

Paper 105-29

Meta-Data for End-Users

A crucial part of your Data Warehouse quality policy

Gad Kotler, Executive Information Systems Ltd., Herzlia, ISRAEL

ABSTRACT

Many organizations has invested a lot of efforts implementing Data Warehouse projects using tools to Extract, Load and Transform data into meaningful information for end users across the organization. While the quality of the information is as good as the quality of the data, exploiting this data by End-users is as good as the quality of the Meta-Data.

Generally speaking, Meta-Data should help users understand the complexity of the data and how it should be used in order to transform it into information (Reports).

Experience with many End-Users shows that the biggest obstacle for exploiting the huge investments in Data Warehouse infrastructure is this usually “forgotten” unimplemented stage of the project.

The term “Meta-Data” has various implementations with different focuses and there is no “One Version” of what should be the scope of a Meta-Data repository and how it should be presented to End-Users.

The paper will discuss our vision and experience with SAS based Data Warehouses regarding Meta-Data issues. Specific details for SAS End-Users will be highlighted and a Unique SAS based Meta-Data repository used in our SIGMA WAREHOUSE solution will be presented.

SCOPE OF META-DATA

While trying to determine what will be the scope of information in the Meta-Data repository, It is necessary to make a distinction between two different needs for Meta-Data:

META-DATA FOR DATA WAREHOUSE ADMINISTRATORS

The ETL (Extract-Transform-Load) process done in every Data Warehouse project is a major supplier of information needed by administrators. Information about source and target tables, format conversions, user exits code, calculations etc is of a main interest for

them. This kind of information is of a little interest for End-Users.

META-DATA FOR END USERS

End-Users want to know two major piece of information:

1. Where this SAS variable came from – The screen number (or name) from the operational system where they can see the same values. (Note that the term “screen” does not necessarily mean “source table” since one screen can have many source tables behind it in the operational system).
2. Business information related to this variable. E.g. free format explanation about the role of this variable, it’s correctness and suggestions about using it in reports.

META-DATA USAGE

Meta-Data must be incorporated into a report’s writing tool in order to assist the user in the process of data analysis and reports writing.

There are two points where the end-user will need assistance:

Where clause – writing a correct and efficient where clause in SAS can be a very tricky job which needs a lot of SAS experience. Meta-Data, which is integrated in this process, can ease the pain.

Variable Selection – Which variable holds the values that I need for the report. Meta-Data must be supplied in this stage in order to select the right one out of hundreds of available variables in a Data Warehouse.

SPECIFICS FOR SAS DATA WAREHOUSE

A SAS based Data Warehouse which contains SAS data sets have a unique feature which enables users to create their own data sets by manipulating existing tables from the Data Warehouse by variety of SAS techniques like Subsetting, Merging, Sorting and Datasteps programming. This feature (Which usually not found in ORACLE or other products based

Data warehouses) helps users to perform very complex tasks often needed for a sophisticated analysis of data and advanced reporting. On the other hand, it implies that the Meta-Data, which was associated with the original SAS data set, must be inherited to the work library data sets, which the user creates in his SAS session.

Typically a SAS Data Warehouse consist two areas of data:

DATA SETS STAGING AREA

The staging area of the Data Warehouse consists the replication of tables from the operational system as SAS data sets.

DATA MODELS AREA

Data models are created from SAS data sets stored in the staging area. Models are usually created by merging and manipulating data sets to represent business logic to End-Users in a more easy way. The Model is a regular SAS data set but it contains variables from different source data sets.

In SIGMA WAREHOUSE we enhance this schema by adding hierarchies to variables as explained here by.

