DIMENSIONAL MODELING BY USING A NEW RESPONSE TO SLOWLY CHANGING DIMENSIONS

LARS FRANK and CHRISTIAN FRANK, Department of Informatics, Copenhagen Business School, Howitzvej 60, DK-2000 Frederiksberg, Denmark. Phone: +45 38152400, Fax: +45 38152401, e-mail frank@cbs.dk.

CHRISTIAN S: JENSEN and TORBEN BACH PEDERSEN, Department of Computer Science, Aalborg University, Fredrik Bajers Vej 7E, 9220 Aalborg Ø, Denmark.

Abstract

Dimensions are defined as *dynamic* or *slowly changing* if the attributes or relationships of a dimension can be updated. Aggregations to dynamic dimensions might be misleading if the measures are aggregated without regarding the changes of the dimensions. Kimball et al. has described three classic solutions/responses to handling the aggregation problems caused by slowly changing dimensions. In this paper, we will describe a fourth solution. A special aspect of our new response is that it should be used before the other responses, as it will change the design of the data warehouse. Afterwards, it may be necessary to use the classic responses to improve the design further.

Keywords: Data Warehousing, Star schema, Dimension hierarchy, Drill functions, Slowly changing dimensions, Dynamic dimensions, OLAP.

1. Introduction

A *data warehouse* is an *OLAP* (*On Line Analytical Processing*) database [1 and 2], where the data is loaded/updated periodically. In other words, a data warehouse is not an *OLTP* (*On Line Transaction Processing*) database [3]. The data warehouse *drill functions* [4] have been developed to accommodate the special needs for aggregating the data stored in the fact table of a data warehouse.

The traditional *drill-down* functions use the one-to-many relationships of the data warehouse to find more detailed information. If we take accumulated data as an example, the drill-down function will show the more detailed data elements of the accumulated data. The *roll-up* function can use the one-to-many relationships of the data warehouse to generate an overview of the detailed information. However, the aggregating drill functions may give misleading results as old fact measures may be aggregated to dimension levels that have changed since the measures were created. The three classic techniques [4] for handling slowly changing dimensions have been described in the following way:

> *Type 1 response*. Overwrite the dimension record with the new values by which historic information is lost.
> *Type 2 response*. Create a new additional dimension record with the current information.
> *Type 3 response*. Create a “Previous” field in the dimension record to store the immediate previous attribute value.

In this paper, we will describe a new response to slowly changing dimensions, which transforms a dynamic hierarchical dimension in such a way that the dimension is split into different dimensions with reduced aggregation problems. In accordance with Kimball’s terminology, we will call this new solution the *Type 4 response* to handling slowly changing dimensions.

The Type 4 response described in this paper is to our knowledge the most important response to the aggregation problems that occur in dynamic dimension hierarchies. Therefore, the Type 4 response is often used in practice. However, at present the Type 4 response is used by instinct, as it has not yet become public knowledge. Anyway, we have also seen many solutions from inexperienced designers where the Type 2 response is used without criticism to solve all the problems of slowly changing dimensions.

The paper is organized as follows:

Section 2 will describe the most important concepts in dimensional modeling used in this paper. In section 3, we will describe the Type 4 response in details. Concluding remarks and suggestions for further research will be presented in section 4.

Related research:

Many authors working with data warehouse design have also analyzed the problems of aggregating fact measures to the levels of slowly changing dimensions e.g. [4 and 5]. In our view, Kimball [4 and 6] have analyzed and greatly improved the description of the problems and some of the solutions.

2. Dimensional Modeling

A *dimension hierarchy* [4] is a hierarchy of tables connected through one-to-many relationships towards the fact table. If the tables in a dimension hierarchy are joined together to a single dimension table, we say that the joined dimension has an *internal dimension hierarchy*. In practice, internal dimension hierarchies are often used as they normally improve the speed of executing aggregation at the costs of extra space for storing redundant information.

