

Implementing a SAS Data Warehouse Considering Star, Fact Constellation, and Snowflake Schemata

Maria Lupetin,
InfoMaker Inc., Glenview, Illinois

Abstract

Data warehousing, data mart, and data mining nomenclature have lacked definitive explanations of what applies when. The key analysis question is what approach is to be taken when considering business intelligence needs and the data warehouse method. We should not be "dazzled" by the approach most popular today.

This work explores a number of architectures for the data warehouse structure. These, more often than not, are driven by the functionality of the software being used. Thus, a general discussion of what works with SAS will unfold.

An implementation of a data mart for an outpatient clinical information system will be presented. The star schema, fact constellation, and snowflake methods will be contrasted.

One will conclude that with any emerging technology, the distinctions between these approaches are often blurred or overlapping. Consequently, it is a challenge to choose the one which best fits the analysis question at hand, as well as, which one fits into the structure of SAS itself.

The Goal of This Paper:

The purpose of this paper is to introduce the reader to data warehousing concepts and terms. It will briefly define concepts such as OLTP, OLAP, enterprise-wide data warehouse, data marts, dimensional models, fact tables, dimension table, star join schema, fact constellation schema and snowflake schema.

The present study will also explore the implementation of a data mart for an outpatient clinical information system using the star schema

We will examine the challenges to choose the schema which best fits the analysis question at hand, as well as one which fits into the structure of SAS itself.

After reviewing the concepts and approaches, one will conclude that the SAS family of products offers an end to end solution for data warehousing.

Why Implement a Data Warehouse or Data Mart?

In the past, a common system application has been of the OLTP kind: *On-line Transaction Processing*. Examples of OLTP systems are accounting, payroll, and billing systems where the lowest level of information is a transaction. These OLTP systems are at the heart of running an operation or department of a firm. In my belief, many of these basic operational needs have been satisfied by OLTP systems in place.

These systems contain vast amounts of data which can be difficult to access. However, this data is still critical for decision makers to better manage a firm. In many cases the data is spread across many disparate OLTP systems and is hard or impossible to access with one user friendly interface. This does not eliminate the need to have this data consistent, accessible and useable by the decision makers. Hence the formalization of data warehouse and data marts has occurred. The reader of is paper will be aware of this need because we have been providing information for decision makers using SAS for years.

A data warehouse is a data base that contains data that has been cleansed for quality and organized for quick querying of information by many different views of the organization. Data warehouses are subject oriented (i.e., customer, vendor, product, activity, patient) rather than functionally oriented, such as the production planning system or human resources system. Data warehouses are integrated; hence the meaning and results of the information is same regardless of organizational source. The data is nonvolatile but can change based upon history. The data is always the same or history changes based on today's definitions. Contrast this to a database used for an OLTP system where the database records are continually updated, deleted, and inserted.

The data is consistence across the enterprise, regardless how the data is examined, "sliced and diced." For example, sales departments will say they have sold 10 million dollars of widgets across all sales regions last year. That volume will be confirmed by the finance department when they access the data warehouse and determined that indeed when the twelves months of last year's national sales are totaled, the result is 10 million

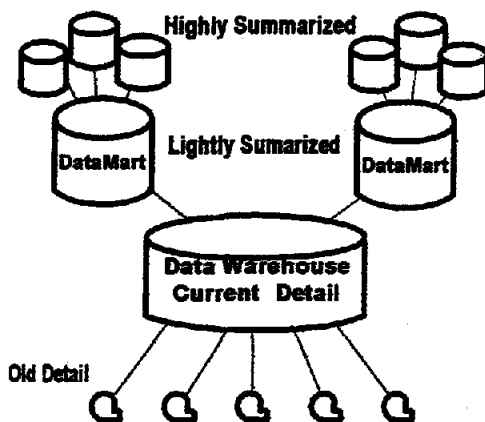
dollars of widgets. The two perspectives and queries render the same result.

