



E-Commerce Business Data Analysis

Topics on this Page

- ▼ [Introduction](#)
- ▼ [Microsoft Data Warehousing Framework](#)
- ▼ [OLAP Cube Architecture](#)
- ▼ [Analyzing Transactional Data](#)
- ▼ [Site Server Usage Analysis Plus Toolkit](#)
- ▼ [SQL Server 2000 and Analysis Services](#)
- ▼ [References](#)

Microsoft Enterprise Services White Paper E-Commerce Technical Readiness

Note This white paper is one of a series of papers about applying Microsoft® Enterprise Services frameworks to e-commerce solutions. [E-Commerce White Paper Series](#) contains a complete list, including descriptions, of all the articles in this series.

Introduction ▲

Purpose

E-commerce organizations must gain competitive advantage by having access to the most current and accurate information available to help them make better decisions about products, customers, partners, and processes. In the customer-focused world of e-commerce on the Web, it is becoming increasingly important to have not only the right information, but also superior analysis tools to enable organizations to improve overall decision-making.

Building and maintaining systems that can consistently deliver timely analytical information has historically been extremely challenging. These complex systems were typically difficult to implement, expensive to maintain, and required specialized software and hardware systems.

Microsoft is making data warehousing and business intelligence easier and more accessible by simplifying and integrating services into the platform itself—Microsoft® SQL Server™ and Microsoft Office—and providing open interfaces to access and share data in a heterogeneous environment.

The purpose of this paper is to introduce data warehousing concepts to IT managers and database administrators, show how these business intelligence concepts are being used today at e-commerce sites, and to preview important data warehousing features that will be included with Microsoft's newest database management system, SQL Server 2000.

Overview

This paper attempts to pull together information about various Microsoft data warehousing concepts and tools to provide value to e-commerce companies that are evaluating various approaches and strategies.

The first section provides a summarized review of the Microsoft Data Warehousing Framework. The concepts discussed in this section are generic and are applicable to a variety of business intelligence scenarios.

The second section is a short introduction to the online analytical processing (OLAP) cube architecture. Brief explanations of dimensions, levels, measures, and cubes are provided to allow a more complete understanding of the following section.

Next, OLAP Manager is explained at a high level by walking through a typical e-commerce scenario. A sample data mart is exploited to help clarify the process of designing and implementing functional OLAP

cubes. This section focuses on building data marts with transactional data—information that resides in the online transaction processing (OLTP) system.

The Site Server Usage Analysis Plus Toolkit is the topic of the next section. This downloadable toolkit enables developers to build more scalable solutions for analyzing Web server log data with OLAP cubes.

The final section previews significant new features in SQL Server 2000 that specifically address highly scalable data warehousing scenarios.

Microsoft Data Warehousing Framework

Introduction

The Microsoft Data Warehousing Framework was created to help simplify the design, implementation, and management of data warehousing solutions. This framework has been designed to provide:

- ✦ Open architecture that is easily integrated with and extended by third-party vendors.
- ✦ Heterogeneous data import, export, validation, and cleansing services with optional data lineage.
- ✦ Integrated metadata for data warehouse design, data extraction/transformation, server management, and end-user analysis tools.
- ✦ Core management services for scheduling, storage management, performance monitoring, alerts/events, and notification.

Architecture Terminology

Data warehousing architecture exists in two basic types: enterprise data warehouses and data marts.

- ✦ Enterprise data warehouses contain enterprise-wide information integrated from multiple operational data sources for consolidated data analysis.
- ✦ Data marts contain a subset of enterprise-wide data that is built for use by an individual department or division in an organization. The information in the data mart can be a subset of an enterprise data warehouse or can come directly from the operational data sources.

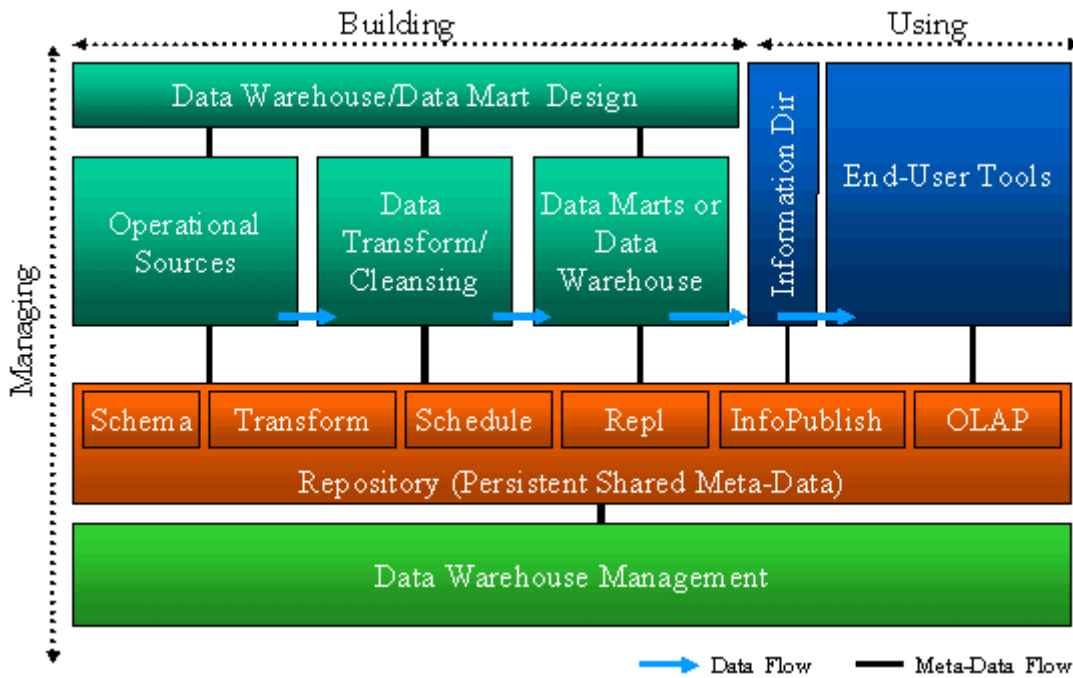
Regardless of size and scope, data warehouses and data marts are constructed and maintained through similar iterative processes.

Data Warehousing Components

A data warehouse consists of a number of components, including:

- ✦ Data sources for production information.
- ✦ Tools for designing and developing data stores and application components.
- ✦ Data extraction, transformation, and loading tools.
- ✦ Database management systems (DBMS).
- ✦ Data access, analysis, and visualization tools.
- ✦ System management tools.

The Microsoft Data Warehousing Framework provides a set of technologies that integrates these components. These core technologies are briefly described in the following sections.



If your browser does not support inline frames, [click here](#) to view on a separate page.

OLE DB

OLE DB is a system-level programming interface that manages data across the organization and provides an open standard for accessing all data types. OLE DB is designed to access both relational and nonrelational information sources.

OLE DB defines a collection of COM interfaces that encapsulates various database management system services. These interfaces enable the creation of software components that implement such services.

Microsoft Repository

The Microsoft Repository is a database that stores descriptive information about software components and their relationships. The items in the repository are available to repository tools through a set of published COM interfaces and information models that define database schema and data transformations through shared metadata. This repository makes it possible for tool vendors to build tools that interact with databases and software components without advance knowledge of the databases or software components.

Data Warehouse Tools

OLAP Manager is included with SQL Server OLAP Services. It is the management utility for OLAP Services and includes the Cube Browser, which is a tool used to analyze the data in the warehouse. Images of OLAP Manager are included in a later section of this paper.

OLAP Manager is a Microsoft Management Console (MMC) application that provides a way to access OLAP Servers and the metadata repositories that define multidimensional database structures.