Normally, the fact table of a data warehouse has a *Time dimension hierarchy* that enables us to aggregate measures to the level of day, month or year. The time hierarchy may be internal or stored in separate tables as a dimension hierarchy. In the following example, the Time dimension has been designed with an internal dimension hierarchy for performance reasons. This will not produce aggregation problems, as the Time dimension including its hierarchies is static. However, the example also illustrates that both internal and external dimension hierarchies with dynamic dimension relationships automatically will have serious aggregation problems.

Example 2.1

In Figure 2.1, the central fact table of the snowflake schema has three dimensions and a dimension hierarchy. Therefore, the fact table has attributes for the four corresponding foreign dimension keys. In the Figure, the primary key of each table is underlined.

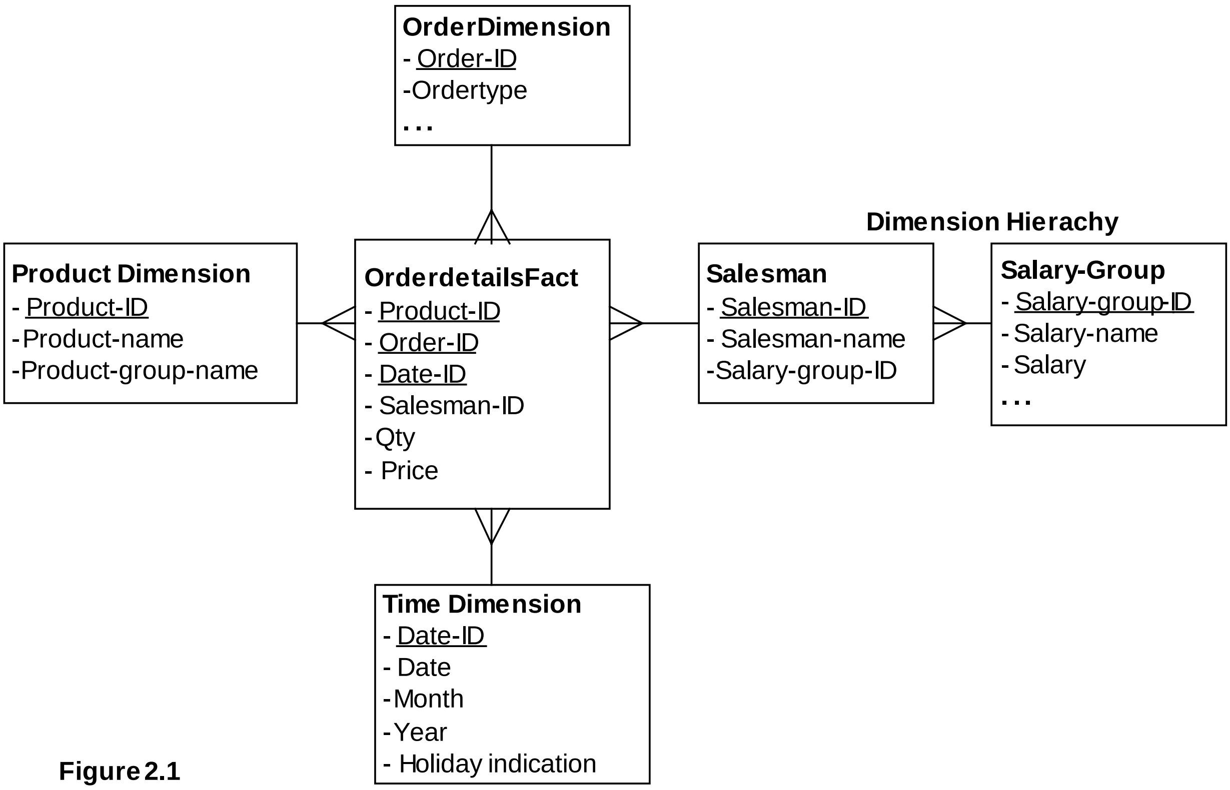

Figure 2.1

The Product dimension is dynamic as the Product-group of a Product may change. The Salary-group of the dimension hierarchy is dynamic as its relationship to a Salesman may change. The Salary-group dimension is stored in a hierarchy as this may save a lot of storage space if many salesmen are related to a few salary-groups with many attributes to describe the salary contracts. It may be interesting to aggregate the turnover (Qty*Price) to both the Salesman and Salary-group levels to evaluate each individual salesman as well as groups of salesmen. However, the aggregation to the Salary-group level is without meaning as some salesmen may have changed salary group. For these salesmen, the turnover that should have been aggregated to the old salary group is wrongly aggregated to the new salary group.

