Paper 36-25

SAS® Metadata Architecture and Current Industry Metadata Trends

Vernée Stevens, SAS Institute Inc., Cary, NC

ABSTRACT

We all learned from data warehousing gurus and, sometimes from suffering the chaotic creations of our own, that we need to develop a metadata strategy and *use it* as we grow our data warehouse environments. The uses for metadata are growing and it is becoming an area of data warehousing that may be more dynamic than the data warehouse itself. The industry is seeking to develop supporting standards. Vendors are continuously adjusting to the new demands created by their own development efforts, as well as those of others that create or use new metadata.

SAS Institute is developing Metadata Architecture to support flexible storage and use of metadata by SAS applications, as well as other applications needing it. These demands are increasingly diverse, requiring a robust architecture to ensure metadata integrity across this distributed environment. We also work to stay in step with industry-wide efforts toward consistent metadata standards.

INTRODUCTION

From the time that business people first began to ask more than could be delivered from core operational systems, we have been faced with fundamental questions about the data itself:

Where is it?

What is it?

Where did it come from?

How old is it?

Does it mean what I think it means?

This information has been in both structured and unstructured documentation, or in the heads of in-house data gurus since the first "rogue" business applications data were hand entered into spreadsheets from our favorite weekly line printer reports. When this process grew up, and became the data warehousing industry, our conceptual leaders called this information "metadata".

We learned that all good data warehouse builders created and managed this metadata. In the wild, so to speak, data warehouse builders were very busy designing and building data warehouses, moving data around, and providing multidimensional reports to users, in hopes that they would say that one simple word, "Wow". We hoped to get to the task fully developing our metadata sometime next year, or the year after that. Discussion of metadata strategy seemed to create that glazed-over look in the eyes of our project team. We counted on our warehouse management tools to collect what they could without much embellishment from us. We hoped our exploitation tools would require minimal drudgery and pick up whatever they must in order to run. Basic metadata stored in a data warehousing environment might include (but not be limited to) the following information:

For Operational Data, other source data, and target repository:

- Location
- Table and column definitions
- Owner, administrator

Information about transformations:

- Mapping of source to target (and intermediate steps)
- Derived data definition
- Business rule descriptions
- Dependencies
- Computing platform
- Load frequencies

Data warehousing, in itself, is already formidable in architectural considerations. When we begin to suggest that its metadata, a necessary by-product with little "pizzazz", can have architectural requirements of its own, it is likely to excite our imagination… perhaps along the lines of playing golf instead, or considering an alternative career. If we dare to even imagine that architecture, we are faced with the possibility that it TOO may have a layer of data above it, and a layer above that as well.

It is only when we realize what this might allow us to do: things with real "Wow" appeal, that we can seriously engage the task. As the data warehousing industry has grown, so have the range of possibilities for exploitation of the data warehouse. Deeply specialized exploitation, such as data mining, and wide access to the Internet, the availability of thin-client presentation of business intelligence, wireless distribution, are all factors which are driving the industry to get serious about using metadata. Since these applications both use and create metadata, cooperation of the whole industry, as well as substantial investment by individual vendors is necessary. Otherwise we end up in a world with isolated repositories, inconsistency, many versions of reality, all those things that drove us to create an integrated data warehouse to begin with.

This paper will explore how metadata will serve as a foundation for an integrated and automated data warehousing environment.

METADATA CONTENT

Metadata has been categorized in a number of ways in the data warehousing literature. The first breakdown, regardless of the terminology, tends to occur in terms of the primary audience or users of the information. For instance, Technical vs. Business MD. Technical metadata is used by IT professionals in the planning, design, creation, and ongoing development of the data warehouse. Kimball has referred to this as "back room metadata". For example, some of the metadata for a SAS table might specify a certain number of rows and columns, with certain data transformations applied to some columns.

Business users require more descriptive information, which will assist in translating codified information into the business concepts relevant to their domain. This would include the content and purpose of the data, related business rules, ownership and administration, and location.

However, considering technical metadata to be focused exclusively on the extraction, transformation, and loading, or assuming that those functions are of interest only in the "back room", can be a mistake. Many users are interested in understanding the origin of the tables, fields, and values they see… and not necessarily in "watered down" terms. They need to know certain basic facts about the data that are much the same as technical metadata. The same fundamental information may appear in both, and may or may not be expressed

differently. In this sense, one shouldn't interpret these categorizations as mutually exclusive.

While applications may depend heavily upon metadata to function, they may not be able to use that which is available. As a result, application metadata has been created along with the application, however it is possible to create it, and not necessarily in a manner consistent with some standard or grand scheme. As data warehousing initiatives have matured, the range of applications utilizing the data warehouse has become more complex and diverse. They are increasingly interdependent. Metadata to support those applications have become more complex at the same time.

