** property, 263
- Trend** property, 263-265
- TrendGraphic** property, 264
- Value** property, 263-264
- Weight** property, 264

querying, 271

scorecards, 262

session-based KPI

- CREATE** KPI statements, 268-270
- DROP** KPI statements, 268

L

Language property, 39, 54

LastChild aggregation function, 228

.LastChild function, 155

LastNonEmpty aggregation function, 228

LastProcessed method, 678

LastProcessed property

- partitions, 365
- proactive caching, 441

LastSchemaUpdate method, 677

LastSchemaUpdate property, 365

Latency property, 439

lazy processing, 397

Leaves function, 207

Level.GetMembers() function, 618-623

levels, 251

libraries

- automation, 245
- data access client applications, building, 575
- default, 260

linked dimensions, 455-456

- creating, 467
- OLAP farms, 457-460

linked measure groups, 107, 455-463

Linked Object Wizard, 127, 468

linked objects, 32

- applying, 127
- creating, 459
- source updates, 457

loading

- ETL process, 407-408
 - direct-load ETL packages, 409
 - SSIS, 408-414
 - testing packages, 409
- Binding object, 307-308
- DataSource object, 307-309
 - defining via DDL, 310
 - properties of, 310-311
- DataSourceView object, 307
- DSV (Analysis Services), 308
- ImpersonationInfo object
 - defining via DDL, 312
 - properties of, 312-313
- relational data, 310

local cubes

- ADOMD.NET connections, 605
- offline data access, 573

Locale property, 39

LocalSystem accounts, 718

lock command, 493-494

locking objects, 491

- commit locks, 492-494
- deadlocks, 491

LogDuration property, 775

LogFileAppend property, 767

LogFileName property, 767

LogFileRollover property, 767

LogFileSize property, 767

logical cell calculation plans (MDX queries), 542-545

logins

- server logins, 715
- service logon accounts
 - changing, 720
 - choosing, 718
 - read/write permissions, 719

long-command parsing thread pool, 522

long-running requests (parsing), 522

Lookup jobs, 565

low-latency MOLAP proactive caching scenario, 442

LowMemoryLimit (economic memory management model), 504, 507

M

major objects, 38-39

- collections of
 - Drop method, 677, 680
 - LastSchemaUpdate method, 677
 - processable objects, 678
 - Refresh method, 677
 - Update method, 677, 683
 - UpdateMode parameter, 677
 - XmlaWarningCollections collection, 677
- Cube objects, 39
- Database dimension, 43
- DataSource object
 - defining via DDL, 310
 - loading relational data via, 307
 - properties of, 310-311
- Dimension objects, 39
- ID property, 38-40
- ImpersonationInfo object
 - defining via DDL, 312
 - properties of, 312-313
- Name property, 38

major objects (AMO), 670

- collections of
 - Drop method, 677, 680
 - LastSchemaUpdate method, 677
 - processable objects, 678
 - Refresh method, 677
 - Update method, 677, 683
 - UpdateMode parameter, 677
 - XmlaWarningCollections collection, 677
- IMajorObject interface, 676
- processable objects, 678

MajorObjects level (memory allocators), 512

ManagedProvider property, 311

mandatory relationships (dimension attributes), 55-56**many-to-many dimensions, 102**

- defining, 103-104
- measure groups in, 105
- queries, 104

map stores, attribute relationships (dimensions), 356-358**mapping**

- account types, 230
- composite keys, 415-416
- remote partitions, 813

MapQuery, 555**MapQuery object, 559, 563****master servers, 32****Materialization property, 99****MAX aggregations, 79****MaxActiveConnections property, 311****MaximumBalance parameter, 508****maxParallel parameter, block commands, 501****MDDataset-formatted results**

- Axes section, 590-593
- AxisInfo section, 589-590
- CellInfo section, 590-592
- Cells section, 590-593

MDESCHEMA_ACTIONS schema rowset, 290

- discovering actions, 279-280
- mandatory restrictions on, 280
- optional restrictions on, 280

MDX (Multidimensional Expressions), 139, 237**axes**

- defining, 141
- defining coordinates in multidimensional space, 142
- listing, 140
- naming, 141
- slicer axis, 144

cells

- calculation in, 147-148
- security, 748, 760

cube calculations, 189

- assignments, 190, 198-209
- calculated members, 190-198

dimensions-based calculations, 190

- FREEZE statements, 218
- named sets, 190, 209-214
- order of execution in, 215-220
- scripts, 191-193
- semi-additive measures, 190

current coordinates, 174**CurrentMember function, 185****custom member formulas, 225-227****default libraries, 260****dimension security, 748****error handling via XML/A, 595-596****expressions, 153, 199-200****fragments, 153****functions**

- .Children function, 154
- Descendants function, 155
- Filter function, 155-157
- .FirstChild function, 155
- hierarchy navigation, 153, 155
- .LastChild function, 155
- method syntax, 153
- Order function, 158
- property syntax, 153

KPI

- queries, 271
- goal, 265
- status, 265
- trend, 265
- value, 264

nulls, 165

- auto-exist tuples, 167-168
- empty cells, 170-172
- existing tuples, 167-168
- Missing Member mode, 166
- nonexisting tuples, 167-168

objects

- by name, 158
- by qualified name, 159
- type conversion rules, 173-174
- by unique name, 159-160

