



Guideline

Setting Up Dimensional Information in Framework Manager

Product(s): IBM Cognos ReportNet 1.1

Area of Interest: Modeling

Copyright

Copyright © 2008 Cognos ULC (formerly Cognos Incorporated). Cognos ULC is an IBM Company. While every attempt has been made to ensure that the information in this document is accurate and complete, some typographical errors or technical inaccuracies may exist. Cognos does not accept responsibility for any kind of loss resulting from the use of information contained in this document. This document shows the publication date. The information contained in this document is subject to change without notice. Any improvements or changes to the information contained in this document will be documented in subsequent editions. This document contains proprietary information of Cognos. All rights are reserved. No part of this document may be copied, photocopied, reproduced, stored in a retrieval system, transmitted in any form or by any means, or translated into another language without the prior written consent of Cognos. Cognos and the Cognos logo are trademarks of Cognos ULC (formerly Cognos Incorporated) in the United States and/or other countries. IBM and the IBM logo are trademarks of International Business Machines Corporation in the United States, or other countries, or both. All other names are trademarks or registered trademarks of their respective companies. Information about Cognos products can be found at **www.cognos.com**

This document is maintained by the Best Practices, Product and Technology team. You can send comments, suggestions, and additions to cscogpp@ca.ibm.com .

Contents

1 DIMENSIONS 4

1.1 UNDERSTAND THE DIMENSIONALITY OF YOUR DATA. 4

1.2 IDENTIFY THE LEVELS, LEVEL KEYS, LEVEL ATTRIBUTES AND HIERARCHY(IES) OF YOUR DIMENSION. 4

2 DIMENSIONAL INFORMATION AND SQL GENERATION 5

3 OVERVIEW OF DIMENSIONAL INFORMATION COMPONENTS AND THEIR IMPACT ON REPORTNET 6

4 SPECIFY DIMENSIONAL INFORMATION IN FRAMEWORK MANAGER..... 7

4.1 PROVEN PRACTICES TO FOLLOW 8

1 Dimensions

Dimensions are used in the following way for ReportNet.

1. Associate measures with levels to ensure proper aggregation
 - a. For example, to avoid double counting in queries
2. Define keys for groups for query efficiency
3. Model aggregate tables
4. Model multi-grain aggregate tables

Note that dimensions do not appear in the Report Studio and Query Studio user interfaces as they are intended for query planning and not for user exposure.

Dimensions are optionally defined on each query subject and use only the query items defined therein.

1.1 Understand the dimensionality of your data.

When specifying dimensional information in FM it is important to understand the following.

- What are the levels in your dimension?
- What is the order and combination of levels that form hierarchies.
- How do you uniquely identify the members of a level, is it unique (one query item) or does it require multiple query items?
- Association of levels with attributes and measures.
- At what level of granularity is your fact data, do you have more than one level of granularity? Ie. Is some of your data recorded monthly and some daily, this would mean that you would have a fact table relating to your time dimension table at the month and day levels.
- Are there surrogate keys defined in your data source?

1.2 Identify the levels, level keys, level attributes and hierarchy(ies) of your dimension.

A star schema might have a table that looks like:

PRODUCT_KEY	PRODUCT_NUMBER	PRODUCT_NAME	PRODUCT_TYPE	PRODUCT_LINE	PRODUCT_IMAGE
8	8	TrailChef Double Flame	Cooking Gear	Camping Equipment	P08CE1CG1
9	9	TrailChef Kettle	Cooking Gear	Camping Equipment	P09CE1CG1
10	10	TrailChef Utensils	Cooking Gear	Camping Equipment	P10CE1CG1
11	11	Star Lite	Tents	Camping Equipment	P11CE1TN2
12	12	Star Dome	Tents	Camping Equipment	P12CE1TN2

It can be observed from this data that product lines have multiple product types and product types have one or more products. The cardinality of the product line-product type relationship and the product type-product relationship should be verified. A hierarchy that might be useful for reporting would have the levels Product Line, Product Type and Products and would be structured so that Products relate to and are children of Product Types, and Product Types relate to and are children of Product Lines. Once it is clear what the levels are and what the relationships are between the levels it is important to determine what uniquely identifies a level (the key) and which data elements are associated to each level. In this case Product_Line is a unique identifier for the Product Line level, Product_Type is a unique identifier for the Product Type level and Product_Key is a unique identifier for the Product level with Product_Number and Product_Image as attributes. You may wonder why Product Key was chosen as the identifier for the Product level instead of Product Number. The answer is that the datasource is a warehouse that uses surrogate keys and Product Key is the surrogate key for the Product dimension table. Product Number is also an identifier but it is regarded as a business key and will be treated as an attribute when specifying the dimensional information in Framework Manager. The dimensional information is summarized in the following table:

Levels	Product Line	Product Type	Product Name
Keys	Product Line	Product Type	Product Key
Unique	yes	yes	yes
Attributes	N/A	N/A	Product Number, Product Image
Hierarchy	Product Line	Product Type	Product Name

2 Dimensional Information and SQL Generation

It is important when specifying your dimensional information to understand the affect this will have on the SQL that is generated. Dimensional information affects grouping and aggregation of data, not only from the dimension data source but also from other data sources that have relationships with that dimension. Specifying hierarchies for a query subject using query items as levels, either individual or combinations of query items as keys, and associating query items to the appropriate levels as attributes provides FM with enough information to automatically group and aggregate data according to the hierarchies you have specified. Hierarchies in this case can be thought of as ways of specifying the order in which data segmented into levels can be grouped.