HIERARCHIES OF VARIABLES

A typical data model in the data warehouse can have 100-200 variables. For example, let's look at a very "modest" model description:

```

custname = 'Customer Name'
custno   = 'Customer Number'
custst   = 'Customer State'
custcity = 'Customer City'
custstr  = 'Customer Street'
custhno  = 'Customer Str Number'
custzip  = 'Customer Zip'
custtel  = 'Customer Tel'
custfax  = 'Customer Fax'
custmail = 'Customer Email'
custstat = 'Customer Status'
custprof = 'Customer Profession'
custsex  = 'Customer Gender'
custbirt = 'Customer Birth Date'

contrno  = 'Contract Number'
contrst  = 'Contract Status'
contdat  = 'Contract Begin Date'
conttot  = 'Contract Total Debt'
contcred = 'Contract Total Credit'

billno   = 'Bill Number'
billst   = 'Bill Status'
billprds = 'Bill Start Date'
billprde = 'Bill End Date'
servcode = 'Service Code'
servtrf  = 'Service Tariff'
servqnt  = 'Service Quantity'
servtot  = 'Service Total Amount'

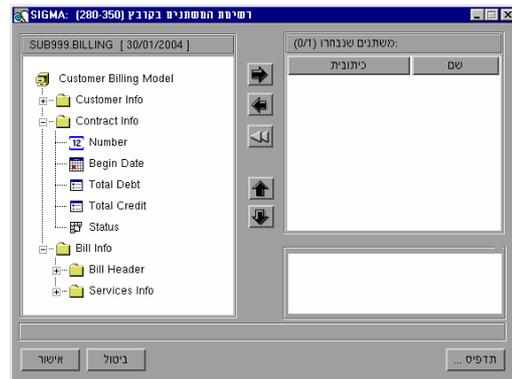
```

SAS applications will display these variables in a "flat" way:

Column Name	Type	Length	Format	Label
custname	Text	30		Customer Name
custno	Number	8		Customer Number
custst	Text	2		Customer State
custcity	Text	30		Customer City
custstr	Text	30		Customer Street
custhno	Text	10		Customer Str Number
custzip	Text	5		Customer Zip
custtel	Text	20		Customer Tel
custstat	Number	8		Customer Status
custprof	Number	8	PROF.	Customer Profession
custsex	Number	8		Customer Gender
custbirt	Number	8	MMDDYY10.	Customer Birth Date
contrno	Number	8		Contract Number
contrst	Number	8	CONSTAT.	Contract Status
contdat	Number	8	MMDDYY10.	Contract Begin Date
conttot	Number	8	COMMA14.2	Contract Total Debt
contcred	Number	8	COMMA14.2	Contract Total Credit
billno	Number	8		Bill Number
billst	Number	8	BILLST.	Bill Status
billprds	Number	8	MMDDYY10.	Bill Start Date
billprde	Number	8	MMDDYY10.	Bill End Date
servcode	Number	8	SRVCODE.	Service Code
servtrf	Number	8		Service Tariff
servqnt	Number	8		Service Quantity

SAS 8.2 – Variable window.

Imposing hierarchy on variables means grouping them in a tree like structure (like an explorer tree) where it is easier to find the variables and to understand their role in the data set:



Sigma Warehouse - Variable selection window

OBJECT ORIENTED META-DATA

Having all these needs and specific attributes of SAS data warehouse in mind, we have designed a unique Meta-Data subsystem built into SIGMA, which is based on an OOD (Object-Oriented-Design).

META-DATA OBJECTS

A Meta-Data object describes a variable, which can be found (Have an Instance), in many models in the Data Warehouse. All these variables inherit the Meta-Data from the same object.

The object contains the following attributes:

Report Label - The label to be printed in a report's headers – Usually a very short one.

Short description -To be displayed in the variables selection window. Contains information about the variable, which gives immediate understanding of the business entity behind the variable.

Long description – more detailed info to be revealed if the user wants to enhance his knowledge about the variable.

Document – A full text document (Text or HTML) if a document is needed to fully understand business rules and formulas associated with this variable.

Operational System Meta-Data – The name and screen number from the operational system where this item can be found. This is one of the most important items requested by End-Users since they are familiar with the screens of the operational system.

Custom fields – Field where every organization can store organization's specific details.

Key Words – Words that can be used in the search tool to find this object.

Format – The SAS (or user defined) format associated with the variable.

Type – The SAS systems distinguish between two data types: Numeric and Character. This is not sufficient for End-Users since it does not give them information about the usage of the variable. The Meta-Data of Sigma distinguish between the following enhanced types:

Class Variable - A variable with distinct values usually contains codes from the operational systems to be formatted using a user-defined format. This variable can be of numeric or character SAS types (E.g. Client's status: 1- Active 2 - Disabled 3 – Not Active 4 - Potential etc.)

Analysis Variable - A numeric variable, which is used for analysis, tasks. (E.g. Budget etc.)

Date Variable – A numeric variable with a data format (E.g. Birth date – ddmmyy10.)