queries

- cell calculation plans, 542-547
- cell properties in, 162-164

- DIMENSION_PROPERTIES clause, 162
- execution via ADOMD.NET, 632
- execution context, 147-148
- execution process, 528
- Filter function, 171
- member properties in, 161-162
- monitoring aggregation usage, 433-434
- optimizing logical cube space in, 541
- parsing, 530-531
- SELECT statements in, 162
- virtual set operation trees, 538-540
- requests, parameters in, 655-657
- security, 248-251
- SELECT statements
 - default members, 144-146
 - defining coordinates in multidimensional space, 141-142
 - FROM clause, 140
 - ON clause, 140
 - SELECT clause, 140
 - WHERE clause, 140, 146-57
- semi-additive measures, 227-232
- server object model, 251
 - metadata, 252, 255
 - operations on MDX objects, 255-257
- sets
 - algebra, 149
 - CrossJoin sets, 151-152
 - Except sets, 150
 - Extract sets, 152
 - filtering 155-157
 - Intersect sets, 150
 - Union sets, 149
 - WHERE clause, 177-179
- stored procedures
 - callbacks in, 257-260
 - calling, 246-247
 - configuring CLR assemblies, 239-242
 - creating, 239
 - implementing COM assemblies, 245
 - sending CLR assemblies, 244
- strong relationships, 176
- subcubes, 180-184
 - CREATE_SUBCUBE statements, 180-184
 - DROP_SUBCUBE statements, 180
 - FROM clauses, 182
 - NON_EMPTY clauses, 182
 - SELECT statements, 182
 - WITH clauses, 182
 - SubSelects, 180, 182, 184
 - unary operators, 221-224
 - visual totals, 186-187
 - WHERE clause, 177-179
- MdxMissingMemberMode property, 167**
- MDXScripts parameter, 64**
- measure groups, 75, 81, 126**
 - caches, 548-550
 - cubes, 375
 - defining, 82-84
 - dimensions, 84
 - attributes of, 87-89
 - defining, 86
 - granularity, 84-89
 - hierarchies of, 87
 - Dimensions property, 82
 - EstimatedRows property, 83
 - granularity, 81-89
 - IgnoreUnrelatedDimensions property, 82
 - linked measure groups, 107
 - many-to-many dimensions, 105
 - Measures property, 82
 - OLAP farms, 455-463
 - partitions, processing in, 392
 - properties of, 82-83
 - retrieving data from, 556-558
 - DISTINCT_COUNT measures, 560-563
 - linked measure groups, 563
 - measure groups with indirect dimensions, 564-566
 - Type property, 82
- MeasureExpression property, 77**
- MeasureGroups collections, 72**
- MeasureGroups parameters, 64**
- measures, 13, 75, 126**
 - AggregateFunction property, 77-78
 - BackColor property, 77

- bindings, defining via DDL, 322
- COUNT aggregations, 79
- currency conversion example, 105-107
- DataType property, 77-78
- defining, 76, 78, 106
- Description property, 76
- DisplayFolder property, 77
- DISTINCT COUNT aggregations, 79-80
- FontFlags property, 77
- FontName property, 77
- FontSize property, 77
- ForeColor property, 77
- FormatString property, 77
- groups,. See measure groups
- ID property, 76
- MAX aggregations, 79
- MeasureExpression property, 77
- MIN aggregations, 79
- Name property, 76
- properties of, 76
- Source property, 76
- SUM aggregations, 78
- Translations property, 76
- Visible property, 76

Measures dimension, 75**Measures property, 82****medium-latency MOLAP proactive caching scenario, 442****Member Creation, 54****Member. GetChildren() function, 619****Member. GetChildren(long start, long count function), 619****Member. GetChildren(long start, long count, params MemberFilter [] filters) function, 619****Member. GetChildren(long start, long count, string [] properties, params MemberFilter [] filters) function, 619****MemberCollection class**

- ADOMD.NET, 618-624
- GetMembers(long start, long count, string [] properties, params MemberFilter [] filters) overload method, 623-624
- Level.GetMembers() function, 618-620
- Level.GetMembers(long start, long count) function, 619-621

- Level.GetMembers(long start, long count, params MemberFilter [] filters) function, 619

- Level.GetMembers(long start, long count, string [] properties, params MemberFilter [] filters) function, 619-623

- Member. GetChildren() function, 619

- Member. GetChildren(long start, long count) function, 619

- Member. GetChildren(long start, long count, params MemberFilter [] filters) function, 619

- Member. GetChildren(long start, long count, string [] properties, params MemberFilter [] filters) function, 619

MemberNameUnique property, 46, 53**members, 251**

- attributes of, 18
- calculated members, 190-192
 - calculated measures, 193
 - CREATE_MEMBER statements, 193-196
 - creating, 194
 - defining, 193
 - DROP_MEMBER statements, 196
 - MDX scripts, 193
 - NON_EMPTY_BEHAVIOR property, 197-198
 - queries, 196
 - SELECT clauses, 196
 - WITH clauses, 193, 196
- keys (dimension attributes), 50-53
- Missing Member mode, 166
- names (dimension attributes), 53
- null members, 165
- properties, 161-162
- references (dimension hierarchies), 59
- values, 17, 142

MemberUniqueNameStyle parameter, 67**memory**

- allocators, 505, 511
 - levels of, 512
 - types of, 511
- cache system memory model, 509
- cleanup, 507
- distributing, 512

- economic memory management model, 504
 - Income parameter, 508
 - InitialBonus parameter, 508
 - MaximumBalance parameter, 508
 - MinimumBalance parameter, 508
 - parameters of, 508
 - Tax parameter, 508
- file stores, managing in, 510
- Memory Governor, 512-513
 - attribute processing, 515-516
 - building aggregations, 517-518
 - building indexes, 518-519
 - partition processing, 515-516
- memory holders, 504
 - allocators, 505, 511-512
 - shrinkable memory holders, 505
- Memory Manager
 - cache system memory model, 509
 - file store memory management, 510
 - memory cleanup, 507
 - memory holders, 504-505
 - subsystem memory management, 509-510
 - user sessions memory management, 510
- subsystem memory management
 - cache system memory model, 509
 - file stores, 510
 - user sessions, 510
- user sessions, 510
- MergePartitions command, 399**
- merging partitions, 399-400**
- metadata**
 - ADOMD.NET
 - caching metadata, 615-617
 - collection operations, 612-615
 - Count method, 612
 - Find method, 612
 - GetEnumerator method, 613
 - GetSchemaDataSet method, 625-630
 - handling metadata not in object form, 625-630
 - Item method, 612
 - iteration in, 613-614
 - MemberCollection class, 618-624
 - Properties collection, 615
 - retrieving dimension ordinals, 615
 - schema rowset requests, 614-615
 - DbStorageLocation property, 473
 - linked dimensions, 460
 - objects, 239, 251-255
 - Count option, 252
 - Find option, 255
 - GetEnumerator option, 255
 - Item option, 252
 - partitions, 374
 - query plans, creating, 556-558
- methods (MDX functions), 153**
- MidMemoryPrice (economic memory management model), 504**
- MimeType property, 51**
- MIN aggregations, 79**
- MinimumBalance parameter, 508**
- minor objects, 38, 672**
 - ID property, 674
 - named objects
 - Annotation property, 674
 - collections of, 674-676
 - Description property, 674
 - ID property, 674
 - IFormattable interface, 674
 - INamedComponent interface, 674
 - unnamed objects, 672-674
- Missing Member mode, 166**
- models, multidimensional**
 - cubes, 124-136
 - data sources, 110-111
 - DDL files, 112-113
 - DDVs, 114-116
 - dimensions, 118-123
 - modifying data sources, 111-112
- modifying**
 - cubes, 125-130
 - dimensions, 119-123
 - DSVs, 115-116