OLAP Manager is used to:

- ✦ Specify databases and data sources.
- ✦ Build and process cubes.
- ✦ Specify storage options and optimize query performance.
- ✦ Manage server security.
- ✦ Browse cube data.

Replication

Database replication is used extensively in many data warehousing scenarios to move operational data to analysis staging servers.

With replication, recent copies of data are duplicated and distributed from a source database to a destination database, usually on a separate server. Databases participating in replication can be located on a large server servicing hundreds of users but can also be located on a single user's computer, making replication useful for a wide variety of applications.

Replication is typically used for, but not limited to, data that is duplicated from one database to another. For example, replication might be used for a product catalog that is maintained at a central office and replicated to branch offices.

Extracting, Transforming, and Loading (ETL)

Data warehouses centralize data to improve corporate decision making. However, the source data from various operational systems is often in a large variety of formats stored on a number of different databases. By using Data Transformation Services (DTS), you can import, export, and transform data among multiple homogeneous or heterogeneous sources and destinations using an OLE DB-based architecture.

DTS allows the following operations:

- ✦ Copy table schemas and data between database management systems (DBMSs).
- ✦ Create custom transformation objects that can be integrated into third-party products.
- ✦ Build data warehouses and data marts in SQL Server by importing and transferring data from multiple heterogeneous sources interactively or automatically on a regularly scheduled basis.
- ✦ Access applications using third-party OLE DB providers, which allows you to use applications for which an OLE DB provider exists as sources and destinations of data.

Client-Side Processing

Microsoft PivotTable® Service is a client-side component that allows client access to multidimensional data. PivotTable Service makes calls to the multidimensional data on the OLAP Server and presents the data to the client. PivotTable Service can also cache multidimensional data (in memory and on disk) and allow clients to browse the data locally.

Client applications can access the data provided by PivotTable Service through OLE DB for OLAP or through an application that uses Microsoft ActiveX® Data Objects (Multidimensional), or ADO MD, to access OLE DB. Each client interface allows you to:

- ✦ Connect to an OLAP server.
- ✦ Browse multidimensional schema.
- ✦ Query a cube and retrieve the results.

OLAP Cube Architecture

OLAP Cubes

The primary OLAP object is the cube, a multidimensional representation of detail and summary data. A cube consists of a data source, dimensions, measures, and partitions. You design cubes based on the analytical requirements of users. A data warehouse can support many different cubes, such as a sales cube, an inventory cube, and so on.

A cube's data source identifies and connects to the database containing the data warehouse data that is the source of data for the cube.

Dimensions

Dimensions map data warehouse dimension table information into a hierarchy of levels, such as a geography dimension with levels of continent, country, state/province, and city. Dimensions can be independently created and shared among cubes for ease of cube construction and to ensure consistency

of analysis data summarization. For example, if a shared dimension is used for a product hierarchy in all appropriate cubes, the organization of summarized product information will be consistent among the cubes that use the dimension.

Virtual Dimensions

A virtual dimension is a special type of dimension that maps the properties of members of another dimension into a dimension that can then be used in cubes. For example, a virtual dimension of a product's size property enables a cube to summarize data such as sale quantity by product by size, or such as the quantity of shirts sold by style by size. Virtual dimensions and member properties are evaluated as necessary for queries, and they require no physical cube storage.

Measures

Measures identify the numerical values from the fact table that are summarized for analysis, such as price, cost, or quantity sold.

Partitions

Partitions are the multidimensional storage containers that hold cube data. Each cube contains at least one partition, and a cube's data can be combined from multiple partitions. Each partition can take its data from a different data source and can be stored in a separate location. A partition's data can be updated independently of other partitions in a cube. For example, a cube's data can be divided by time, with a partition for the current year's data, another partition for the previous year's data, and a third partition for all data prior to the previous year.

A cube's partitions can be independently stored in different storage modes with different degrees of summarization. Partitions are invisible to the user, to whom the cube appears to be a single object, yet they provide the administrator with a wide variety of options to manage the underlying OLAP data.

Virtual Cubes

A virtual cube is a logical view of portions of one or more cubes. A virtual cube can be used to join relatively dissimilar cubes that share a common dimension, such as a sales cube and a warehouse cube, for special analysis purposes while retaining the separate cubes for simplicity. Dimensions and measures can be selected from the joined cubes to be presented in the virtual cube.

Analyzing Transactional Data

Introduction

The preceding section reviewed basic data warehousing processes, terminology, and cube structure. In this part of the paper we will describe how data marts are designed, constructed, populated, and used in a typical high volume e-commerce environment.

Though the information in this section is generalized to meet the requirements of this paper, the examples are based on real-world experiences acquired by Microsoft Consulting Services (MCS) at a number of major Internet retail sites.

For illustrative purposes, we will use the FoodMart sample database that is included with Microsoft SQL Server 7.0 OLAP Services. FoodMart is a fictitious international grocery store chain that uses OLAP data marts to track and analyze sales, customers, warehouses, stores, and products.

The overall data warehouse development process is quite involved and thus beyond the scope of this paper. However, there are several references at the end of the paper that treat this subject in more depth.

The emphasis in this section is on the following:

1. Showing a sample data mart in the OLAP Manager to illustrate how dimensions, measures, and cubes are depicted and manipulated.
2. By comparing two dissimilar fictional companies we can show how cubes can be built to satisfy the requirements of each.

- Expected challenges on typical data warehousing projects in dot-com environments.

Comparing Two Fictitious Companies

For the purposes of this paper we need to compare and contrast the business models and operating principles of our two companies: FoodMart Incorporated and FictitiousVirtualStorefront.com.

FoodMart Incorporated

FoodMart, incorporated in 1946, is a brick and mortar international grocery chain with stores and warehouses located in Canada, Mexico, and the USA.

FoodMart is concerned about efficiencies in its widespread stores and warehouses so therefore gathers detailed information about these operations. FoodMart also tracks the buying patterns of its customers.

FoodMart is a large operation with hundreds of employees across 24 store locations and 24 warehouses serving thousands of customers.

FictitiousVirtualStorefront.com

FictitiousVirtualStorefront.com is a high-volume, multistorefront Internet shopping portal with business offices in Lake Forest, CA. Currently, FictitiousVirtualStorefront.com has 10 different stores available on its Web site, ranging from an office furniture outlet to a lawn mower super store. They also sell books and CDs.

FictitiousVirtualStorefront.com eschews owning warehouses, instead preferring to let proven distributors handle all fulfillment tasks. Having no warehouses allows the company to easily add new stores and avoid problems associated with managing huge inventories of disparate goods.

FictitiousVirtualStorefront.com's business model allows the company to serve hundreds of thousands of customers with a staff of less than 500 employees. FictitiousVirtualStorefront.com was incorporated a few months ago.

The Two Companies Compared

The table below compares the two companies in a number of ways. These comparisons will provide a basis to drive the high level design of the FictitiousVirtualStorefront.com data mart.

FoodMart	FictitiousVirtualStorefront.com
Brick and mortar	Online virtual storefront
Many locations	One location
Owns warehouses	Uses distributors
Inventory in warehouses	No inventory
Thousands of customers	Hundreds of thousands of customers
Conducts many product promotions	Conducts many product promotions
Sells grocery products	Sells almost every type of product imaginable
Will always just sell grocery products	Can add new product lines and virtual stores quickly
Always uses its own warehouses	Can select from a variety of distributors for the same product
Tracks customer buying patterns and demographics	Tracks customer buying patterns, demographics, and Web usage
Tracks product sales	Tracks product sales
Tracks warehouse and store performance	Tracks distributor fulfillment performance

Runs the business on narrow margins and cost savings where possible	Relies on high volume sales and extremely competitive margins
Contacts customers through flyers on intermittent basis	Sends e-mail to customers on a regular basis

FoodMart's Data Analysis Requirements

Business analysts and executives at FoodMart require detailed information regarding the following:

Store Performance

- ✎ Which stores are the high producers and which are the laggards?
- ✎ Which products sell better at some stores than others?
- ✎ Which promotions are successful at which stores?