Data Warehouse vs Data Mart

An enterprise-wide data warehouse is meant to support the informational needs of the entire enterprise. These data warehouses will contain both a very detailed level of data and summarized data to serve decision makers. Several years of history will typically be stored in the data warehouse. It is not unusual to have at least ten years of granular data stored with its many perspectives such as geography, date, style, salesperson, etc.

Many times, development of such a reservoir of data is deemed too formidable, or not seen as a universal need throughout the organization. A data warehouse for a Fortune 200 firm can be several gigabytes to several terabytes in size. In those cases, many firms opt to develop data marts instead.

A data mart is in structure and purpose similar to a data warehouse, except its audience is limited to a departmental or specific subject need. For example, InfoMaker Inc. recently developed a data mart for the sales department of a Fortune 100 firm. The subject was limited to seven years of history of order and sales data at both an individual order and aggregate level.



Data Warehousing Concepts

There are several ways a data warehouse or data mart can be structured: multidimensional, star, fact constellation, and snowflake. However, an underlying concept used by all the models method above is that of a dimension. A dimension is the different ways the information can be

“cut” and summarized such as geography, time intervals, product groups, salesperson(s).

Common to the star, fact constellation, and snowflake methods is the fact tables. The *fact table* is a database table that contains the data (factual history) like sales volume, cost or quantity for usually for the smallest time period or a low level of all the dimensions.

This low level of data is call *granular data* or *atomic data*. It is not aggregated at all. An example of a record in a fact table for an outpatient clinical information system is information on a single event such as a medical procedure given to a patient on a given day at a specified charge.

For an outpatient clinical information system a simple fact table would have the following columns (SAS variables).

PATIENT: unique patient identification
LOCATION: where the procedure was performed
SERVEMD: doctor who performed the procedure
PROVIDER: the insurance company who pays the bill
PROCEDUR: standard CPT procedure code
DIAGNOS1: primary diagnosis as an ICD9 code
DIAGNOS2: secondary diagnosis as an ICD9 code
DATESERV: date of the procedure was performed
FULLPRIC: the “list price” of the procedure”
DISCOUNT: the discount given to provider or patient
ACTBILL: the final actual charge for the procedure
AGE: age of patient at time of procedure
COUNT: the frequency of the event

The first eight variables from PATIENT to DATESERV are dimension variables. The final five variables, FULLPRIC, DISCOUNT, ACTBILL, AGE and COUNT are numerical variables to be used for arithmetical or statistical operations.

In addition to the fact tables, there are also dimension tables in the database. These *dimension tables* describe the options to “cut” or view the data in the fact table. The star, fact constellation and snowflake schemata all use more than one dimension table in their database. The records in a single dimension table represent the levels or choices of aggregation for the given dimension. The classic example given is the for the sales dimension of a corporation. The records in the dimension table will indicate that the fact table data can be aggregated by salesperson, sales districts, sales regions, domestic, international, etc. Another dimension would be date. Using the date dimension we would be able to analyze

Using the date dimension we would be able to analyze data by a single date or dates aggregated by month, quarter, fiscal year, calendar year, holidays, etc.

On-line Analytical Processing (OLAP) is the analytical capabilities provided by the data warehouse or data mart. One can view granular data or various aggregations of data for business analyses using graphical-user-friendly tools.

In our outpatient clinical information example the individual dimension table would capture views by:

- Patient (gender and age)
- Location of service (office and ownership)
- Doctor performing the procedure (name and affiliation)
- Insurance Provider (name, type)
- Procedure performed (CPT code and groupings)
- Diagnoses (ICD9 code and groupings)
- Date Procedure was performed (date, month, etc.)

Star Join Schema

The star join schema (also known as the *star schema*) is a database in which there is a single fact table and many dimension tables. These tables are not normalized. They are unlike traditional operational data bases where one attempts to normalize the tables. In the fact table there is one segment (a column, a SAS variable) for each dimension. The fact table uses a compound key made up of the group of the dimensions. In addition, the fact table usually contains additional variables which typically are additive numbers, i.e., numeric facts. In SAS terms, the dimension are like the "classes" in Proc Summary, and the facts in the fact table are the "variables" in Proc Summary.