In Figure 2.1, the Product dimension has an internal dimension hierarchy as each Product has a relationship to one Product-group and each Product-group may have relationships to many Products. This relationship is dynamic, and, therefore, aggregation to the Product-group level in the internal Product hierarchy may be wrong. However, the Product-group is a dynamic classification criterion as one or more products may change there Product to Product-group relationship by a management decision. However, the same static Product-group definition should be used over time in aggregations even though the groups may have changed over time. Otherwise, it is without meaning from a semantic point of view to compare aggregations to the Product-group level over time. Therefore, a history destroying response should be used when data is aggregated from the Product to Product-group level.

The Time dimension is implemented with an internal hierarchy, too. This will give no aggregation problems as both the attributes of the dimension and the internal relationships between day, month, and year are static. However, in the real world, the Time dimension may be dynamic as some event may change a weekday to a holiday. Anyway, this may be handled as an error in the initial load of the Time dimension, and therefore the history destroying Type 1 response to slowly changing dimensions can be recommended.

3. The New Response Type 4 to Slowly Changing Dimensions

In this section, we will describe our new fourth response to slowly changing dimensions. We will recommend applying this response as the first as it reduces the complexity of the data warehouse.

The aggregation problems in internal and external dynamic dimension hierarchies disappear if the data corresponding to a dynamic entity in the hierarchy is removed and stored in a separate independent dimension directly related to the fact table. By doing so the dynamic data is either changed to static dimensions or changed to dynamic dimensions without aggregation problems. Response 4 may be used recursively as different layers of dynamic entities may occur in both internal and external dimension hierarchies.

Example 3.1
In Figure 3.1, the dynamic Salesmen dimension hierarchy from example 2.1 has been divided into two independent dimensions corresponding to the entities Salesman and Salary-group.