Product Sales and Trends

- ✎ Which are the hot products and which ones are not moving?
- ✎ What are the profit margins on each of the products?
- ✎ How are buying patterns for individual products affected by price fluctuations?
- ✎ How does the time of purchase affect the popularity of a product?
- ✎ What are the characteristics of customers that buy this product?
- ✎ Which brands sell the best to which customer base?

Promotion Performance

- ✎ What was the effect of the promotion on the volume of products sold?
- ✎ Was the promotion cost effective?

Customer Preferences and Buying Patterns

- ✎ How many men buy this product?
- ✎ How many married women shop in this store?
- ✎ How does education affect the buying patterns of this product family?
- ✎ How does changing product margins affect men's versus women's purchases?

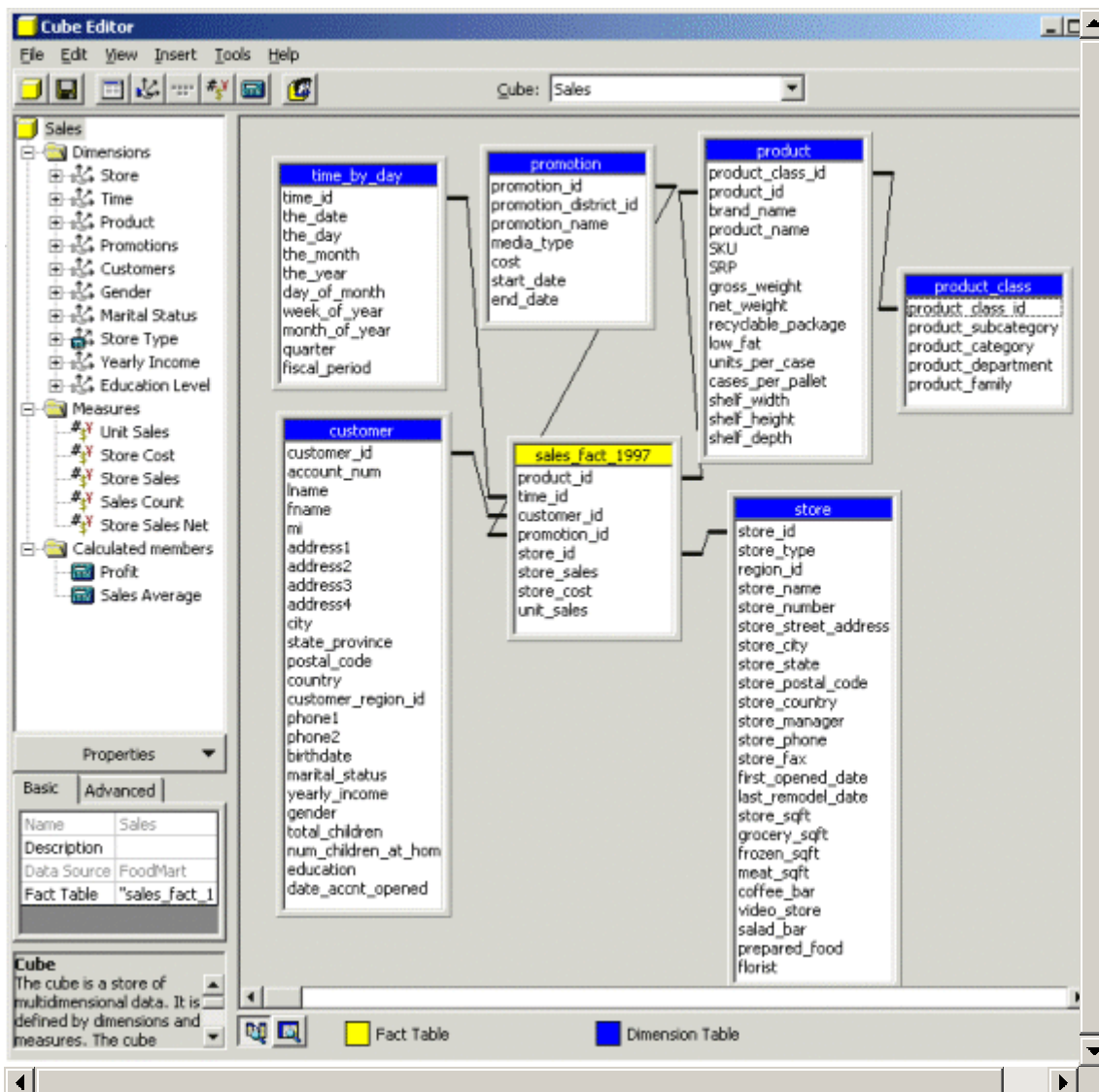
Warehouses

- ✎ Which warehouses are supplying products in a timely fashion?
- ✎ What are the inventory levels at each of the warehouses?
- ✎ Which warehouses are the most profitable?

OLAP Cubes Constructed to Meet FoodMart's Requirements

Now that we know the types of queries that FoodMart is interested in getting answers to, we will now show and discuss the OLAP cubes that are designed to meet these needs.

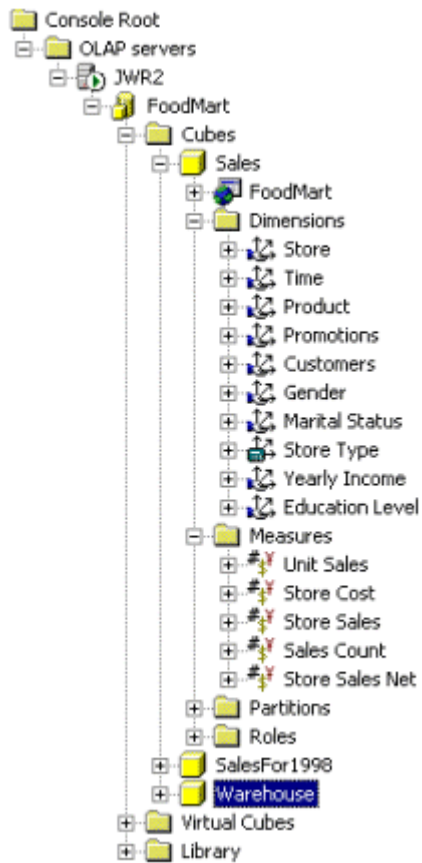
First, here is the Sales cube shown in the OLAP Manager Cube Editor.



If your browser does not support inline frames, [click here](#) to view on a separate page.

Dimensions, measures, and calculated members are displayed along the left side of the screen in the tree view. The right-hand portion of the editor displays the table view for the Sales cube. The tables depicted are the source tables for all the data in the Sales cube.

The following is an expanded and more close-up view of the tree view for the Sales cube.



The next image is of the Metadata view of the same Sales cube. Here we can clearly see that the Store dimension is made up of the following levels: Country, State, City, and Name. By looking back at the table view above we can see that there are several dimensions related to fields in the customer table: Customers, Gender, Marital Status, Yearly Income, and Educational Level. The Metadata view of the Sales cube also includes all of the measures, calculated members, and other administrative information.