Each dimension is represented as a single table and is not normalized. These dimension tables are joined to the fact table to create the user's selected views of the data.

Within a dimension table, there is a "level" column that indicates the nesting of hierarchies, such as (Level 1 =

date, Level 2= month, Level 3 = quarter). This level column must be carried and always defined.

Finally and most importantly, the same fact table must contain some or all combinations of aggregated data usually needed by the decision maker.

For the full star schema of our outpatient clinical information system, see Exhibit 1 at the end of this paper.

Fact Constellation Schema

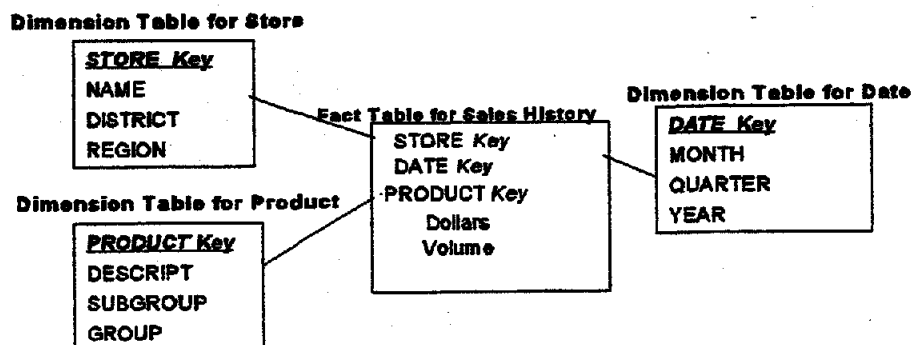
The *fact constellation* schema for a data warehouse contains many fact tables and many dimension tables. These fact tables do not contain atomic data, but data aggregated to the "Level = 2". There are no records from other higher or lower level of aggregation. Thus in our sales example, the one fact table would have the data aggregated to the sales district. Another fact table in the fact constellation would have the data aggregated to the monthly level. Similarly, in an outpatient clinical information system, one fact table would be aggregated to the type of insurance provider rather than the individual provider. There would be eight fact tables in the constellation representing the level 2 aggregate for each of the eight dimensions in the data warehouse. See Exhibit 2 for depiction of a fact constellation.

Snowflake Schema

A data warehouse using the snowflake schema is a database that also contains many more fact table and dimension tables than the fact constellation. However, in the *snowflake schema*, the dimension tables are normalized by attribute level for each dimension. Each normalized dimension table points to a unique aggregated fact table.

In the clinical example, we would have a fact table having the lowest level of data, plus a fact table aggregated for each level of time hierarchy and its own accompanying dimension table. A picture of a snowflake schema is shown at the end of this paper as Exhibit 3.

Star Schema



The Pros and Cons of the Various Schema

The star schema is the simplest in concept, yet it has several drawbacks:

- the compound dimension keys add complexity and make processing slow;
- concatenated keys don't handle nulls well; and
- one has to remember the meaning of all the level indicators when accessing the aggregated data in the same table.

There are ways to overcome the compound key problem. One solution would be to create a unique key. This approach is discussed by Michael A. Raithe's SUGI 22 Paper 113, pages 514-518.

The fact constellation, with only its level 2 aggregates, is only practical for data marts. By itself it is not a true data warehouse. One may only use this model in selected applications, perhaps for departmental OLAP needs. Since many fact tables are needed, there is a lot more management overhead for this structure. For obvious reasons, the fact constellation is not commonly used.

The snowflake schema eliminates the level coding need, but the trade off is that many fact and dimension tables must also be maintained. The complexity of having many tables each with unique dimension tables does not lend itself well to a SAS implementation both in comprehension of the data available and structure to the user, and the maintenance of all the tables.

