

InfoCube Modeling – Dimension Design

Introduction:

This article illustrates InfoCube design aspects pertaining to the design of dimensions.

InfoCube Quality:

The quality of an InfoCube is generally measured with respect to

- Query Efficiency
- Load Efficiency
- Manageability

Following are the major modeling aspects that influence the load and query performance of an InfoCube:

- Design of Dimensions
- Partitioning (logical & physical)
- Data Compression (InfoCube level and Database level)
- Aggregates

“*Design of Dimensions*” is the focus of this article and we will examine the impact of the InfoCube design on load & query efficiency. The database specific information presented in this article is relevant to Oracle.

Extended Star Schema:

Behind every standard InfoCube there is a star schema which comprises of several database tables & indexes. When you create an InfoCube the BW system creates two fact tables (named E and F fact tables) and one DIM table for every dimension. A Line Item dimension is an exception because the system does not create a dimension table for Line Item Dimensions. The system also creates one unique index on the fact table with all dimensions, one non-unique index on the fact table with all dimensions and one separate index for every dimension. The Dimension tables connect the fact table with the master data tables.

Load Process:

When a record is loaded into the InfoCube the system has to make sure that the characteristic values for every characteristic exist in the system, otherwise a new SID has to be created. (It is possible to configure the system in such a way that it does not to create SIDs when the characteristic values do not exist in the master data.) System also has to check if the characteristic SID combinations already exist in the respective dimension tables, otherwise a new DIMID has to be created for the SID combination. A Line Item dimension is an exception to this process because it does not have a dimension table.

Read Process:

When a query is executed on an InfoCube, depending on the characteristics and the attributes being used in the query, the database joins the respective dimension tables, fact tables and the master data SID tables (attribute, text and hierarchy tables) to process the query. Conceptually, during a query read the reverse of load process happens.

Hash Memory:

The database uses Hash Memory area to join tables. Since most BW queries generate several joins between tables it is important to make sure that enough hash memory is allocated. Typically, a BW system requires more Hash memory than the R/3 system.

Based on the load process described earlier, it is clear that every additional characteristic or a dimension incorporated into the InfoCube is a burden on the system because of the DIMIDs and the SIDs that are to be created or located for reuse. Hence, it is highly recommended to include

in the InfoCube only the characteristics that will really be used in the queries. However, if you would like to incorporate additional characteristics that will potentially be used for future reporting, it would be beneficial to house this data in an ODS.

Characteristics vs. Navigational Attributes:

If reporting has to be done on an attribute of a characteristic it can be modeled in two ways:

1. To load the attribute of the characteristic directly into the InfoCube
2. To use the navigational attribute of the characteristic in the report

Note: A navigational attribute is an attribute on which slicing and dicing can be done in the report.

Each option has its pros and cons. In the first case, you are spending more time while loading data to lookup the master data to identify the attribute value. In the second case, you are spending more time while running the query to do a join between Dimension Tables and Master data tables.

Line Item Dimensions:

Suppose that you have to build a Sales report on an InfoCube which has to display the Document number. In this case the size of the dimension which contains the Sales Document number is comparable to the size of the Fact table. Since the system joins the Fact Table with the Dimension table at query run time, if both the fact and dimension tables are large the query runtime will be very high. In such scenario, it is better to create a Line Item dimension for Document number. A Line Item dimension can have only one characteristic. The rule of thumb is to create a line item dimension if the cardinality of the characteristic is greater than 10% of the fact table size. However, this is not a necessary condition, meaning that you can create line item dimensions with lower cardinality characteristics as well. If the total number of characteristics in the InfoCube is always going to be less than or equal to 13 then creating 13 line item dimensions might be optimal.

In conclusion, creating optimal dimensions is specific to each scenario though there exist some rules of thumb.

High Cardinality Dimensions:

Bitmap indexes are efficient for low cardinality fields like Gender, whereas B-Tree indexes are efficient for high cardinality fields like document number. By default, the type of the index on the dimension tables and fact tables is Bitmap. However, if the cardinality of the dimension is high you can check the High Cardinality Dimension checkbox for the dimension so that the system will use B-Tree index. Note that High Cardinality can only be checked if the dimension is a line item dimension.

Based on the Read Process described above it is important to minimize the size of the dimension tables. However, it is wrong to increase the number of dimensions just for the sake of decreasing the size of the dimensions. If the total number of characteristics is greater than 13, then you need to achieve a good balance between the size of the dimensions and the number of dimensions by grouping related characteristics in the same dimension. For example, customer & customer group are related characteristics.

In summary,

- Do not include a characteristic in an InfoCube that will never be used in any query on that InfoCube.
- Between using a navigational attribute vs. a characteristic, the first option is better for load performance whereas the second option is better for query performance.
- Minimize the size of the dimension tables
- Minimize the number of dimensions, but it is more important to minimize dimension sizes

- If more than 13 characteristics are included in the InfoCube, group related characteristics into a dimension to achieve a good balance between the sizes of the dimensions vs. the number of dimensions
- For high cardinality characteristics use Line Item dimensions
- For high cardinality dimensions use B-Tree index type
- If the number of characteristics will always be less than or equal to 13, create one dimension for each characteristic and mark each dimension as a Line Item Dimension.

Disclaimer

Erada makes no representations about the suitability of the Information contained in the documents for any purpose. All such documents and related graphics are provided to you “AS IS” without warranty of any kind. All sample code is provided by Erada for illustrative purposes only. These examples have not been thoroughly tested under all conditions. Erada, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

SAP and mySAP.com are trademarks or registered trademarks of SAP AG. Other SAP Product or Service names mentioned herein are registered or unregistered trademarks of SAP AG.

©2001 Erada Systems, Inc. All rights reserved.