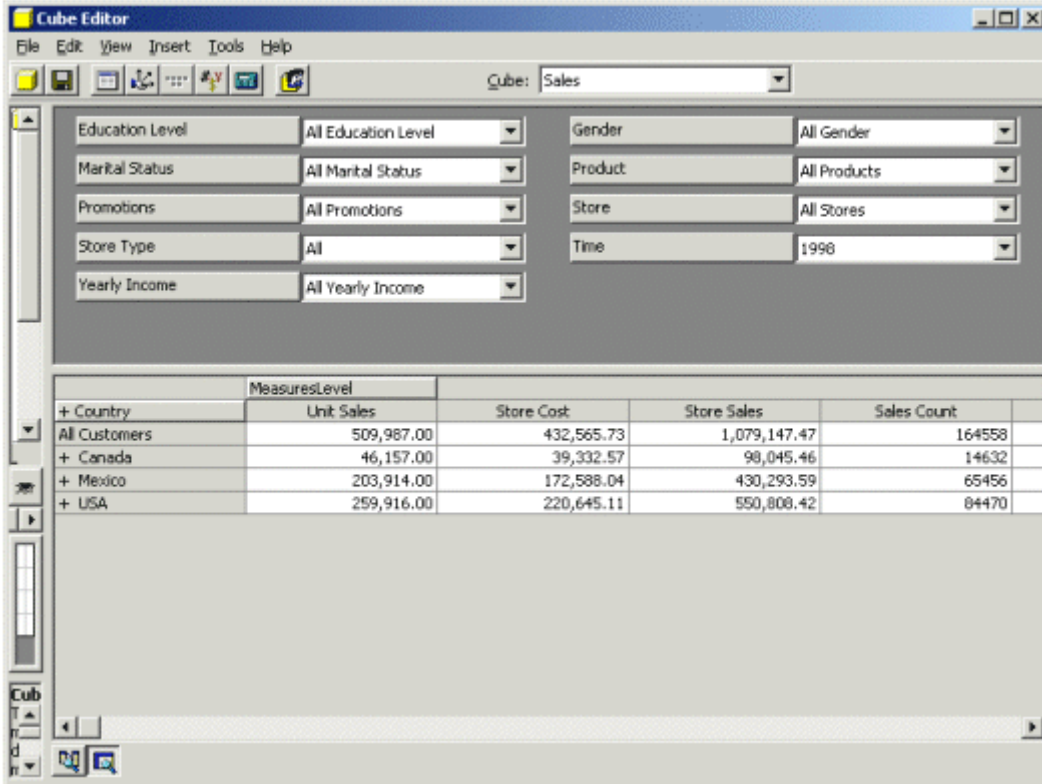
Getting Started		Metadata	Data
Sales			
Dimensions:	Store, Time, Product, Promotions, Customers, Gender, Marital Status, Store Type, Yearly Income, Education Level		
Store	(All), Store Country, Store State, Store City, Store Name		
Time	Year, Quarter, Month		
Product	(All), Product Family, Product Department, Product Category, Product Subcategory, Brand Name, Product Name		
Promotions	(All), Promotion Name		
Customers	(All), Country, State Province, City, Name		
Gender	(All), Gender		
Marital Status	(All), Marital Status		
Store Type	(All), Store Type		
Yearly Income	(All), Yearly Income		
Education Level	(All), Education Level		
Measures:	Unit Sales, Store Cost, Store Sales, Sales Count, Store Sales		
Calculated Members:	Profit, Sales Average		
Source Tables:	"product", "product_class", "sales_fact_1997", "promotion", "time_by_day", "customer"		
Processed:	6/17/2000 5:40:01 PM		
Type:	MOLAP		
Size:	2.90MB		
Data Source:	FoodMart		
Cube is:	Read Only		

If your browser does not support inline frames, [click here](#) to view on a separate page.

The last view of the Sales cube is the Data view in the Cube Editor, which is displayed below. This view is used to drill down into the data for development and design verification purposes. It is not intended as an end-user tool and does not have the extended feature sets that are found in tools such as [Knosys ProClarity](#), [OLAP@Work](#) from [BusinessObjects](#), or Cognos' [NovaView](#).

However, it should be clear that the Sales cube has the correct dimensions and measures to provide answers to the questions posed in FoodMart's requirements. (Note that not all of the measures are visible in the image.) You can experiment with levels within the measures by using the drop-down list boxes provided.

To drill down into lower levels of geographical information you just need to double-click the "+" on the left side of the grid. In this manner it is possible to drill down through Country, State, City, and all the way to the individual customer.



The screenshot shows the 'Cube Editor' application window. The title bar reads 'Cube Editor'. The menu bar includes 'File', 'Edit', 'View', 'Insert', 'Tools', and 'Help'. Below the menu bar is a toolbar with several icons. A dropdown menu shows 'Cube: Sales'. The main area contains several filter controls:

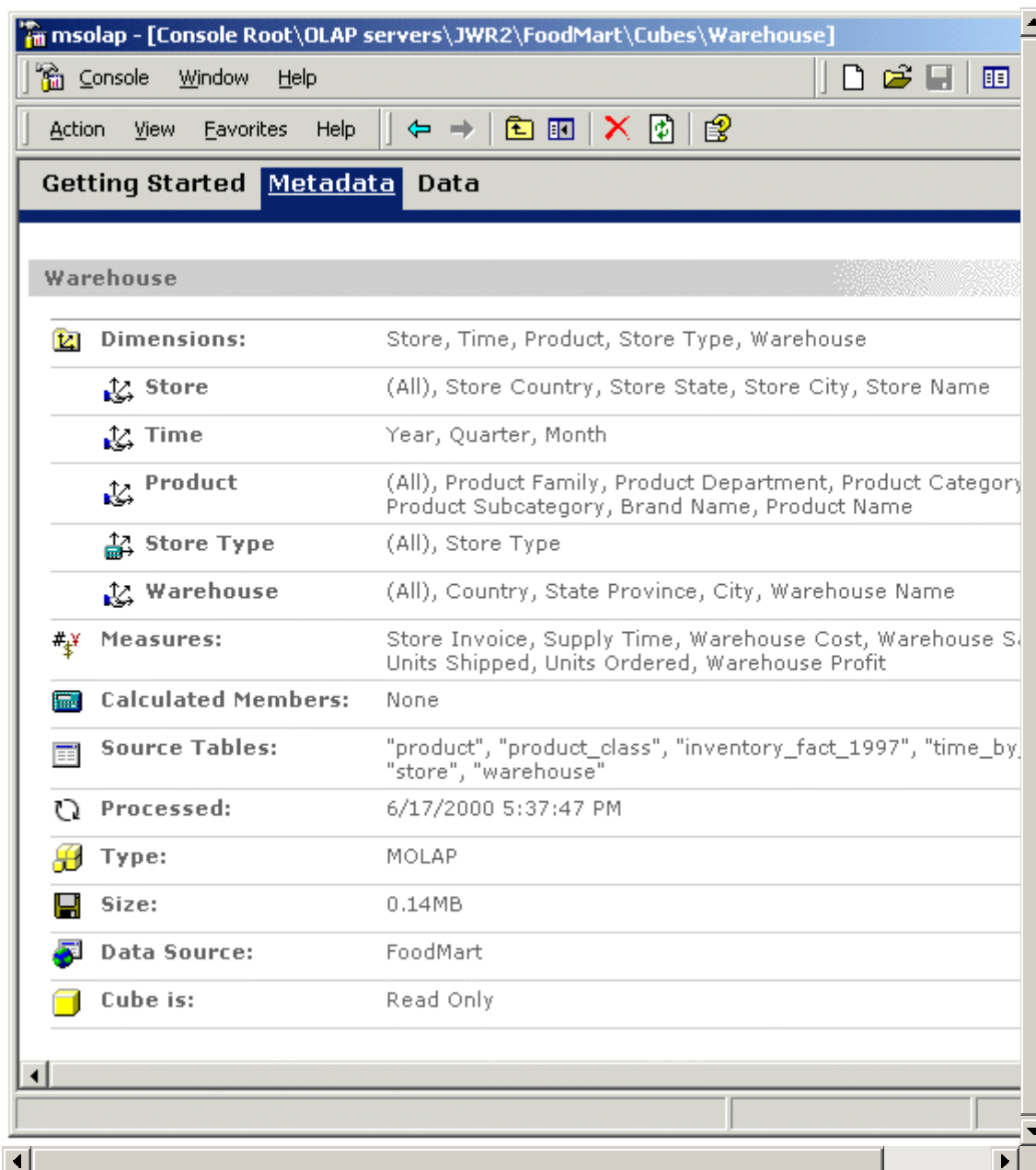
- Education Level: All Education Level
- Gender: All Gender
- Marital Status: All Marital Status
- Product: All Products
- Promotions: All Promotions
- Store: All Stores
- Store Type: All
- Time: 1998
- Yearly Income: All Yearly Income

