

# The Problem with Dimensional Modeling

Bill Inmon

DM Review Magazine, May 2000

The problem with dimensional modeling is... well there really isn't a problem with dimensional modeling. In terms of capturing requirements for DSS analysis, dimensional modeling is undoubtedly the best technique for design that there is. Hundreds of organizations have used dimensional modeling successfully and are quite satisfied with the results. There are classes and conferences where dimensional modeling is the focus and the attendees walk away with a happy smile on their face.

So whatever could be the problem with dimensional modeling? In fact, is there really a problem here?

In order to understand what the issues are, consider the results of a dimensional modeling exercise. Typically the database design that results from a dimensional model is called a star join (or sometimes a snowflake structure). A star join is made up of one or more fact tables coupled with any number of related dimensions. A star join is built by gathering and assimilating user requirements. A star join represents the optimal denormalization of data that best suits the requirements of the users.

The starting point for the design of the star join and dimensional modeling is the process of requirements gathering. Users are interviewed and requirements are synthesized from the users' statements. Once the requirements are gathered, an assimilation of requirements from all users interviewed takes place, leading to the determination of the shape and content of the star join. The database designer knows how to build the star join because of the requirements analysis that precede the design activities.

The result of the star join when built properly is that the shape and content of the star join are optimal for the requirements of the users that have participated in the requirements-gathering process. Star joins are great for representing the views of people who are of a like mind, but different groups of people want their own star join.

While the star join fits the needs of the users who have been interviewed, not surprisingly the star join does not fit the needs of all users. For a variety of reasons, there will be users who do not have input into the dimensional modeling process. Indeed, the problem with a star join design is that it optimizes the access of data for one group of people at the expense of everyone else. A star join with one shape and set of contents is optimal for one group of people and another star join with another shape and contents is optimal for another group of people. Because the star join is shaped around user requirements and because user requirements vary from one type of user to the next, not surprisingly different star joins are optimal for different types of users.

The advocates of dimensional modeling would like to believe that a single star join can be created for the optimization of everyone's requirements. Such is not the case at all. The world has long ago discovered that the single database for all purposes was pie in the sky. The single database to serve all purposes and all requirements simply has never been a reality.

There are a host of very good reasons why different organizations need their own unique star joins and cannot share a star join with another organization. Some of these reasons are:

- **Sequencing of data.** Finance likes to see data sequenced one way; marketing likes to see data sequenced another way.
- **Data definitions.** Sales defines a sale as closed business; accounting defines a sale as booked business.
- **Granularity.** Finance looks at things in terms of monthly revenue; accounting looks at things in terms of quarterly revenue.
- **Geography.** Sales looks at things in terms of sales territories by ZIP code; marketing looks at things in terms of states.
- **Products.** Engineering looks at things in terms of existing products; sales looks at things in terms of future products.
- **Data relationships.** Finance looks at the relationship between a customer and a sale; accounting looks at things through the eyes of an account/sales relationship.
- **Time.** Accounting looks at the world through scheduled closing dates; finance looks at the world through calendar dates.
- **Technologies.** One star join (from finance) is in one technology and another star join (from marketing) is in another technology.
- **Sources of data.** One source system feeds one star join while another source system feeds another star join.
- **Management.** Management looks at things through the organization chart, the investors look at things through the marketplace.

The differences in the way that parts of the business operate are much more vast and much more profound than the obvious examples presented here.

The way that the different departments of the organization conduct their business is like the six blind men describing an elephant. Each blind man is touching a different part of the elephant and is describing it. To a casual observer, there appears to be no cohesive pattern in the very different descriptions given by the six blind men. Having different organizations care for different aspects of the business is simply the way business runs. There are good reasons for different departments looking at the data and the business in very different lights. The consequence of this need to look at the same business in different ways results in the need for very different star joins for different departments. The star join that is optimal for one department is hardly optimal (or even recognizable) for another department. The result is that because of the nature of the way business is run different departments require different star joins.

When each department has its own star join which has been designed specifically for it, the world is at peace. But herein lies a host of problems.

The problems that arise with star joins and dimensional modeling do not become apparent until multiple star joins appear on the horizon. (The problems to be discussed with multiple star joins cohabiting in the same environment is well documented and will be only lightly touched upon here in the interest of not repeating what has been well discussed in the industry.)

When there are multiple independent star join environments, the same detailed data appears in each star join. There is no reconcilability of data, and new star joins require the same amount of work for creation as older star joins. As a result:

- Each star join contains much of the same detailed data as each other star join. Star joins grow unnecessarily large when each star join thinks that it has to go back and gather the same detailed data that all other star joins have already gathered.
- The results obtained from each star join are inconsistent with the results obtained from every other star join. Furthermore, the ability to reconcile the differences is

almost nil.

- New star joins require as much work to create as the original star joins. There is no foundation to build on when each star join is built independently.
- The interface to the supporting applications that feed the star joins becomes unmanageable.

In short, simply doing dimensional modeling as a basis for data warehouse design leads down a dark path when multiple star joins are considered. It is never apparent that there is a problem with star joins when you are looking at just one star join. But when you look at multiple star joins, the limitations of dimensional modeling become apparent.

Does this mean that dimensional modeling is invalid as a database design technique for data warehousing? The answer is: not at all. Dimensional modeling and star joins fit the bill very nicely for data marts. In fact, if I had to design a data mart tomorrow, I would not consider using any other approach.

But when it comes to the foundation data, that's another story. Figure 1 shows that the foundation data the data warehouse requires an altogether different treatment than dimensional modeling. The data warehouse which is the proper foundation for all DSS activity, including star joins, requires very granular, very flexible data. The ideal structure for the data warehouse is normalized data. The normalized data can be bent and shaped any old way.

 Different styles of data structures fit in different places.

Trying to superimpose a star join on the data warehouse implies that the data warehouse is going to be optimal for the support of one set of users at the expense of everyone else.

Not only does dimensional modeling not fit the data warehouse, dimensional modeling does not fit in many other places in the data warehouse/corporate information factory environment, such as:

- exploration warehouses,
- near-line storage,
- project warehouses and
- parts of the operational data store (ODS).

In short, dimensional modeling really only fits the data mart environment, where requirements for processing are known before the infrastructure is built.

The advocates of dimensional modeling are quite proud of their pioneering efforts and well they should be. They have rightfully advanced the industry in a proper direction. But in their zeal to convert people to their way of thinking, they seem to have forgotten that there are other valid approaches to solving the database design problem for the corporate information factory. The dimensional modeling approach fits in a few places in the DSS environment, but it does not fit everywhere. The dimensional modelers have crafted a fine hammer. Now everything looks like a nail to them.

For more information on related topics, visit the following channels:

- Business Intelligence (BI)
- DW Design, Methodology

©2008 Data Management Review and SourceMedia, Inc. All rights reserved. SourceMedia is an Investcorp company. Use, duplication, or sale of this service, or data contained herein, is strictly prohibited.