MOLAP (multidimensional online analytical processing), 11

- MOLAP to MOLAP transition mode (data latency), 437
- MOLAP-ROLAP-MOLAP transition mode (data latency), 437, 439
- partitions, 301, 390
- proactive caching, 440
 - automatic scenario, 441
 - low-latency scenario, 442
 - medium-latency scenario, 442
 - scheduled scenario, 440

money, currency conversion example (measure expressions), 105-107

monitoring

- commands, 818-819
- connections, 818-819
- DMV, 816
 - Discover_Commands, 818-819
 - Discover_Connections, 818-819
 - Discover_Object_Activity, 820
 - Discover_Sessions, 818-819
 - queries, 817
- perfmon counters, 821-822
- proactive caching events, 448
- SchemaRowsets, 816-817
- server state, 820-821
- sessions, 818-819
- SQL constructs, 817-818
- user logins, 715

Move method, 674

moving data, ETL process, 407-408

- direct-load ETL packages, 409
- SSIS, 408-414
- testing packages, 409

multidimension space, defining coordinates in, 141-142

Multidimensional Data Model (Analysis Services)

- aggregation functions, 24
- application data model, 9
- cells, 22
- conceptual data model, 9, 37
 - DDL, 37-43
 - multilanguage support, 39, 41

dimensions, 13

- attributes of, 13, 20, 44-45
- cardinality of, 18
- hierarchies, 21-22, 57-61
- members of, 17
- size of, 18

four-tier architectures, 28-31

measures, 13

one-tier architectures, 27-29

physical data model, 9

slices, 19

subcubes, 24

three-tier architectures, 27, 30

tuples, 19

two-tier architectures, 27-28

UDM, 11-12, 32

multidimensional models

cubes

- building perspectives, 130
- creating, 124-125
- defining translations, 131-132
- modifying, 125-130
- viewing, 133-136

data sources

- creating, 110-111
- modifying, 111-112

DDL files, 112-113

dimensions

- creating, 118-119
- modifying, 119-123

DSVs, 114-116

relational schemas, building, 334

Cube Wizard (BI Dev Studio), 336, 339

Schema Generation Wizard (BI Dev Studio), 337, 341

templates, 339, 341

multilanguage support, 39-41

N

Name parameter, cube dimensions, 67

Name property, 55

- Create command, 767
- major objects, 38, 670
- measures, 76
- minor objects, 672

NameColumn property, 46

named calculations, 117, 320-321

named minor objects, 672

Annotation property, 674

collections of

Add() method, 674

Add(string name) method, 675

Add(string name, string id) method, 675

Contains(string id) method, 675

ContainsName(string name) method, 675

Find(string id) method, 675

FindByName method, 675

GetByName(string name) method, 675

GetNewID method, 675

GetNewName method, 675

IndexOf(string id) method, 676

IndexOfName(string name) method, 676

IsValidID method, 676

IsValidName method, 676

Item method, 676

Remove method, 676

Description property, 674

ID property, 674

IFormattable interface, 674

INamedComponent interface, 674

named queries, 319, 321

named sets, 190, 209, 251

CREATE_SET statements, 210, 213

DROP_SET statements, 210

dynamic name sets, 213-214

static name sets, 210, 212

WHERE clauses, 213-214

WITH clauses, 209-210

native hierarchies, 87

natural hierarchies (dimensions), 57

nested Scope statements, 204, 206

NLB (Network Load Balancing), OLAP farms, 455

nonaggregatable attributes (aggregation), 427

None aggregation function, 228

nonprocessable major objects, 672

nonshrinkable memory holders, 505, 511

nontransactional Batch commands, 498

nonupdatable cells, 298-299

NON_EMPTY clauses, subcubes, 182

NON_EMPTY operators, 170-173

MDX query optimization, 541

NON_EMPTY_BEHAVIOR property, 197-198

Notification events, 770

notifications, proactive caching, 445, 449

client initiated notifications, 446

scheduled polling notification mechanism, 446

SQL Server, 446

NotifyTableChange requests, 446

NTLM, 715

NullKeyConvertedToUnknown property, 403

NullKeyNotAllowed property, 403

NullProcessing property, 52

nulls, 165

empty cells, 170-172

members, 165

Missing Member mode, 166

tuples, 165

O

ObjectAllocator memory allocators, 511

ObjectExpansion parameter, 485

objects

aggregation objects, 423-426

Annotation property, 38

Catalog object, 38

creating, 484

Database object, 38

deleting, 486

Description property, 38

editing, 484-485

Language property, 39

limitations of, 39

linked objects, 459

Locale property, 39

locking, 491

commit locks, 492-494

deadlocks, 491

major objects

- Cube objects, 39
- DDL, 38-39
- Dimension objects, 39
- ID property, 38-40
- Name property, 38

MDX

- by name, 158
- by qualified name, 159
- by unique name, 159-160
- type conversion rules, 173-174

metadata, 251

minor objects

- DDL, 38

permission objects

- CellPermission object, 730
- CubePermission object, 729
- DatabasePermission object, 728-729
- DataSourcePermission object, 729
- defining, 726
- DimensionPermission object, 729
- inheritance in, 727
- security, 723
- union in, 728

processing, 443, 486, 489

security, 721

server models, 252

symmetry, 644-647

Translation property, 40

offline data access, 573**OLAP (online analytical processing), 8**

ADOMD.NET, 652

farms, 453

- data storage, 453-455
- linked dimensions, 455-460
- linked object source updates, 457
- measure groups, 455-463
- NLB, 455
- queries, executing, 453
- remote partitions, 454
- writeback data support, 455

HOLAP connections, 11

MOLAP connections, 11

relational databases versus, 8

ROLAP connections, 11

OlapInfo object, 637-639**OlapInfo section (MDDataset-formatted results), 589****OLE DB**

- Binary XML, 571
- binary XML features, enabling, 410
- compression
 - data format, 571
 - features, enabling, 410
- data access, 576
- ETL packages, 410
- XML/A protocol, 410

OLEDB for OLAP, 29**OLTP (online transaction processing), 7**

- ETL process, 407-408

ON clause (SELECT statements), 140**one-tier architectures (Multidimensional Data Model), 27, 29****One-to-Many relationships (dimension attributes), 55-56****One-to-One relationships (dimension attributes), 55-56****OnlineMode property, 439****OperationException class, 706-707****OptimizedState parameter, 70****optional relationships (dimension attributes), 55-56****Optionality property, 55-56****Order function, 158****order stores, hierarchy data structures (dimensions), 362****OrderBy property, 46****OrderByAttributeID property, 46****ordering, rules of (DDL), 41****ordinals, 615****OutOfSyncException class, 708-709****overwrites (implicit), 176****P****PageAllocator memory allocators, 511****parallel execution blocks, syntax of, 501**

Parent method, AMO, 673**parent-child dimensions, processing, 389****parent-child relationships**