Below the filters is a table with the following data:

	MeasuresLevel				
+ Country	Unit Sales	Store Cost	Store Sales	Sales Count	
All Customers	509,987.00	432,565.73	1,079,147.47	164558	
+ Canada	46,157.00	39,332.57	98,045.46	14632	
+ Mexico	203,914.00	172,588.04	430,293.59	65456	
+ USA	259,916.00	220,645.11	550,808.42	84470	

If your browser does not support inline frames, [click here](#) to view on a separate page.

For completeness, the Warehouse cube is shown in the Metadata view to verify that it meets the requirements stated previously.



If your browser does not support inline frames, [click here](#) to view on a separate page.

FictitiousVirtualStorefront.com's Data Analysis Requirements

We now have a good idea of FoodMart's requirements and how the designed cubes generally meet these requirements. In this section we will examine FictitiousVirtualStorefront.com's requirements in more detail and determine what changes and additions are needed.

In general, the questions that the data warehouse system needs to answer are similar to those for FoodMart. In terms of store performance, product sales, promotions, and customer preferences the two businesses require almost identical analysis structures.

The manner in which the organizations use this information may be quite different, however. For instance, the ability to personalize the shopper's experience could be enhanced by making use of buyer preferences, intelligent cross-selling strategies, and targeted e-mail campaigns. This topic is beyond the survey nature of this paper.

Another difference is that the "stores" in the case of FictitiousVirtualStorefront.com are purely virtual and essentially represent a broad category of product.

Of course, FictitiousVirtualStorefront.com does not own warehouses, therefore the warehouse cube is not needed to run their business.

Keeping Track of Orders and Shipments

Because FictitiousVirtualStorefront.com does not ship items from its own warehouses, it must depend on outside suppliers to do so. This process requires more record-keeping in the form of additional support tables and special matching processes.

To understand why these tables are needed, we must first look more deeply into the entire fulfillment process.

When an online customer orders an item (or several items), an Order record is created for that purchase. Individual Order Item records are also created for each line item of the order. Eventually these records are processed, including a credit authorization.

At this point, the order is ready to be sent electronically to the correct fulfillment partner (distributor). Just before the order is sent, a Fulfillment (Shipment) record is created to track status of the order. Later, after the order is sent to the customer by the distributor, FictitiousVirtualStorefront.com receives notification and the Fulfillment record is updated with information such as shipping agent, time of shipment, backorder status, actual shipping cost, whether the order was a full or partial shipment, and other related information. This additional information can be used to produce more useful measures in their respective cubes.

Because of this necessary additional order-related complexity, FictitiousVirtualStorefront.com requires additional cubes to provide the needed analysis on a daily basis. These cubes allow product managers and other company officials quick access to orders "booked" (taken), orders "shipped" (completed), and discrepancies between the two.

The Orders, Order Items, Shipped, and Shipped Items cubes represent the primary additions required relative to FoodMart's array of cubes. These cubes will look similar to FoodMart's Sales cube, with the addition of the information available from the fulfillment house. Cubes for RMA (Returned Merchandise Authorizations) and various reporting cubes could be added as well.

With the additional cubes, FictitiousVirtualStorefront.com can now answer the following:

1. Which distributors offer the most competitive prices?
2. Which distributors offer the quickest service?
3. How often are customers subjected to backorders, delays, and partial shipments?
4. How many outstanding and unfulfilled orders are in the pipeline?
5. What is the average time it takes for a customer to receive an order?
6. How much profit do shipping and handling charges generate for various products?
7. What is the cost of offering free shipping on a promotional basis?

Typical Challenges and Constraints

There are many common challenges that arise during the planning and building of an enterprise data warehousing solution. These challenges often continue throughout the life cycle of the project.

Here we list some of the more typical problems that must be addressed and overcome. In rapidly moving environments, such as dot-com companies, these challenges are often even more prevalent.

- ✗ If the project does not have a clear vision and scope then it will likely not succeed. It is essential to establish the overall vision, make it available to all stakeholders, and then obtain approvals from the appropriate management level. The scope should clearly state what is included in the project and what is not.
- ✗ Successful software projects of any kind must also have a detailed functional specification that serves as an overall plan for the project. Attention not paid to building this fundamental document will result in projects that are late, cancelled, or that fail to meet the company's needs.
- ✗ It is often not possible to assemble all the key end-user personnel at one time to get initial requirements. This situation is improved by acquiring strong sponsorship at the executive level and following up, in person, on all meeting requests.
- ✗ Correct business requirements are not captured because the interviewer does not "speak the language" of business. This can occur when a more development oriented person attempts to run the interview process. It is best to have a business analyst conduct interviews.
- ✗ During requirements gathering interviews, end users attempt to provide too much information regarding the "look and feel" of the target application instead of focusing on information

requirements.

- ✗ Business managers sometimes try to provide all the requirement details instead of allowing their staff analysts to contribute. Often the managers are not close enough to the everyday operations to fully understand the requirements.
- ✗ Often, some of the data used in data marts relies on information from disparate sources such as financial and CRM databases. This requires considerable coordination to have all the information from these various sources arrive in a timely and accurate manner. This issue underscores the importance of having a separate staging system that collects the data from different sources.
- ✗ To support the notion of meaningful levels within dimensions, it may be necessary to create consistent hierarchies within the different informational stores in the enterprise. For instance, product databases often require the definition of new type/subtype hierarchies in order to be more useful within the context of a data mart. This becomes even more of a design challenge when working with products that have wide-ranging characteristics, as is the case with FictitiousVirtualStorefront.com and its multiple storefronts.
- ✗ In resource constrained environments it is often the case that data warehouse development must be conducted on multi-use machines—possibly the OLAP production (report) server. Because it will likely negatively impact performance, this practice should be avoided at all cost. If the business is seriously resource-constrained and the sharing condition cannot be avoided, then incremental processing of cubes may be necessary to mitigate any performance issues.
- ✗ In a fast-paced and immature production environment, as is often found at dot-coms, the OLTP system has minimal or no technical information regarding structure of the databases, tables, and procedures. Without useful data dictionaries and process diagrams, OLAP developers will need to spend too much time analyzing existing tables and interviewing key database administrators—if the administrators are even available.
- ✗ Inconsistent and undocumented hardware configurations also pose tremendous challenges to the OLAP development team. Unscheduled and unanticipated hardware and network upgrades also cause inefficiencies in the development process.
- ✗ Though database replication is often a preferred way to move data within the enterprise, poor configuration and inconsistent operational practices can disallow replication as a data movement option. This may require other, more restrictive data transfer methods to be used.
- ✗ Currently, an Excel spreadsheet cannot contain more than 64,000 rows. There are some common dimensions (for example, products or customers) that normally exceed this row limit. If required, large dimensions can be fully viewed with other third-party tools such as Knosys ProClarity.
- ✗ Too often a "big bang" approach is used in an attempt to build an overly ambitious and complex initial data warehouse. Avoid this by concentrating on building smaller, focused, and simpler data marts that offer obvious high value to the key decision makers of the company. Then build on these early successes.
- ✗ Large-scale data warehouses and data marts require a tremendous amount of system resources, including processor power, system memory, and disk space. Make sure that you plan carefully for the eventual growth of the system. There are precise formulas for sizing the solution available in the Microsoft documentation.
- ✗ Though it is a good idea to start out with smaller, well-defined data marts, eventually the enterprise must develop an overall data warehousing strategy to avoid continued integration and operational problems.