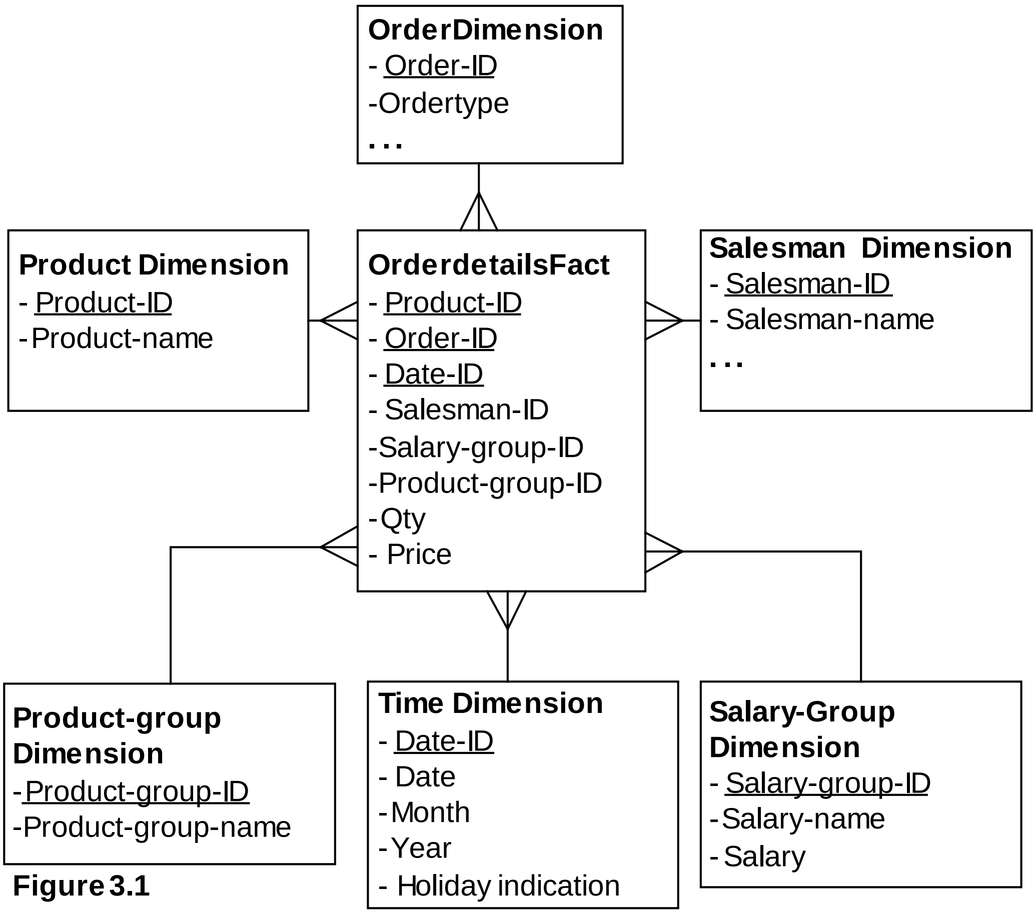

Figure 3.1

These dimensions are still dynamic, but they do not include hierarchies and therefore response 4 cannot be used any more. However, the transformation has solved all the aggregation problems as the turnover can be aggregated to the Salary-group level. Anyway, the dimensions are still dynamic as historic information may be lost when the dimensions are updated. Therefore, it may still be useful to use one of the other responses to slowly changing dimensions.

The dynamic internal dimension hierarchy of the Products dimension from example 2.1 has also been divided into two independent dimensions corresponding to the entities Product and Product-group. Also, these dimensions are still dynamic, but they do not include more hierarchies, and the aggregation problems have apparently been solved. Anyway, as described later, these dimensions may also benefit from other responses.

4. Conclusions and Suggestions for Further Research

To our knowledge, the new Type 4 response is the most important response to the aggregation problems that occur in dynamic dimension hierarchies. The Type 4 response is changing both internal and external dimension hierarchies to ordinary dimensions. Therefore, it should be used early in the design process where the major design decisions are taken.

For the time being, we are working with many new response types to slowly changing dimensions. However, the new response types have a rather complex interaction with the other response types, and, therefore, we have not finished our analyses. Anyway, we believe that the new Type 4 response is so important that it should be disclosed before we have finished our analyses and descriptions.

References

[1] Codd, E (1993), Providing OLAP (on line analytical processing) to user-analysts: an IT mandate, Technical Report, E.F. Codd and Associates.

[2] Codd, E. et al. (1993), Beyond Decision Support, Computerworld, 27:30, July.

[3] Gray, Jim and Andreas Reuter (1993), Transaction Processing, Morgan Kaufman.

[4] Kimball, R., Laura Reeves, Margy Ross and Warren Thornthwaite (1998), The Data Warehouse Lifecycle Toolkit: Expert Methods for Designing, Developing, and Deploying Data Warehouses, ISBN: 0-471-25547-5, John Wiley & Sons.

[5] Anahory, S. and Dennis Murray (1997), 'Data Warehousing in the Real World', Addison Wesley.

[6] Kimball, R. and Margy Ross (2002), The Data Warehouse Toolkit: The Complete Guide to Dimensional Modelling, 2nd Edition, ISBN: 0-471-26943-3, John Wiley & Sons.