Reprinted from DM REVIEW

CLAUDIA IMHOFF

It Takes Two to Tango...

Claudia wishes to thank Jonathan G. Geiger for his contributions to this month's column.

When data warehousing was first introduced in the 1980s, companies tended to build either a large data warehouse ("build it and they will come") or multiple independent data marts ("Chaos Theory").

The first approach involves a data store whose primary purpose is to compile an enterprise view of data for eventual access by a user community. In some architectures, this is called a back office; in the Corporate Information Factory, it's known as the data warehouse.

The second approach involves a set of data stores whose primary purpose is to provide the business users with easy, direct access to their data. These are almost universally known as data marts. Today, data marts may be either logical views into the data warehouse, a la Teradata or Netezza, or physically separate (cubes, stars, data sets, etc.), loaded into traditional relational databases.

Each of these early approaches to business intelligence (BI) was limited to the degree that it ignored the other.

Companies that implemented independent data marts with no thought about the back office or data warehouse were able to satisfy individual needs quickly, but found that they promulgated the inconsistency problems that were inherent in their legacy system environment. Further, because the same data was often used in several data marts, extract processes were duplicated, maintenance costs grew and the environment became very complex and unsustainable.[1]

On the other hand, companies that implemented a large data warehouse only, with no form of data marts considered in the design phase, certainly gained the advantage of having a single consolidated source of data. However, these companies also ran into significant problems.

We've had numerous engagements in which we encountered problems because the architectures did not include any form of design or preparation for their data marts. By the time we came in, the data warehouse reputation was suffering, and the cost of redesigning it and building the data marts was significant.

> It's a mistake not to include both data warehouse and data mart components in your architecture from the start – even if the marts are virtual.

Let's examine why a "data warehouse only" environment for traditional relational databases is less than ideal.

Design: The data warehouse is optimized to meet its primary objectives, which are collect data from multiple sources and disseminate it to data marts that can be accessed by business users. The data warehouse is based on the enterprise's business data model and is derived by transforming that model using a methodology for modeling the data warehouse. Within the Corporate Information Factory, the resultant model starts as a third normal form model that is subsequently denormalized to satisfy the storage and load performance objectives of the data warehouse. While this model meets these objectives of the data warehouse, it does not make it easy for the typical business user to access data directly. In the multidimensional architecture, the back office serves this function, and direct access to the back office is prohibited for many of the same reasons.

Empowerment: Business users want to be able to get information without intervention by the information technology group. The data marts, combined with business intelligence tools, are designed to provide an environment in which business users can access the information specific to their needs by themselves. Without data marts, the complexity of the data warehouse design requires intervention by either a power user or someone in information technology. Users sometimes get so frustrated that they avoid using the data warehouse at all and create extracts directly from the source systems, thus increasing the number of inconsistent and possibly redundant independent data marts.

Information Overload: The data warehouse (eventually) contains all of the detailed information needed to support all forms of strategic decisions. Unlike a data mart environment that can be tuned to each group of users and a specific set of needs, when data is accessed directly from the data warehouse, the programmer must deal with a larger and more complex environment to capture the needed data.

Performance: Data collection and dissemination are batch operations; data access is an interactive operation. Actions taken to optimize data collection can adversely impact performance of the data access activities. Similarly, actions taken to optimize direct data access activities can adversely impact performance of data collection and dissemination processes.

Flexibility: The data warehouse is built with full recognition that we don't know all the requirements. The warehouse design evolves; and over

time, tables and relationships need to be changed. When a data warehouse is used to disseminate data to the data marts, the end users are not impacted by changes to the data warehouse. In an environment in which reports and queries (sometimes hundreds of these) go directly against the data warehouse, a data warehouse change has a serious impact on the many query and reporting programs. Due to this impact, warehouse changes are more costly, and the flexibility of the warehouse suffers.

Development Time: The last problem we'll discuss is the development time. Due in part to the flexibility problems previously cited, building a data warehouse that needs to satisfy direct queries takes longer. The development time is lengthened because the requirements need to be better defined and because the design needs to consider both the data content and the information presentation. When data marts are included in the architecture, the data warehouse design focuses on data content and the data mart design focuses on information presentation.

Given loads of firsthand experience these problems, data warehouse experts now promote environments that contain these two data stores. After 20+ years, there are still significant differences in design philosophies, but it's a mistake not to include both data warehouse and data mart components in your architecture from the start – even if the marts are virtual.

References:

1. The problems with the marts-only environment is described in "Take a Trip and Never Leave the Farm," Claudia Imhoff, January 2003 issue of *DM Review*.

Claudia Imhoff, Ph.D., is the president and founder of Intelligent Solutions (www.intelsols.com), a leading consultancy on CRM and business intelligence technologies and strategies. She is a popular speaker, and internationally recognized expert and serves as an advisor to many corporations, universities and leading technology companies. She has coauthored five books and more than 50 articles on these topics. She may be reached at cimhoff@intelsols.com.

Jonathan Geiger is an executive vice president at Intelligent Solutions. His 30-year career includes data and repository management, data warehousing, project management, systems development and support, planning, training and quality assurance. He may be reached at jgeiger@intelsols.com.