Site Server Usage Analysis Plus Toolkit

This section summarizes the functionality of the Site Server Usage Analysis Plus Toolkit, which is available for download on the Microsoft Web site. This toolkit takes full advantage of the Data Warehousing Framework to allow end users to analyze detailed site usage information stored in multiple Web server logs.

As discussed in the previous section, OLAP Services can be used to gather, aggregate, and organize transactional data at an e-commerce Web site. Because transactional data resides in the enterprise OLTP system, the required movement of this information to the analysis system is from one database to one or more other DTS databases. Data extraction, transformation, and loading is accomplished through a combination of DTS, replication, log shipping, and database copy operations.

Web server log information is not captured in a database but rather in simple log files, which grow to be very large on high-usage sites. Though the final destination of the log information is a data mart within the OLAP system, it is important to note significant differences between transactional data and Web server log information.

Transactional data on a popular e-commerce Web site is characterized by the following:

- ✗ Closely associated with product purchases, revenue, and profit
- ✗ Medium to high volume
- ✗ Links people with product purchases
- ✗ Links products with stores, distributors, promotions
- ✗ Contains little information about overall site usage patterns

In contrast, Web server logs capture the following:

- ✗ Detailed page view information including top requested pages, duration of view, viewing patterns over time
- ✗ Referring sites
- ✗ Exit pages
- ✗ General activity levels
- ✗ Browser information
- ✗ Unique visitor information

It is clear that the two data capture systems have the potential of providing rich and complementary analytical information. Having previously discussed how transactional data is used in e-commerce site data marts, we will now briefly summarize how log data is transformed into useful analytical information for sites that run Microsoft Site Server 3.0.

Issues With Site Server Analysis

Site Server Analysis is a powerful tool for analyzing both the usage and content of Internet sites. It includes some 40 standard reports and allows for customization and creation of new reports.

However, one of the drawbacks of the Site Server Analysis component is that it was designed with performance characteristics useful mostly in intranet environments. As a result, when this tool is utilized in the Internet environment, it is exercised beyond its designed performance envelope. The Usage Analysis Toolkit provides a set of documents, stored procedures, and cube templates to help address the following three main issues facing Site Server Analysis today, when deployed in large Internet sites:

- ✗ Internet Information Server (IIS) log import time is excessive when the log file is large (approximately 5 gigabytes [GB]).
- ✗ Report Writer runs for an excessive amount of time when the analysis database grows large (beyond 10 GB).
- ✗ Customization of Report Writer templates takes too much time and requires developer-level resources.

Concept

The basic concept of Site Server Usage Analysis Plus is to:

- ✗ Utilize existing tools to import the log files.
- ✗ Extend the database schema to support new reporting requirements.
- ✗ Use OLAP Services to build multidimensional OLAP cubes.
- ✗ Use common OLAP reporting tools, such as Excel or other third-party offerings, to allow analysts to view the multidimensional site data.

Solution

The current Site Server Analysis database schema is constructed as a data warehouse that stores imported log file data. The schema includes the expected fact tables and their dimensions.

The solution addresses both query time and data storage trade-offs by leveraging the existing Analysis database and creating custom aggregate tables designed specifically to improve report performance. Stored procedures are used to populate these aggregate tables for loading into the multidimensional cube. This is the approach used in this effort.

Once the data has been loaded into the OLAP cube, any OLAP client-access tool that utilizes the

PivotTable Services may be used to report on the data.

Here is the step-by-step process used to accomplish the goals of improved log file analysis:

- ✎ Analyze reports desired with a view to design aggregate tables required to support the creation of necessary OLAP cubes.
- ✎ Use Site Server Usage Import to load IIS log data into the analysis database. Add new indexes to the database to reduce OLAP cube loading time.
- ✎ Use stored procedures to create and populate the aggregate tables.
- ✎ Define OLAP cubes with the Cube Editor Wizard tool provided by OLAP Services.
- ✎ Import the analysis data warehouse and aggregate tables into the OLAP cubes for processing.
- ✎ Use client tools to view and report on the analytical data.

Available Reports

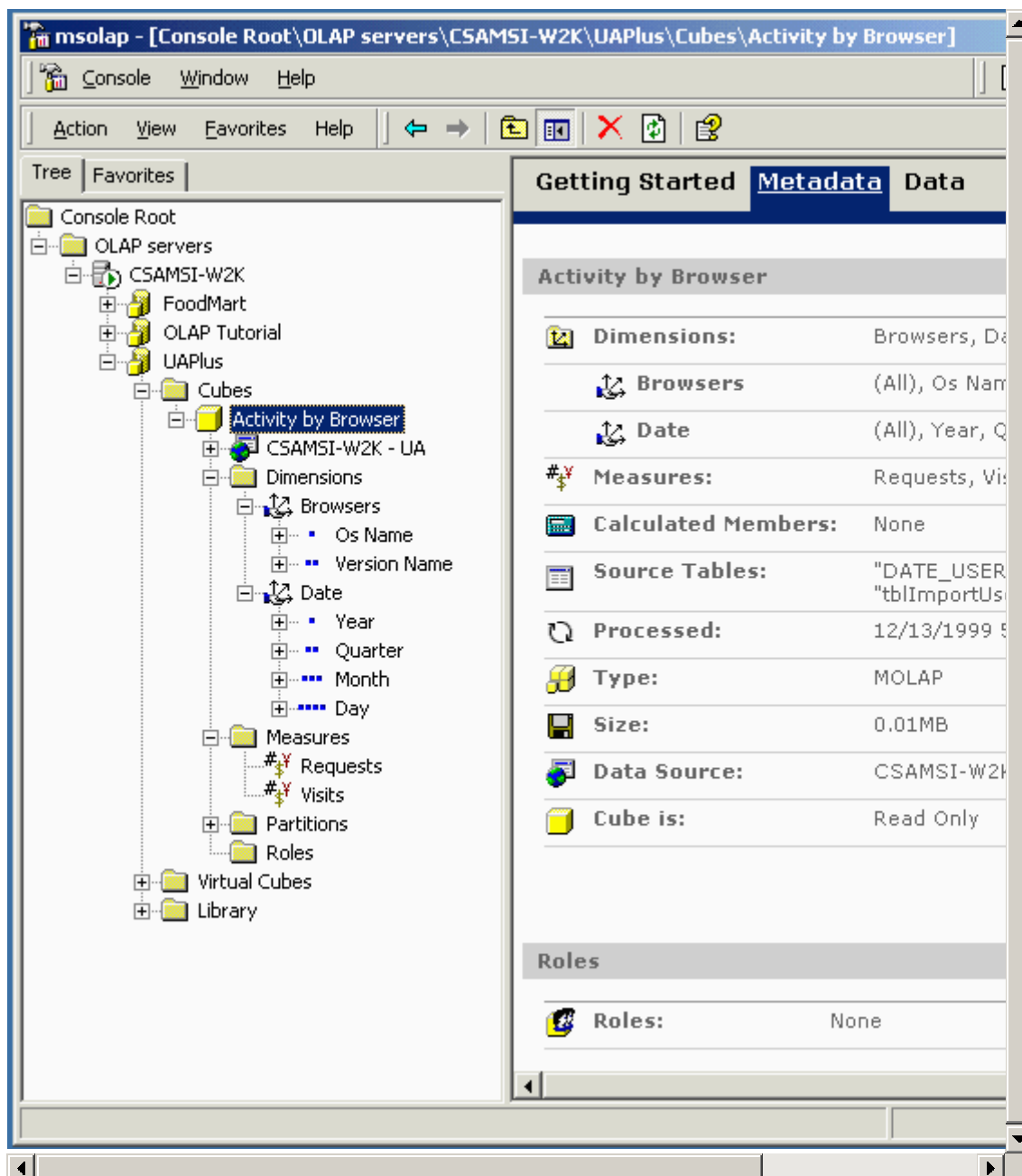
The following table lists the reports that are available in the toolkit.