- DataMember property, 93
- defining, 92
- Employee attribute, 93-94
- Employee dimension, 94
- hierarchies in, 94-95
- levels in, 92
- Usage parameter, 92

ParentKpiID property, KPI object, 264**Parent_KPI property, CREATE KPI statements, 269****parsing**

- MDX queries, 530-531
- thread pools, 522

Partition Processing Destination editor, 414**partition-loading ETL packages**

- debugging, 415
- SSIS, creating in, 414
- testing, 415

Partition1 object, 437**partitions, 83**

- AggregationDesignID property, 364
- AggregationPrefix property, 364
- aggregations, building in, 393-395
- attributes, decoding, 369
- CreatedTimestamp property, 365
- cube processing
 - building aggregations, 397
 - building indexes, 397
 - incremental partition updates, 398-399
 - lazy processing, 397
 - loading data into partitions, 397
 - merging partitions, 399-400
- data, 368
- defining, 364-367
- ErrorConfiguration property, 365
- EstimatedRows property, 365
- EstimatedSize property, 365
- HOLAP data storage mode, 390
 - aggregations in, 397
 - indexes in, 397

indexes

- aggregation indexes, 373
- building, 370-371, 393
- LastProcessed property, 365
- LastSchemaUpdate property, 365
- loading data into (cube processing), 396
- measure groups, processing in, 392
- Memory Governor partition processing, 515-516
- merging, 399-400
- metadata files, 374
- MOLAP
 - data storage mode, 390
 - partitions, 301
- ProactiveCaching property, 365
- processing
 - activity, tracing, 779
 - Process Data jobs, 391-392
 - Read Data jobs, 391
 - Write Data jobs, 392-393
- ProcessingMode property, 365
- ProcessingPriority property, 365
- properties of, 364, 366
- query bindings, defining via, 326-327
- Query Partition jobs, 556-562
- remote partitions, 374-375, 464
 - creating, 464-467
 - mapping, 813
 - OLAP farms, 454
 - retrieving data from, 563
 - synchronizing, 812-814
- RemoteDataSourceID property, 365
- ROLAP
 - data storage mode, 390
 - partitions, 301
 - retrieving data from, 559, 562
- Segment jobs, 558, 562
- Slice property, 366
- slices, 368
- table bindings, defining via, 326
- updates, incremental updates, 398-399
- writeback partitions, 300
 - converting to regular partitions, 303
 - deleting, 303

Password property, 313-314

passwords

- ADOMD.NET connections, 606
- database attachments, 472

Path property, report actions, 276

perfmon counters, 821-822

performance

- KPI, 262
 - cube-based KPI, 262-268
 - data retrieval, 270
 - defining, 262
 - querying, 271
 - scorecards, 262
 - session-based KPI, 268-270
- KPIs, 251
- memory cleanup, 507
- relational databases, tuning in, 332

permanent writeback, 291

- lifetime of updates, 299-301

permissions

- CellPermission object, 730
- CubePermission object, 729
- DatabasePermission object, 728-729
- DataSourcePermission object, 729
- defining in, 726
- DimensionPermission object, 729
- inheritance in, 727
- object security, 721
- properties of, 726
- role objects, granting to, 721
- security, 723
- union in, 728

PermissionSet property, 248

Persistence property, 459

perspectives, cubes, 72

- building, 130
- defining for, 72-74

Perspectives parameter, 64

physical cell calculation plans (MDX queries)

- building, 546-547
- executing, 547

physical data model, 9

- cubes, 364
 - aggregation indexes, 373
 - building partition indexes, 370-371

- data structure overview, 375
- decoding partition attributes, 369
- defining partitions, 364, 367
- measure groups, 375
- partition data, 368
- partition metadata files, 374
- partition properties, 364-366
- partition slices, 368
- remote partitions, 374-375

data storage

- bit stores, 348
- compressed stores, 349-350
- data stores, 346
- file stores, 346, 348
- hash stores, 350
- string stores, 348

dimension data structures

- attribute data structures, 351-359
- attribute relationships, 355-356
- hierarchy data structures, 360-363

pipeline processing, 410

polling queries, 446

pooling connections, 314

private information, storing, 314

proactive caching, 12, 436

- considerations for, 448
- data latency, 437
- HOLAP real-time scenario, 442
- long-running MOLAP processing, 439
- MOLAP, 440

- automatic scenario, 441
- low-latency scenario, 442
- medium-latency scenario, 442
- scheduled scenario, 440

monitoring activity, 448

notifications, 445

- client initiated notifications, 446
- scheduled polling notification mechanism, 446

SQL Server, 446

object scheduling, processing, 443

ROLAP real-time scenario, 443

updates, 447-448

- scheduling, 443
- setting frequency of, 438

ProactiveCaching object

- defining, 437
- partitions, 365
- properties of, 439

Process command

- elements of, 488-489
- remote partitions, 466
- syntax of, 486
- types of, 489

process data jobs, 382-383

- partitions, processing, 391-392

Process method, 678**Process property, 727****process-execution thread pool, 523****processable major objects, 670, 678**

- CanProcess method, 678
- IProcessable interface, 678
- LastProcessed method, 678
- Process method, 678

ProcessAdd command, 423**ProcessAdd cube processing option, 396****ProcessAffectedObjects parameter, 499-501****ProcessClear cube processing option, 396****ProcessClearIndexes cube processing option, 396****ProcessData cube processing option, 396-397****ProcessFull command, 426****ProcessFull cube processing option, 396****ProcessIndex command, aggregations, 426****ProcessIndex cube processing option, 396****processing activity, tracing, 776**

- dimensions, 776-779
- partitions, 779
- queries, 780
 - complex queries, 782-784
 - simple queries, 780-782

ProcessingMode property, partitions, 365**ProcessingPriority property, partitions, 365****Product dimension, many-to-many dimensions, 102****Progress Report Begin event, 776-782****Progress Report Current event, 776****Progress Report End event, 776, 782****Progress Report Error event, 776****Progress Report events**

- ConnectionID property, 449
- proactive caching, 449

progress reports, 400**Progress_Report event, 770****Progress_Report_Current event, 769****projects**

- cubes, 133-136
- deploying, 135

properties (MDX functions), 153**Properties collection, ADOMD.NET, 615****Properties parameter**

- Discover method, 584-587
- Execute method, 588-593

property stores, attribute data structures (dimensions), 353-354**publisher servers, 32**

Q

qualified name, referencing objects by, 159**qualified names, stored procedures, 246****queries**