Overall, I recommend a hybrid approach when using SAS as the data warehouse tool. First, an atomic fact table with as few dimension tables as possible should be used to minimize multiple joins. This can be accomplished by using some of the dimension information as part of the "factual data" and not executing any arithmetic operations on those fields. For example, in the clinical information system, the secondary diagnosis field, DIAGNOS2 would be a variable containing the ICD9 code and another two variables would also exist in the same record. These additional fields in the fact table would be the secondary diagnosis group and subgroup. Similarly, the doctor's code would exist along with another variable for his/her affiliation in the atomic level fact table. Using this approach would create a great deal of redundant data in the fact tables. Our goal in this design is ease of access and speed of query, rather than saving on storage space.

I also suggest, aggregate fact tables be constructed selectively. Not all possible aggregates should be created. Known commonly requested aggregations would have

their own aggregated fact tables. The group of these aggregated fact table and their related dimension tables would form a data mart. The data mart would provide answers to common questions such as how many patients with angina did we treat last year? What were the most frequent diagnoses we made last month? How much did we bill Medicare for sleep apnea patients for each quarter this year?

What About Data Analysis, OLAP, Data Mining

Data warehouse and data marts exist to answer questions and find business opportunities. There are many ways to analyze data using SAS procedures such as Proc Freq, Proc Summary, Proc Univariate, Proc Means, Proc Tabulate for simple statistics, frequencies and summaries. For the more sophisticated user, SAS/STAT provides a wealth of statistical procedures to analyze the data from the data warehouse, including multiple regression, logistic regression and time series.

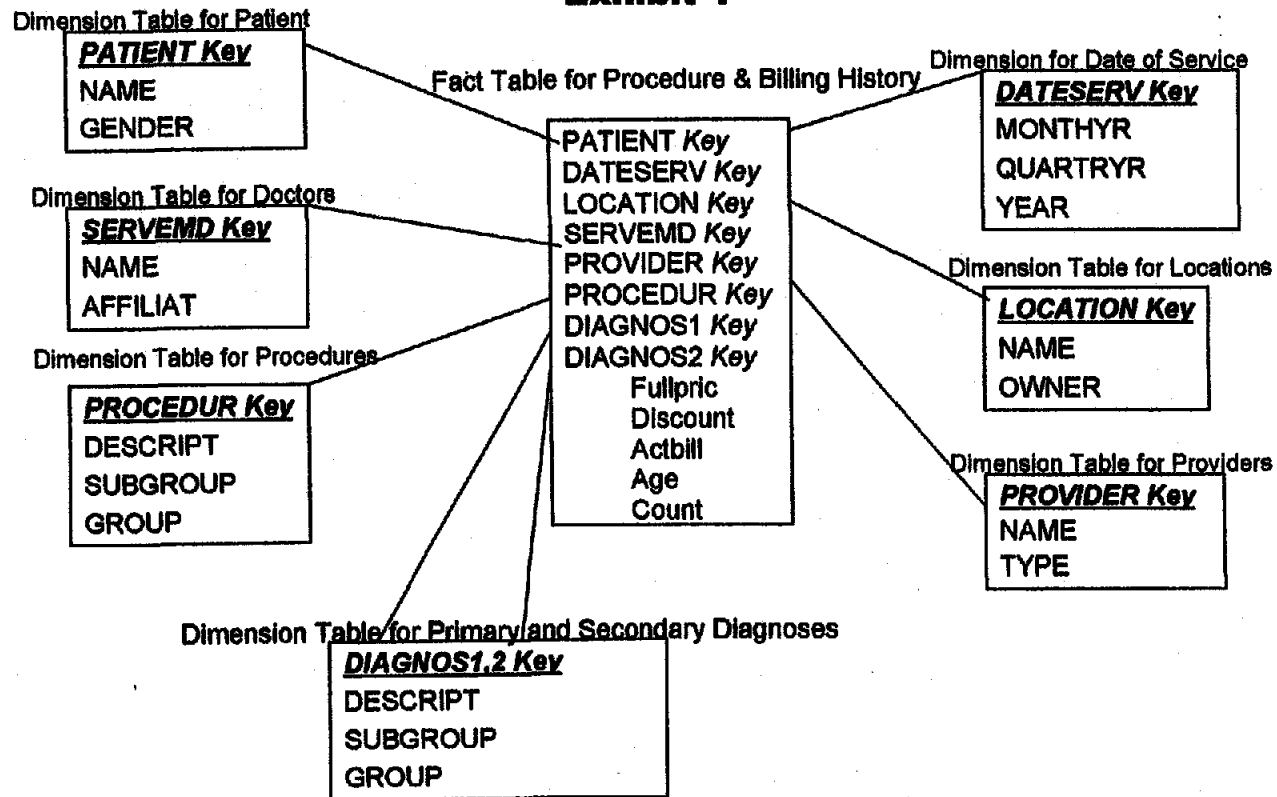
Finally, *data mining* is the name given to newer statistical techniques used to explore voluminous data stores. These techniques include decision trees and neural networks. These methods, like neural networks, can sometimes handle collinearity better than the older statistical techniques. The SAS Institute is offering a new module that will contain these techniques. It will be available in 1997 as an add on to the SAS/STAT module.