No.	Report name
1	Number of page views to a target site
2	Number of page views to a target domain
3	Number of unique GUIDs/users to a target site
4	Number of unique GUIDs/users to a target domain
5	Number of page views from a referral site
6	Number of unique GUIDs from a referral site
7	The hours during which a target site receives the most page views
8	Most visited pages in a target site
9	Top N referral pages by page views
10	Top N target sites by unique GUIDs
11	Page views to a target site from month to month
12	Page views to a target site from quarter to quarter
13	Which referral site provides the most referrals from month to month, or from quarter to quarter
14	In which month a target site receives the most/least page views
15	Number of page views to a target page
16	Number of page views to a target site
17	Number of page views from a referring page
18	Number of page views from a referring page to a target page
19	Number of page views from a referring domain to a target page
20	Number of page views from a referring site to a target page
21	Browser share by unique GUIDs
22	Browser market share by usage (by hits)
23	Number of hits by user agent (browser)
24	Frequently used page before exiting across the network
25	Number of page views to a main directory
26	Number of page views to a subdirectory

27	Number of hits by user agent (browser)
28	Most requested pages
29	Least requested pages
30	Top entry pages
31	Top entry requests
32	Top exit pages
33	Single access pages
34	Most accessed directories
35	Summary of activity by time increment
36	Activity level by day of the week
37	Top referring sites
39	Top referring URLs
40	Most used browsers
41	Netscape browsers
42	Microsoft Internet Explorer browsers
43	Most used platforms

OLAP Cubes for Log Data Analysis

The documentation included with the toolkit provides detailed information on how to define the required cubes using OLAP Manager tool, described earlier in this paper. Here is an example of the Activity by Browser cube. This simple cube tracks browser requests and visits by operating system, version, and date.



If your browser does not support inline frames, [click here](#) to view on a separate page.

The entire list of cubes provided in the toolkit includes:

- ✦ Activity by Browser
- ✦ Activity by Commerce Product
- ✦ Activity by Hour
- ✦ Activity by Referrer
- ✦ Activity by Referrer and First Request
- ✦ Activity by URI
- ✦ Ad Activity

More Information

In this section we summarized the features of the Site Server Usage Analysis Plus Toolkit and how this set of components utilizes the Data Warehousing Framework to provide useful site information. This information is not available in the transactional data capture systems. The complete documentation, utilities, and other toolkit components are located on:

<http://www.microsoft.com/siteserver/commerce/DeployAdmin/uaplus.htm>.

After downloading the toolkit and unzipping its contents, refer to the document named UAPlus.doc for detailed information on the following topics:

- ⚡ Designing aggregate tables and stored procedures
- ⚡ Importing IIS logs
- ⚡ Running required stored procedures
- ⚡ Creating OLAP cubes
- ⚡ Hardware planning
- ⚡ Constructing artificial levels (which will not be required with the introduction of SQL Server 2000)
- ⚡ Processing cubes
- ⚡ Creating reports
- ⚡ Software requirements

SQL Server 2000 and Analysis Services

Introduction

This section summarizes features in SQL Server 2000 that support advanced data analysis. The complete SQL Server 2000 Beta 2 Evaluation Guide can be downloaded at:

<http://www.microsoft.com/sql/productinfo/sql2krev.htm>

Integrated and Extensible Analysis Services

The goal for Analysis Services (formerly OLAP Services) is to provide a complete, end-to-end platform for analysis. This analysis platform includes relational storage, data extraction, OLAP optimization and querying, data mining, and semantic modeling.

Analysis Services is designed for database administrators and application developers rather than an elite audience of statistical experts. Anyone with a solid foundation in Structured Query Language (SQL) and Microsoft Visual Basic® should be able to understand, use, and programmatically extend the offerings in Analysis Services.

Beyond the core data warehousing and OLAP features that were offered in SQL Server 7.0, Analysis Services in SQL Server 2000 provides integrated data mining and new graphical tools for building and managing analysis data. Security and dimension enhancements improve flexibility.

Analysis Manager allows multiple users to administer an analysis server. Remote partitions can be used to distribute a cube's data among multiple analysis servers and administer the cube on a central analysis server.

The Virtual Cube Editor allows for virtual cube editing and is similar in function to the Cube Editor.

Security is now definable both for dimensions and cells. Roles can be used to control end-user access to dimensions, limiting access to individual dimensions, levels, and members, and enabling a variety of read and read/write permissions.

Cell security may also be defined by setting role options in Analysis Manager.

Roles can be restricted to any combination of a cube's cells.

New Dimension Types

The following new dimension types are introduced in SQL Server 2000 Analysis Services:

- ⚡ *Parent-child*. Supports hierarchies based on parent-child links between members in columns in a source table.

- ⚡ *Ragged*. Has at least one member whose logical parent is not in the level immediately above the member.
- ⚡ *Changing*. Permits a wider array of changes than other dimensions without requiring a cube to be fully processed after changes.
- ⚡ *Write-enabled*. Can be updated through Analysis Manager and any client applications that supports dimension writeback.
- ⚡ *Virtual*. Has members that are determined by the members of another dimension.

Integrated Data Mining

Integrated data mining is a new offering and is delivered as part of Analysis Services. Data mining technology helps users analyze data in relational databases and multidimensional OLAP cubes to uncover patterns and trends that can be used to make predictions.

- ⚡ The data mining capabilities in SQL Server 2000 are integrated with both relational and OLAP data sources.
- ⚡ The results of data mining can be used to create additional cube dimensions for further OLAP data analysis.
- ⚡ The data mining features are incorporated in an open and extensible implementation of the new OLE DB for Data Mining specification.
- ⚡ Microsoft Decision Trees is a new data mining algorithm developed by Microsoft Research. An implementation of the Microsoft Decision Trees algorithm could be used to identify individuals who are most likely to click on a particular banner ad or buy a specific product from an e-commerce site.
- ⚡ The Microsoft Clustering algorithm uses a nearest neighbor method to group records into clusters that exhibit some similar, predictable characteristic. For example, the Microsoft Clustering algorithm might be used to assess customer purchase behavior by age.
- ⚡ New wizards, editors, and other user interface elements are provided to simplify the design, creation, training, and browsing of data mining models.

Closed-loop Business Internet Analytics

Integrated data mining is a key element of Microsoft's strategy to deliver closed-loop Business Internet Analytics in Windows DNA 2000. Closed-loop Business Internet Analytics involves:

1. Collecting information from the online behavior of customers as they browse and search a Web site.
2. Analyzing that information to uncover trends and make predictions (using data mining).
3. Personalizing ads and content for users based on this analysis, such as displaying appropriate banner ads to cross-sell products.
4. Driving decisions back into the operational systems using OLAP Actions.

Integration with Commerce Server 2000 simplifies data collection for user clickstreams, transactions, purchase histories, and other customer activity data that builds the holistic view of the business activity on a site. This integrated data then enables business managers to modify or create new marketing programs, promotions, and advertising campaigns as well as drive merchandising and site personalization.

Linked Cubes and HTTP Access to Cubes

Analysis Services introduces linked cubes and HTTP access to cubes, two technologies that Web-enable analysis so that users can leverage cubes owned by partners or offered for sale by research firms.