- aggregations, 555
 - measure groups, 559
 - monitoring usage of, 433-434
- asynchronously execution/cancellation of, 659-662
- bindings, 326-327
- bitmap indexes, 555, 559
- calculated members, 196
- calculation scopes, 532
 - global scopes, 535-536, 538
 - session scopes, 536, 538
- custom properties in, 227
- data decoder, 555, 559
- Data jobs, 565
- Datacache objects, 555, 563-565
- datasets, populating with query results, 647
- defining, 139
- DMV, 817
- DRILLTHROUGH statements, 285-287
- execution process, 528, 554-555

- Exists function, 183
- iterating results in ADOMD.NET, 650-652
- KPI data, 271
- Lookup jobs, 565
- managing, 576
- many-to-many relationships, 104
- MapQuery, 555, 559, 563
- MDX, 139
 - cell calculation in, 147-148
 - defining axes, 141
 - defining coordinates in multidimensional space, 142
 - expressions, 153
 - filtering sets, 155-157
 - functions, 153-158
 - listing axes, 140
 - naming axes, 141
 - object references, 158-160
 - query execution context, 147-148
 - SELECT statements, 140-147
 - set algebra, 149-152
 - slicer axis, 144
- MDX queries
 - cell calculation plans, 542-547
 - cell properties in, 162-164
 - CurrentMember function, 185
 - DIMENSION_PROPERTIES clause, 162
 - executing via ADOMD.NET, 632
 - execution process, 528
 - Filter function, 171
 - member properties in, 161-162
 - optimizing logical cube space in, 541
 - parsing, 530-531
 - SELECT statements, 162
 - virtual set operation trees, 538, 540
- measure groups, 556-558
 - DISTINCT_COUNT measures, 560-563
 - linked measure groups, 563
 - measure groups with indirect dimensions, 564-566
- named queries, 319-321
- OLAP farms, 453
- parsing, 530-531
- polling queries, 446
- processing activity, tracing, 780
 - complex queries, 782-784
 - simple queries, 780-782
- Query Analyzer, 565
- query-execution thread pool, 522
- query logs, 428
 - columns in, 429
 - configuring, 428-430
 - server properties, 428
- Query Partition jobs, 555-562
- query plans, measure groups, 556-558
- Query Range Group jobs, 562
- remote partitions, 563
- ROLAP partitions, 559, 562
- server-state discover requests, 776
- SQL, 139
- trace files, capturing in, 332
- Usage-Based Optimization Wizard, 428
- Query Analyzer, 565**
- Query Begin event, 780**
- Query Cube Begin event, 782**
- Query Cube Begin Event event, 780**
- Query Cube event, 782**
- Query Dimension event, 783**
- Query Partition jobs, 555-562**
- Query Range Group jobs, 562**
- Query Subcube event, 781**
- Query Subcube Verbose event, 781**
- QueryBinding object, 321**
- queues, thread pools, 523-524**

R

- read data jobs, 381**
- Read Data jobs, 391**
- Read property, 727**
- read-only databases, 473**
- read/write permissions, 719**
- ReadDefinition property, 727**
- ReadSourceData property, 729**
- real-time**
 - HOLAP proactive caching scenario, 442
 - ROLAP proactive caching scenario, 443

recursion resolution, 218, 220

referenced dimensions

defining, 100-102

Geography dimension, 98-99

Materialization property, 99

referenced objects (AMO), 678

Refresh method, 677

Register Database Assembly dialog, 243

regular expressions, 256

related attributes, 176

related dimension attributes, 47-56

flexible relationships, 55-56

mandatory relationships, 55-56

One-to-Many relationships, 55-56

One-to-One relationships, 55-56

optional relationships, 55-56

rigid relationships, 54-56

relating attributes, 176

relational databases

data warehouses, 329-331

indexes in, 331-332

loading data from, 310

Binding object, 307-308

DataSource object, 307-311

DataSourceView object, 307

DSV (Analysis Services), 308

multidimensional models, building from, 334

Cube Wizard (BI Dev Studio), 336, 339

Schema Generation Wizard (BI Dev Studio), 337-341

templates, 339-341

OLAP versus, 8

optimizing, 331-334

performance, tuning, 332

schemas, 329

security, 313

relational reporting-style dimensions, 420-422

Relationship object, properties of, 54

relationships (strong), 176

RelationshipType property, 54-56

remote partitions, 32, 374-375, 464

creating, 464-467

mapping, 813

OLAP farms, 454

retrieving data from, 563

synchronizing, 812-814

remote servers, 32

RemoteDataSourceID property, 365

Remove method

named object collections, 676

unnamed object collections, 674

RemoveAt method, 674

ReportFormatParameters property, 276

ReportParameters property, 277

reports

progress reports, 400

properties of, 276

ReportServer property, 277

Request level (memory allocators), 512

requests, 241-242

RequestType parameter (Discover method), 583

RequireClientAuthentication property, 715

resource monitoring

commands, 818-819

connections, 818-819

DMV, 816

Discover_Commands, 818-819

Discover_Connections, 818-819

Discover_Object_Activity, 820

Discover_Sessions, 818-819

queries, 817

perfmon counters, 821-822

SchemaRowsets, 816-817

server state, 820-821

sessions, 818-819

SQL constructs, 817-818

ResponseFormatException class, 707

Restore command, 811

Restrictions parameter (Discover method), 583

retrieving data

Data jobs, 565

KPI data, 270

Lookup jobs, 565

managing, 576

measure groups, 556, 558

DISTINCT_COUNT measures, 560-563

- linked measure groups, 563
- measure groups with indirect dimensions, 564-566
- query execution process, 554-555
- remote partitions, 563
- ROLAP partitions, 559-562
- rigid aggregations, 422-423**
- rigid relationships (dimension attributes), 54, 56**
- ROLAP (relational online analytical processing)**
 - partitions, 11, 301**
 - aggregations, 440
 - dimension processing, 388
 - partitions, 390
 - proactive caching, 443
 - retrieving data from partitions, 559-562
- role-based security, 248**
- role-playing cube dimensions, 70-71**
- RoleID property, 727**
- roles**
 - database roles, 730
 - permissions, granting, 721
 - security, 721
- RollbackTransaction command, 489-490**
- Root function, assignments, 206**
- Row bindings, 323**
- RowBinding object, 321-323**
- rules**
 - of inheritance (DDL), 42
 - of ordering (DDL), 41