With the dimensional information specified in the model it will no longer be necessary for a report writer to understand the relationships between the data elements and do explicit grouping in a report in order to get correctly aggregated data. The model will have sufficient information to write the correct SQL so that data joined on Product_Key can be aggregated properly when reported by Product Line or Product Type or both.

3 Overview of Dimensional Information Components and their impact on ReportNet

The key is optional and used to identify a query item(s) that guarantee uniqueness. By default the query item that is the level is set as the key. A single query item key is used when one item uniquely identifies the level - for example, product number uniquely identifies each product. . If a single query item determines uniqueness, then use the Unique Key option to inform ReportNet.

Multiple part keys are used when two or more query items are required to uniquely identify the level - for example, the combination of country and city are required to uniquely identify each city. Note that each level must be uniquely identified on its own, not by other levels within a hierarchy.

Facts are associated with levels to tell ReportNet which quantitative query items can be aggregated against that level. For example, if a query subject had the country and city query items and a population column, ReportNet would need to know whether the population was for the country or the city. Where facts are stored at different grains (for different levels) in the same table, ReportNet needs to know which facts are associated with each level. For example, a query subject with country and city might have two population query items one for country and one for city. By properly associating the fact with the appropriate level in Framework Manager, you ensure that ReportNet will correctly aggregate the facts avoiding such issues as double counting.

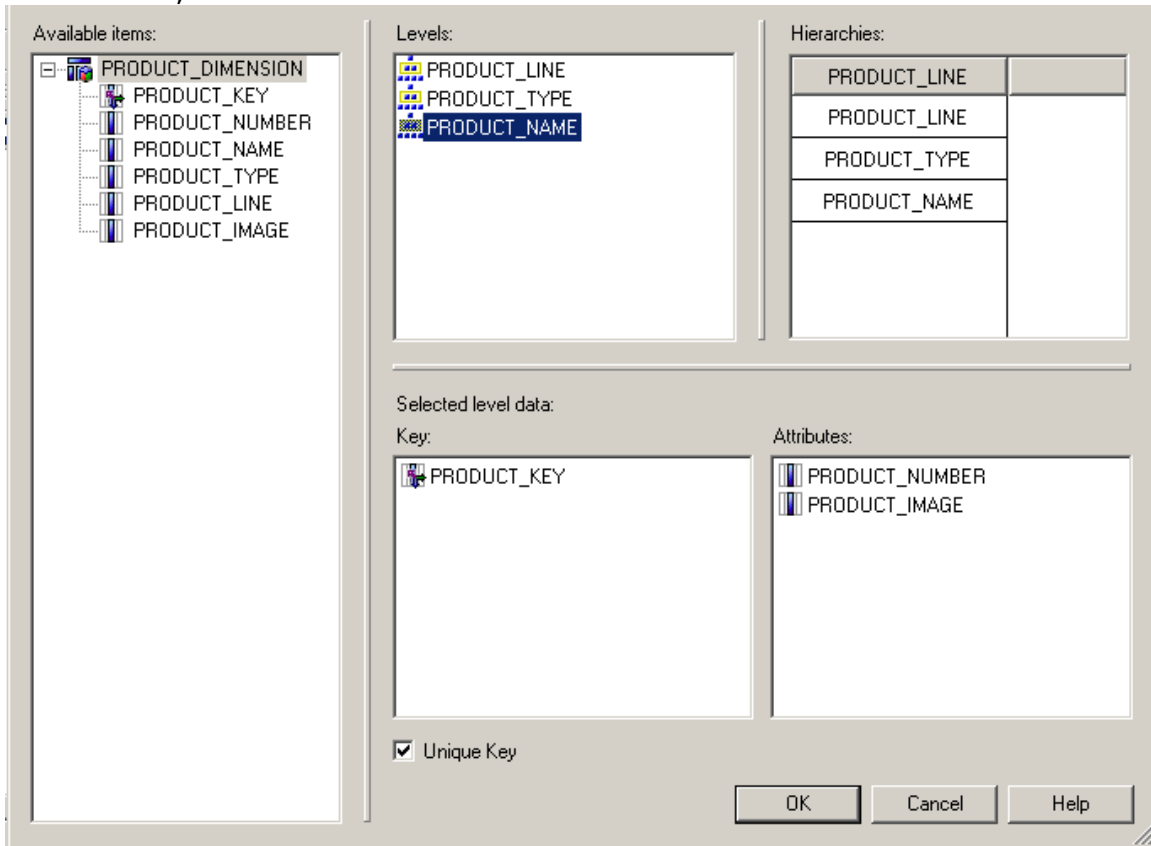
Attributes are qualitative columns which share a 1 to 1 relationship with the level. For example, a product description exists for each product. Attributes are columns which are generally not aggregated.

Once levels have been defined they can optionally be used in hierarchies. ReportNet will attempt to derive hierarchies if they are not set. The hierarchies establish the relationship between the levels - for example, a hierarchy of country - city tells ReportNet that country is a higher level grouping to city.

4 Specify Dimensional Information in Framework Manager

Before Specifying Dimensional Information in FM ensure that you have verified your relationships, query item usages and aggregation properties.

In FM, dimensional information described above would look like this.



4.1 Proven Practices to follow

Below are some proven practices to follow when Specifying Dimensional Information in Framework Manager

1. Ensure that keys and other identifiers have their usage specified as identifier.
2. When a surrogate key is available for a level use it as the level key and treat the business key as an attribute.
3. Use the item you would use in reporting as the level and the id (forms one end of a relationship) as the key for the level (remember to reflect this in the prompt info for the sake of query efficiency).
4. For each level associate any query items whose granularity is specific to that level.