Linked cubes are cubes that are defined and stored on other analysis servers, including servers outside the corporate firewall. Linked cubes allow data providers to create, store, and maintain a cube on one analysis server while the cube is also available as one or more linked cubes on multiple analysis servers; data sources using HTTP and HTTPS are supported.

Analysis Services uses an HTTP listener built into the server to provide cube access over HTTP. This enables organizations sharing cubes or accessing remote cubes to do so securely over HTTP and through a firewall without opening dedicated ports on a Web server.

Distributed Partitioned Cubes

SQL Server 2000 extends software scale out to data warehousing solutions through Distributed Partitioned Cubes. In order to attain flexible data storage and additional query performance, developers can use the Partition Wizard to easily scale out one logical cube into separate physical partitions on multiple servers with complete transparency.

Large Data Set Analysis

Analysis Services in SQL Server 2000 supports MOLAP (Multidimensional OLAP) dimensions in the range of tens of millions of members by utilizing up to 64 GB of memory. Through the addition of ROLAP (Relational OLAP) dimensions, which are dimensions stored as relational tables, Analysis Services supports dimensions on the order of hundreds of millions of members.

In contrast to the MOLAP storage mode, ROLAP does not cause a copy of the source data to be stored; the partition's fact table is accessed to answer queries when the results cannot be derived from the aggregations or client cache. Typical applications of ROLAP include large data sets that are infrequently queried (such as historical data) or data sets that are too large to be stored in MOLAP or HOLAP mode.

DTS Improvements

Data Transformation Services (DTS) has been updated to improve its abilities to move and transform data from any source.

The DTS Wizard now supports the movement of primary and foreign keys, which simplifies data migration between different vendors' relational database management systems.

Developers can access and manipulate the operations of the DTS data pump through an expanded number of interfaces and at multiple points during the progression of data transformations. The multiphase data pump allows for more flexible handling of transformation and insert errors or failures.

DTS packages can now be saved as Visual Basic code. This allows an easy learning path to developing DTS packages through programmatic interfaces, and when used in conjunction with Microsoft Visual SourceSafe® version control system provides an alternative method of version control and backup for packages.

DTS packages can now call each other when they are executed, which allows for more package reuse.

Web-based Analysis

Analysis Services includes DISTINCT COUNT, which is useful for analyzing user traffic on Web sites. The DISTINCT COUNT feature is a new type of measure that allows analysts to answer important questions like "How many unique users hit my site today?"

OLAP Actions

OLAP Actions allow end users to act upon the outcomes of their analyses to automatically drive business processes. An action is an end user-initiated operation upon a selected cube or portion of a cube. The operation can launch an application with the selected item as a parameter or retrieve information about the selected item.

By implementing actions, which are defined with the Action Wizard, developers can transform client applications from sophisticated data rendering tools to integral parts of a feedback loop in an enterprise operational system.

Custom Rollups

By default, rollups in Analysis Services are additive. New custom rollup operators provide a simple way to control how member values are rolled up to their parents' values. Members are tagged with one of the following valid operators: +, -, *, /, or ~. Each of these performs its indicated mathematical action. The custom rollup operator is applied to the member when evaluating the value of the member's parents.

Custom rollup operators are assigned to the name of a column, either when creating them as an optional feature of the new parent-child dimensions in Dimension Wizard or when adding them to existing dimensions via the Dimension Editor or Cube Editor.

Copy Database Wizard

The new Copy Database Wizard enables the simple task of copying and moving databases with minimal server downtime. This feature can be used to set up test environments, move databases between servers or instances, and perform hardware migrations. Copying operational databases to staging areas is a common practice in dynamic data warehousing environments.

Database copying is implemented as DTS tasks whose packages are executed on the destination machine(s), allowing for flexible scheduling. This procedure supports the copying of global namespace objects, error messages, logins, and jobs.

English Query

The primary purpose of the data warehouse is to provide useful information directly to a variety of analysts.

- ✦ English Query allows end users to pose questions in English instead of forming a query with an SQL statement. English Query transforms the user's question into a proper SQL query that pulls the desired results from the database or data mart.
- ✦ Authoring and deployment of English Query applications is now hosted in the Microsoft Visual Studio® version 6.0 development environment, which is included with SQL Server 2000. Visual Studio 6.0 includes wizards and other enhanced support for the development of English Query applications.
- ✦ Developers can target English Query applications at OLAP cubes via the Multidimensional Expression (MDX) generation capabilities of English Query.
- ✦ Question Builder enables end users to build queries graphically and retrieve information about an English Query model, including what English phrases can be used to ask about relationships.

SMP-related Features

SQL Server 2000 does more operations in parallel to take greater advantage of increasingly common symmetric multiprocessing (SMP) hardware, from dual processor systems prevalent even in small businesses to 16-way and 32-way systems in the data centers of Fortune 500 companies.

Parallel index creation is enabled by building subindexes for specific ranges of an index. Separate threads (running on separate processors if available) build these subindexes, fed by parallel scans. When the subindexes are completed, they are combined into a complete index by a coordinating thread. Multiterabyte data warehouses benefit from this feature when creating an index on a fact table that might otherwise take hours to complete.

Large Memory and SMP support

On Windows 2000 DataCenter Server, SQL Server 2000 extends its reach to 64 GB of RAM and up to 32 CPUs. This represents thorough support for scale-up scenarios and can be used in conjunction with scale-out techniques to handle the largest data sets and transactional loads.

VI SAN support

The VI SAN (Virtual Interface System Area Network) support in SQL Server 2000 allows SQL Server to communicate directly with devices connected via a SAN to transfer high volumes of data or transactions with low latency. Microsoft has worked closely with Gigaset (cLan) and Compaq (Servernet 2) to offer this direct access to SAN devices.

References

Books

Craig, Robert S., Joseph A. Vivona, and David Bercovitch. *Microsoft Data Warehousing: Building Distributed Decision Support Systems*. New York: John Wiley and Sons, 1999.

Kimball, Ralph, and Richard Mertz. *The Data Webhouse Toolkit: Building the Web-Enabled Warehouse*. New York: John Wiley and Sons, 2000.

Kimball, Ralph, Laura Reeves, Mary Ross, and Warren Thornthwaite. *The Data Warehouse Lifecycle Toolkit: Expert Methods for Designing, Developing, and Deploying Data Warehouses*. New York: John Wiley and Sons, 1998.

Peterson, Tim, Jim Pinkelman, and Bob Pfeiff. *Microsoft OLAP Unleashed*. Indianapolis, IN: Sams, 1999.

Scott, Mark. "Data Warehousing Step By Step," *SQL Server Magazine*, February, 2000.

Soukup, Ron, and Kalen Delaney. *Inside Microsoft SQL Server 7.0*. Redmond, WA: Microsoft Press, 1999.

Sturm, Jake. *Data Warehousing with Microsoft SQL Server 7.0 Technical Reference*. Redmond, WA: Microsoft Press®, 2000.

Thomsen, Erik, George Spofford, and Dick Chase. *Microsoft OLAP Solutions*. New York: John Wiley and Sons, 1999.

Online References

Microsoft SQL Server 7.0 Data Warehousing Framework

Web Sites

For more information on Microsoft's enterprise frameworks and offerings, see

<http://www.microsoft.com/enterpriseservices/>

For more information on OLAP, data analysis, and data warehousing see

<http://www.microsoft.com/sql/techinfo/olap.htm>

<http://www.microsoft.com/sql/techinfo/datanaly.htm>

<http://www.microsoft.com/sql/techinfo/datawarehousing.htm>

<http://www.microsoft.com/sql/default.htm>

Last updated July 13, 2000

© 2001 Microsoft Corporation. All rights reserved. Terms of use.