S

- scalability, 451**
 - Attach command, 472
 - DbStorageLocation property, 473
 - Detach command, 470-471
 - linked dimensions, 455-460, 467
 - measure groups, 455-463
 - OLAP farms, 453
 - data storage, 453-455
 - linked dimensions, 455-460
 - linked object source updates, 457
 - measure groups, 455-463
 - NLB, 455
 - query execution in, 453
 - remote partitions, 454
 - writeback data support, 455
 - read-only databases, 473
 - remote partitions, 464-467
 - scale-out approach, 452
 - scale-up approach, 451-452
 - shared scalable databases, 470
- scalars, 153**
- scheduling**
 - MOLAP proactive caching scenario, 440
 - object processing, 443
- SchemaRowsets, 816-817**
- schemas**
 - dimension processing schemas, 385
- Scope objects**
 - calculation scopes, 532
 - global scopes, 535-538
 - session scopes, 536-538
- Scope parameters**
 - Alter command, 485
 - Create command, 484
- Scope statements**
 - assignments, 203-206
 - dynamic cell security, 759
 - nested Scope statements, 204-206
- scorecards, 262**
- Scripter object, 694**
- scripts, MDX**
 - calculated members, 193
 - creating, 191
- security**
 - administrative security, 713
 - ADOMD.NET connections, 606
 - BudgetCubeUsers role, 724, 728
 - cell security, 731-733
 - CellPermission object, 751-752
 - contingent cell security, 756
 - defining, 751-754
 - dynamic cell security, 758-760
 - MDX scripts, 748, 760
 - testing, 754

- code access security, 714
- connection security, 713-714
 - anonymous access, 716
 - authentication, 715-717
 - constrained delegation, 716
 - DISCOVER_CONNECTIONS requests, 715
 - HTTP, 715-717
 - TCP/IP, 714-715
- CubePermission objects, 726
- data security, 713
- data source objects, 312-314
- Database role, 723
- database role management, 730
- dimension security, 731-732
 - architecture of, 748
 - AttributePermission object, 735-737, 740, 747, 750
 - defining, 734
 - defining via user interface, 742-744
 - dynamic dimension security, 746-747
 - MDX scripts, 748
 - testing, 744
- external data access security, 714
 - changing service logon accounts, 720
 - choosing service logon accounts, 718
 - configuring data source access, 719-720
 - failover cluster operations, 721
 - named instances (SQL Server Browser), 721
- models, 248
 - code access, 248
 - role-based, 248
 - user-based, 249-251
- objects, 721
- Permission objects, 723
 - CellPermission object, 730
 - CubePermission object, 729
 - DatabasePermission object, 728-729
 - DataSourcePermission object, 729
 - defining, 726
 - DimensionPermission object, 729
 - inheritance in, 727
 - properties of, 726
 - union in, 728
 - relational databases, 313
 - server administrators, 722-723
- Segment jobs, 558, 562**
- SELECT statements, 140**
 - calculated members, 196
 - CELL_PROPERTIES clause, 163-164
 - default members, 144-146
 - DIMENSION_PROPERTIES clause, 163
 - FROM clause, 140
 - MDX queries, 162
 - multidimensional space, defining
 - coordinates in, 141-142
 - ON clause, 140
 - SELECT clause, 140
 - subcubes, 182
 - WHERE clause, 140, 146-148, 156-157
- semi-additive measures, 190, 227-229**
 - AverageOfChildren functions, 229
 - ByAccount functions, 229-232
 - FirstChild functions, 228
 - FirstNonEmpty functions, 228
 - LastChild functions, 228
 - LastNonEmpty functions, 228
 - None functions, 228
- server-state discover requests, 776**
- Server.CancelCommand method, 688, 692**
- servers**
 - ADOMD.NET connections, 608-610
 - managing, 482
 - master servers, 32
 - object models, 251
 - MDX objects, 255-257
 - metadata, 252, 255
 - publisher servers, 32
 - query log properties, 428
 - remote servers, 32
 - security, 722-723
 - state, 820-821
 - subscriber servers, 32
 - user logins, 715

service logon accounts

- choosing, 718-720
- read/write permissions, 719

Session level (memory allocators), 512**Session Manager, 581-583****Session objects, 580-581****session-based KPI (Key Performance Indicators)**

- CREATE KPI statements, 268-270
- DROP KPI statements, 268

sessions, 765

- ADOMD.NET connections
 - creating in, 606
 - multiple connections using one session, 607-608
- calculation scopes, 532
 - caches, 536
 - lifetimes of, 536-538
- deleting, 494
- managing, 481-482
- monitoring, 818-819

sets

- algebra, 142, 149
 - CrossJoin sets, 151-152
 - Except sets, 150
 - Extract sets, 152
 - Intersect sets, 150
 - Union sets, 149
- arbitrary sets, 179
- defining, 142
- empty sets, 165
- hierarchy data structures (dimensions), 361-362
- filtering, 155-157
- named sets, 190, 209, 251
 - CREATE_SET statements, 210, 213
 - DROP_SET statements, 210
 - dynamic name sets, 213-214
 - static name sets, 210, 212
 - WHERE clauses, 213-214
 - WITH clauses, 209-210
- stored procedures, 256
- WHERE clause, 177-179

shared scalable databases, 470**short-command parsing thread pool, 522****short-running requests (parsing), 522****shrinkable memory holders, 505****SilenceInterval property, 439****SilenceOverrideInterval property, 439****simple keys (dimension attributes), 50****Slice property, partitions, 366****slicer axis, 144-146****slices, 19**

- defining, 19
- partitions, 368

SnapshotDefinitionFile property, 775**SnapshotFrequency property, 775****snowflake schemas, 330-331****SOAP (Simple Object Access Protocol)**

- data access, 569-570
- XML/A, 579-581

SOLVE_ORDER property, 216**Source objects, 458, 463****Source property**

- measures, 76
- partitions, 366

SourceAttributeID property, 57**SQL (Structured Query Language), 139**

- SQL constructs, 817-818
- SQL Server
 - proactive caching notifications, 446
 - SSIS, 408-415
- SQL Server Browser, 721
- SQL Server Profiler
 - trace, 768-772
- Trace Properties dialog, 768-769

SSIS (SQL Server Integration Services)

- data flow component, 410-414
- ETL packages
 - building, 408
 - data processing, 410
 - dimension-loading packages, 410-413
 - direct-load ETL packages, 409
 - partition-loading packages, 414-415

star schemas, 330**State property, 366****static inheritance, 42****static name sets, 210-212**