Why SAS Is Such a Natural Choice for Data Marts

The most common implementation today across "Corporate America" of the data warehousing approach is the development and use of data marts. The SAS Institute provides software to create tables; clean, load, aggregate, maintain, query data; and data mine these data marts. Just think of the power we SAS developers, analyst, and users have had at our finger tips all these years!

SAS Procedures like Download, Copy, SQL, APPEND, FSEDIT, Format, and SUMMARY are only a few of the many tools available to build a data mart. Recently, the SAS Institute introduced new products such as the SAS/MDDB Server and SAS/Warehouse Administrator. The SAS/MDDB Server is used for the development of a multidimensional data warehouses and data marts. SAS/Warehouse Administrator is used to create and maintain table based data warehouses and data marts. The SAS Institute provides an integrated end to end solution for data warehousing.

Star Schema for Outpatient Clinical Information System Exhibit 1



Fact Constellation with Selected Tables

Exhibit 2

Patient Gender Fact Table

GENDER Key
 DATESERV Key
 LOCATION Key
 SERVEMD Key
 PROVIDER Key
 PROCEDUR Key
 DIAGNOS1 Key
 DIAGNOS2 Key
 Fullpric
 Discount
 Actbill
 Age
 Count

Owner of Location Fact Table

PATIENT Key
 DATESERV Key
 OWNER Key
 SERVEMD Key
 PROVIDER Key
 PROCEDUR Key
 DIAGNOS1 Key
 DIAGNOS2 Key
 Fullpric
 Discount
 Actbill
 Age
 Count

Type of Provider Fact Table

PATIENT Key
 DATESERV Key
 LOCATION Key
 SERVEMD Key
 TYPE Key
 PROCEDUR Key
 DIAGNOS1 Key
 DIAGNOS2 Key
 Fullpric
 Discount
 Actbill
 Age
 Count

Primary Diagnosis Sub-Group Fact Table

PATIENT Key
 DATESERV Key
 LOCATION Key
 SERVEMD Key
 PROVIDER Key
 PROCEDUR Key
 SUBGROUP Key
 DIAGNOS2 Key
 Fullpric
 Discount
 Actbill
 Age
 Count

Month Year Fact Table

PATIENT Key
 MONTHYR Key
 LOCATION Key
 SERVEMD Key
 PROVIDER Key
 PROCEDUR Key
 DIAGNOS1 Key
 DIAGNOS2 Key
 Fullpric
 Discount
 Actbill
 Age
 Count

Doctor Affiliation Fact Table

PATIENT Key
 DATESERV Key
 LOCATION Key
 AFFILIAT Key
 PROVIDER Key
 PROCEDUR Key
 DIAGNOS1 Key
 DIAGNOS2 Key
 Fullpric
 Discount
 Actbill
 Age
 Count

Procedure Sub-Group Fact Table

PATIENT Key
 DATESERV Key
 LOCATION Key
 SERVEMD Key
 PROVIDER Key
 SUBGROUP Key
 DIAGNOS1 Key
 DIAGNOS2 Key
 Fullpric
 Discount
 Actbill
 Age
 Count