Time Variable – A numeric variable with a time format (E.g. Peak hour – time5.)

DateTime Variable – A numeric variable with datetime format (E.g. email datetime 21MAY2004:13:45:23).

Numerator – A numeric or character variable which contains digits only (E.g. Customer number – 6678009, Zip code 46785 etc.) These “numbers” can't be used in calculations.

String – A character variable

As we will illustrate later, this information is heavily used in the report writing process.

INHERITANCE CONCEPT

Since a variable can have instances in many data models, It is important to save its Meta-Data object once in the repository (We use a SAS data set for this purpose) and to delegate the attributes to each instance of the variable.

We recall the special characteristic of SAS data warehouse where every user can create data sets derived from models for his specific analysis needs. This implies that the inheritance chain must be continued for these data sets as well.

In SIGMA WAREHOUSE we use the built-in LABEL attribute of variables, which is automatically inherited by every data set created by SAS. Version 8 of the SAS systems allows labels up to 256 characters long. SIGMA WAREHOUSE uses a specific structured layout of this attribute in order to save the hierarchies of each variable in a data model and the pointer to the Meta-Data father object.

Even if the variable will have instance in a data set created later by an End-User after running many data steps and procedures, It will still maintain its hierarchy information and the pointer to the Meta-Data object. New variables created by calculations in a data step or by SAS procedures are labeled the same way.

OVERRIDING CONCEPT

The object-oriented scheme enables the overriding of inherited attributes in a specific model. This feature gives flexibility to add or change business Meta-Data specific to instance in one or more models. For example, Customer type variable can have slightly different explanation in different models so SIGMA allows the overriding of any attribute to reflect these kinds of modifications.

USAGE OF META-DATA

SEARCHING FOR VARIABLES

SIGMA WAREHOUSE enables users to search the Meta-Data repository if they don't know what model holds the relevant variables.

The searching tool can find variables based upon free text format and key words. The results of the search are all instances of the relevant variables in all data models.

REPORT WRITING

Meta-data is integrated in every window within SIGMA WAREHOUSE. The content of the Meta-data is used in a clever way in order to make automatic selections for the End-User to eliminate the need to know SAS code and complexity. The enhanced type system helps SIGMA to place variables correctly in SAS code as in the following examples:

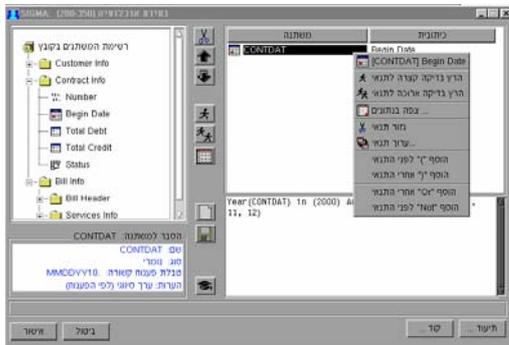
VARIABLE SELECTION:



SIGMA WAREHOUSE – Detailed report variable selection.

The above example shows an application where class variable are automatically assigned as “Order” variables in Proc Report while analysis variables are assigned as “Sum” variables and have “Break After” attribute.

WHERE CLAUSES



SIGMA WAREHOUSE – WHERE Clause composing

Where clauses composing is the most painful element of report writing. The above tool combines the Meta Data objects to first help the end-user to select the appropriate variable

and then to compose the where clause as automatically as possible. For example, writing the phrase “Bill Date is in January or March 2004” can be written in SAS as:

```
Where Year(Bill_d)=2004 and
      Month (Bill_d) in (1,3);
```

Which is unreachable for most users. SIGMA tool detects the enhanced “Date” type of this variable and allows the users to compose the above where clause through a series of structured pop-menus.

CONCLUSION

Meta-Data is an important part of every Data Warehouse project. Good Meta-data that is involved in the process of analyzing data can contribute to the quality of outputs.

TRADEMARKS

SAS is a registered trademark of SAS Institute Inc, USA

SIGMA WAREHOUSE is a trademark of EIS Ltd., ISRAEL

AUTHOR CONTACT

Gad Kotler
 President
 EIS Ltd.
 22 Ben-Gurion St., Herzlia, 46785
 ISRAEL
 Tel: 972-9-9575299
 EMAIL: gady_k @ eis.co.il