Status property, 263-265**Status_Graphic property**

CREATE KPI statements, 269

KPI object, 263

storage

bit stores, 348

callbacks in procedures, 257-260

compressed stores, 349-350

data

OLAP farms, 453-455

stores, 346

distributed storage

linked objects, 32

remote partitions, 32

thick clients, 33

thin clients, 33

file stores, 346-348

hash stores, 350

MDX

calling, 246-247

configuring CLR assemblies, 239-242

creating, 239

implementing COM assemblies, 245

sending CLR assemblies, 244

partitions, 390

private information, 314

retrieving, 554-555

sets, filtering, 256

string stores, 348

StorageLocation property, partitions, 366**StorageMode property, partitions, 366****string stores, 348****StringAllocator memory allocators, 511****strong relationships, 176****structure stores, 360****subcubes, 24, 180, 183-184**

assigning values to, 202

assignments, 198-201

calculation scopes, 534

CREATE_SUBCUBE statements, 180-184, 200

defining, 199-200

DROP_SUBCUBE statements, 180

FROM clauses, 182

MDX query execution context, 147

NON_EMPTY clauses, 182

SELECT statements, 182

WITH clauses, 182

subscriber servers, 32**SubSelects, 180-184, 187****sub_cube_expression, 199-200****SUM aggregations, measures, 78****symmetry (objects), 644-647****Synchronization command, OLAP farms, 455****Synchronize command**

defining, 810

deploying databases via, 805-808

failover clusters, 814

options for, 810

remote partitions, 812

Restore command's similarities to, 811

Synchronize Database Wizard, 809, 813**synchronizing**

databases

DDL command, 809-811

Synchronize command, 805-808

Synchronize Database Wizard, 809

failover clusters, 814

remote partitions, 812-814

synchronizing transactions via commit locks, 492-494**System.ComponentModel.Component class, 672****T****TableBinding object, 321-326****tables, 362-363****tabular bindings, 324, 326****TargetType property, 274****Tax parameter, 508****TCP (Transmission Control Protocol) data access, 569****TCP/IP (Transmission Control Protocol/Internet Protocol)**

connection security, 714-715

data access via, 569-571

templates, 339-341**temporary writeback, 291-292**

- lifetime of updates, 301
- updates, lifetime of, 299

testing

- cell security, 754
- dimension security, 744
- ETL packages, 409
 - dimension-loading packages, 413
 - partition-loading packages, 415

TextData column (Events Selection Tab), 769**thick clients, 33****thin clients, 33****Thread level (memory allocators), 512****threads**

- managing via different subsystems, 525-526
- Thread Management subsystem, 521-525
- thread pools
 - architecture of, 523-525
 - long-command thread pool, 522
 - parsing, 522
 - process-execution thread pool, 523
 - query-execution thread pool, 522
 - queues in, 523-524
 - short-command thread pool, 522

three-tier architectures (Multidimensional Data Model), 27, 30**Time dimension, 103****Timeout property, 311****timeouts (connections), 314****TotalMemoryLimit (economic memory management model), 504, 507****trace, 763**

- administrative trace, 765
- architecture of, 764
- DDL, creating via, 766-767
- events, 770-772
- flight recorder trace, 765, 773
 - configuring, 775-776
 - properties of, 775
- processing activity, 776
 - dimensions, 776-779
 - partitions, 779
 - queries, 780-784

session trace, 765

SQL Server notifications, 446

SQL Server Profiler, 768

defining, 768-770

running, 770-772

Trace objects, 764

Trace Properties dialog (SQL Server Profiler), 768-769**TraceDefinitionFile property, 775****traces**

AMO, 697-705

files, 332

Transaction parameter, Batch command, 497**transaction synchronization via commit locks, 492-494****transactional commands, 489-490****Translation property, 40, 47, 55****translations, 123**

defining, 131-132

Translations collection, cubes, 67**Translations property**

- dimension attribute member names, 54
- measures, 76

Translations tab (Dimension Designer), 123**tree of dimension attributes, 48-50****Trend property, 263-265****TrendGraphic property, 264****Trend_Graphic property, CREATE KPI statements, 269****Trimming property, 52****troubleshooting**

ADOMD.NET, 662-663

AdomdCacheExpiredException class, 666-667

AdomdConnectionException class, 666

AdomdErrorResponseException class, 663-665

AdomdUnknownResponseException class, 666

deadlocks, 491

fragmentation, 511

XML/A, 593

cell calculation errors, 597

errors occurring after start of response serialization, 596

- MDX errors, 595-596
 - warnings, 598
 - whole method failure errors, 594
- Tuning Advisor (Database Engine), 331, 334**
- TupleContainer interface, 540**
- TupleCounter interface, 541**
- TupleIterator interface, 540**
- TupleRanker interface, 541**
- tuples**
 - auto-exist tuples, 167-168
 - current coordinates, 174
 - defining, 19
 - dimensionality, 142
 - empty tuples, removing, 541
 - existing tuples, 167-168
 - member values, 142
 - multidimensional space, defining coordinates in, 142
 - nonexisting tuples, 167-168
 - null tuples, 165
 - sets, 142
 - wildcards, 19
- two-tier architectures (Multidimensional Data Model), 27-28**
- type conversion rules, 173-174**
- Type property, 46**
 - actions, defining function of, 274-275
 - measure groups, 82
 - partitions, 366

U

- UDM (unified dimensional model), 11-12, 32**
- unary member stores, 355**
- unary operators, 221-224**
- UnaryOperatorColumn property, 222**
- union, permission objects, 728**
- Union sets, 149**
- unique name, referencing objects by, 159-160**
- Unknown Members (dimension attributes), 51**
- unlock command, 493**
- unnamed minor objects, 672**
 - collections of, 673
 - Add method, 673
 - CanAdd method, 673
 - Clear method, 673
 - Contains method, 673
 - Count method, 673
 - IndexOf method, 673
 - Insert method, 674
 - Item method, 674
 - Item property, 674
 - Move method, 674
 - Remove method, 674
 - RemoveAt method, 674
 - System.ComponentModel.Component class, 672
- unnatural hierarchies**
 - dimensions, 58
 - measure groups, 87
- Update Isolation Level property, 298**
- Update method, 677, 683**
- UpdateMode parameter, 677**
- updates**
 - cells, 298-299
 - dimension processing, 387-388
 - full updates, 447-448
 - incremental updates, 447-448
 - linked object sources, 457
 - order of updates, UPDATE_CUBE statements, 297-298
 - partitions, incremental partitions, 398-399
 - permanent writeback, 300-301
 - proactive caching, 438, 447-448
 - scheduling, 443
 - temporary writeback, 299-301
- UPDATE_CUBE statements, 293-304**
 - order of updates in, 297-298
 - syntax of, 292, 296
- Usage parameter, 92**
- Usage property, 46**
- Usage-Based Optimization Wizard, 428**
- UserName function, 746**
- users**
 - hierarchies, 141
 - interfaced, 742-744

- security, 249-251
- server logins, 715
- sessions, 510

- USE_EQUAL_ALLOCATION** method, 295
- USE_EQUAL_INCREMENT** method, 296
- USE_WEIGHTED_ALLOCATION** method, 296
- USE_WEIGHTED_INCREMENT** method, 296