Secondary Diagnosis Sub-Group Fact Table

PATIENT Key
 DATESERV Key
 LOCATION Key
 SERVER Key
 PROVIDER Key
 PROCEDUR Key
 DIAGNOS1 Key
 SUBGROUP Key
 Fullpric
 Discount
 Actbill
 Age
 Count

Snowflake Schema for Selected Tables Exhibit 3

DATE DIMENSIONS TABLE

<u>DATESERV Key</u>
MONTHYR
QUARTRYR
YEAR

MONTH DIMENSIONS TABLE

<u>MONTHYR Key</u>
QUARTYR
YEAR

QUARTER DIMENSIONS TABLE

<u>QUARTRYR Key</u>
YEAR

DAILY FACT TABLE

PATIENT Key
DATESERV Key
LOCATION Key
SERVEMD Key
PROVIDER Key
PROCEDUR Key
DIAGNOS1 Key
DIAGNOS2 Key
Fullpric
Discount
Actbill
Age
Count

MONTH FACT TABLE

PATIENT Key
MONTHYR Key
LOCATION Key
SERVEMD Key
PROVIDER Key
PROCEDUR Key
DIAGNOS1 Key
DIAGNOS2 Key
Fullpric
Discount
Actbill
Age
Count

QUARTER FACT TABLE

PATIENT Key
QUARTRYR Key
LOCATION Key
SERVEMD Key
PROVIDER Key
PROCEDUR Key
DIAGNOS1 Key
DIAGNOS2 Key
Fullpric
Discount
Actbill
Age
Count

Why Data Warehouses Are Here to Stay
Simple, we need the answers now, accurate, and consistent to run our businesses.

About the Author

Maria Lupetin is president of InfoMaker Inc., Glenview, Illinois. She has over twenty years of business experience in information technology and applied mathematics. Her background spans manufacturing, distribution, marketing, and healthcare applications assisting many Fortune 200 companies.

Ms. Lupetin is the founder of InfoMaker Inc., a consulting firm specializing in decision support systems and data warehousing. InfoMaker is a SAS Quality Partner, and partner with the Oracle and Informix companies. Before founding InfoMaker, Maria worked at Morton International and Union Pacific Railroad.

Maria has an MBA from the University of Chicago and a M.S. in Operations Research from the University of Minnesota.

Ms. Lupetin can be contacted at:

InfoMaker Inc.
950 Milwaukee Avenue
Glenview, Illinois 60025
Phone: (847)390-6660
Fax: (847)390-6774
e-mail: lupetin@infomaker.com
<http://www.infomaker.com>



Note: Many thanks to Larry Greenfield, LGI Systems Inc., Northfield, IL, whose marvelous web page, "The Data Warehousing Information Center," was a starting point for this research.

References

Archer Decisions Science, Inc., (1995-1996),
"The 'Fact Constellation' Schema,"
http://www.strategy.com/dwf/raden/str101_1.html.

Greenfield, Larry, LGI Systems Inc., (1997),
"The Data Warehousing Information Center,"
<http://pwp.starnetinc.com/arryg/index.html>.

Inmon, W. H., (1996), *Building the Data Warehouse*,
Second Edition, John Wiley & Sons, Inc.

Kimball, Ralph, (1996), *The Data Warehouse Toolkit: Practical Techniques for Building Dimensional Data Warehouses*, John Wiley & Sons, Inc.

Raithel, Michael A. (1997), "Summarizing Impossibly Large SAS Data Sets for the Data Warehouse Server Using Horizontal Summarization," *Proceedings of the 22nd Annual SAS Users Group International Conference*, SUGI 22 Paper 113, pages 514-518.

SAS Institute Inc. (1996), "Implementing a Data Warehouse Using SAS® Software Course Notes," Cary, NC., SAS Institute Inc.

Trademarks or Registered Trademarks:

SAS, SAS/STAT, SAS/Warehouse Administrator, and SAS/MDDB Server.