V

- Validate** method, 673
- Value** property, 263-264
- ValueColumn** property, 46
- values**, 17, 202
- VectorAllocator** memory allocators, 511
- viewing cubes**, 133-136
- views**
 - creating, 114-115
 - modifying, 115-116
- virtual cubes**, creating, 468
- virtual set operation trees**, 538-540
- Visibility** property, 55
- Visible** parameters
 - cube dimensions, 70
 - cubes, 63
- Visible** property, measures, 76
- Visual Basic**, OLE DB data access, 576
- Visual Studio**
 - ADOMD.NET, OLAP data operations, 652
 - AMO in, 685
 - asynchronous cube processing, 689-692
 - canceling long-running operations, 688-692
 - disconnected mode, 693-694
 - error handling, 706-709
 - object loading, 692-693
 - registration, 685
 - Scripter object, 694-696
 - server connections, 685-686
 - sharing ADOMD.NET sessions, 686-688
 - traces, 697-705
 - OLAP data operations, ADOMD.NET, 652

- visual totals**, 186-187
- VisualTotals** property, 735, 740

W

- Warehouse** dimensions, 103
- warehouses** (data), 11
 - data's life cycle in, 407
 - ETL process, 407-408
 - relational database schemas, 329-331
- warnings**, handling in XML/A, 598
- Weight** property, 264
- what-if** analysis, 291
- WHERE** clauses, 140, 146-148
 - filtering sets, 156-157
 - named sets, 213-214
 - sets, 177-179
- wildcards**, tuples, 19
- Windows NT Challenge/Response** authentication, 715
- WITH** clauses
 - calculated members, 193, 196
 - named sets, 209-210
 - subcubes, 182
- WITH_CELL_CALCULATION** clauses, 199
- wizards**
 - Account Time Intelligence Wizard, 231
 - Aggregation Design Algorithm Wizard, 426
 - Cube Wizard, 124, 336, 339
 - Data Source View Wizard, 114
 - Data Source Wizard, 110-111
 - Deployment Wizard, 805-807
 - Dimension Wizard, 118-119
 - Linked Object Wizard, 127, 468
 - Schema Generation Wizard (BI Dev Studio), 337, 341
 - Synchronize Database Wizard, 809, 813
 - Usage-Based Optimization Wizard, 428
- write data jobs**, 383
- Write Data** jobs, 392-393
- Write** property, 727
- write/read** permissions, 719

writeback

- enabling, 301-303
- IRowsetChange OLE DB interface, 304
- permanent writeback, 291, 299-301
- temporary writeback, 291-292, 299-301
- writeback partitions, 300
 - converting to regular partitions, 303
 - deleting, 303
- writeback tables, 301

X**XML (Extensible Markup Language)**

- Binary XML, 570
- binary XML features, 410
- DDL and, 37
- measures, 78

XML/A (XML for Analysis), 579

- action discover requests, 280-283
- data access, 570, 574-575
- DataSource objects, 580
- Discover method
 - Properties parameter, 584-587
 - RequestType parameter, 583
 - Restrictions parameter, 583
 - signature of, 583
- error handling, 593
 - cell calculation errors, 597
 - errors occurring after start of response serialization, 596
 - MDX errors, 595-596
 - whole method failure errors, 594
- Execute method, 587
 - Command parameter, 588
 - Properties parameter, 588-593
- NotifyTableChange requests, 446
- OLE DB provider, 410
- protocol, 28-29
- Session Manager, 581-583
- Session objects, 580-581
- state management, 580-581
- warnings, handling, 598

XmlaWarningCollection collections, 677

This page intentionally left blank

Try Safari Books Online FREE

Get online access to 5,000+ Books and Videos



Safari
Books Online

FREE TRIAL—GET STARTED TODAY!
www.informit.com/safaritrial



Find trusted answers, fast

Only Safari lets you search across thousands of best-selling books from the top technology publishers, including Addison-Wesley Professional, Cisco Press, O'Reilly, Prentice Hall, Que, and Sams.



Master the latest tools and techniques

In addition to gaining access to an incredible inventory of technical books, Safari's extensive collection of video tutorials lets you learn from the leading video training experts.

WAIT, THERE'S MORE!



Keep your competitive edge

With Rough Cuts, get access to the developing manuscript and be among the first to learn the newest technologies.



Stay current with emerging technologies

Short Cuts and Quick Reference Sheets are short, concise, focused content created to get you up-to-speed quickly on new and cutting-edge technologies.

Addison
Wesley

Adobe Press

ALPHA

Cisco Press

FT Press
FINANCIAL TIMES

IBM
Press

lynda.com

Microsoft
Press

New
Riders

O'REILLY

Peachpit
Press

PRENTICE
HALL

QUE

Redbooks

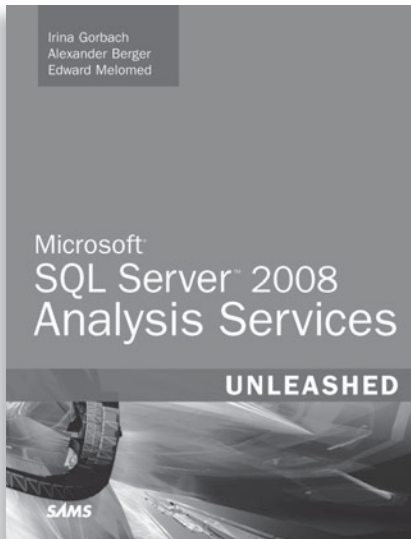
SAMS

SAS
Publishing

Sun
microsystems

Wiley
Publishing

WILEY



FREE Online Edition

Your purchase of **Microsoft SQL Server 2008 Analysis Services Unleashed** includes access to a free online edition for 45 days through the Safari Books Online subscription service. Nearly every Sams book is available online through Safari Books Online, along with more than 5,000 other technical books and videos from publishers such as Addison-Wesley Professional, Cisco Press, Exam Cram, IBM Press, O'Reilly, Prentice Hall, and Que.

SAFARI BOOKS ONLINE allows you to search for a specific answer, cut and paste code, download chapters, and stay current with emerging technologies.

Activate your FREE Online Edition at
www.informit.com/safarifree



STEP 1: Enter the coupon code: APIQYCB.



STEP 2: New Safari users, complete the brief registration form.
Safari subscribers, just log in.

If you have difficulty registering on Safari or accessing the online edition, please e-mail customer-service@safaribooksonline.com

Safari 
Books Online



Adobe Press



Cisco Press



IBM Press



Microsoft Press



O'REILLY



que



SAMS