There is a great deal of common ground in the metadata needs at various points in the Data Warehousing lifecycle. The data may not be exactly the same, but it is probably similar. Yet this does result in compromises when a standard is sought. The need for standards has become recognized within organizations and across the industry. Creation and implementation of standards is, unfortunately, not easy in either case. Within organizations, this cooperation can become a spider web of task dependencies that can stunt progress. To establish standards across the industry requires cooperation between competing vendors.

SAS METADATA ARCHITECTURE

The increasing complexity of metadata requirements has driven most industry vendors to continually develop and revise existing metadata architecture to account for a growing number of metadata sources, server- and client- application needs, and development platforms. In a world without solid standards, the architecture needs to be robust and flexible. As a provider of end-to-end data warehousing solutions, SAS Institute is uniquely positioned to leverage a robust architecture.

SAS based solutions are increasingly including thin client JAVA or Microsoft Foundation Class (MFC) based applications. This has required some architecture changes over traditional Multi Vendor Architecture (MVA) approach. This variety of development platforms calls for a layered architecture that maximizes flexibility.

Specifically, for any given application, there are distinct functional layers:

Layer	Description
Facility	The Common Metadata Facility (CMF) controls manipulation of repositories and the actual creation, deletion, and persistence of metadata objects for access by applications.
Model	The Common Metadata Model (CMM) defines the object-oriented structure of the metadata, with classes, attributes, and methods.
API	The application program interface (API) or presentation layer that supports client applications.
Repository	The physical repository of metadata.

Table 1 - Layers of the SAS Metadata Architecture

COMMON METADATA FACILITY (CMF)

The CMF acts as a moderator, providing tactical control of the metadata. When metadata is needed by an application, the CMF can translate the request as necessary to read any part of the repository. The CMF controls the creation, deletion, updates, and persistence of metadata objects. This provides common metadata services to SAS data warehousing technologies and exploitation applications. Beginning in Version 7, this allowed metadata integration between SAS/Warehouse Administrator and SAS/EIS, SAS/MDDB, and the SAS external file interface. This will extend to other applications in the future. We will also see the CMF develop a new range of capability.

Data warehousing sites that are heavy users of metadata know that these repositories can become vast and complex. Updates to metadata have effects on interdependent repositories. Predicting and testing the effects of changes in the source data, and resulting changes to the data warehouse metadata is extremely important as data warehouse projects mature.

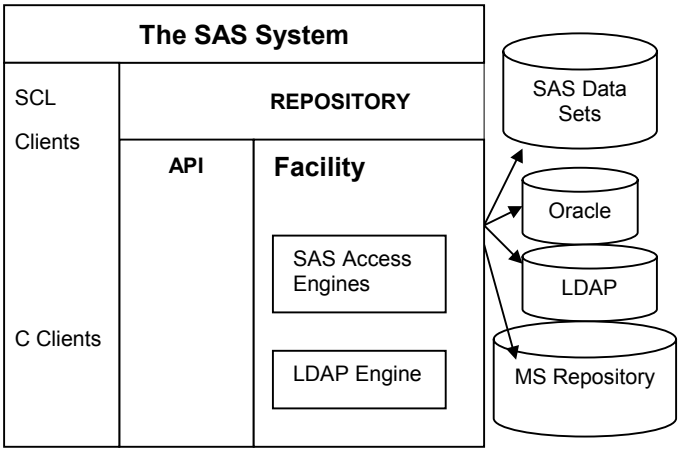

Figure 1- The Facility within the Architecture.

To assist in managing changes and supporting multiple developers, the CMF is growing to support multiple environments, such as production, development, and test environments for metadata. The production environment is the repository used to surface metadata to end users. Check-out and check-In functions operate against the production environment to replicate metadata objects to a development environment. Updates to metadata are applied there, then the metadata is moved to a test environment before being checked-in to production again. This cycle enables high standards for metadata integrity.

In addition to managing change, we often need to know how the metadata looked before various rounds of changes were implemented. The direction for development includes versioning, which allows the spoke version to be compared and to transfer only the more current updates. Check-out and check-In functionality would provide the locking facility allowing the target repository to temporarily own the object.

The CMF is controlling metadata change and movement or deployment. Change can come due to changes in data content or structure in the data warehouse, or it can come due to applications that require existing metadata or create new metadata that should be incorporated into the metadata structures.

COMMON METADATA MODEL (CMM)

The model ensures that client applications which access the repository all produce the same interpretation of the metadata. The model provides a means to determine how complete is the repository.

In order to share metadata successfully across applications, there needs to be a common definition of the metadata objects,

such as hosts, tables, and columns. The CMM contains the objects, relationships, and attributes or properties common across application products or metadata "spokes". The model will cover most of what is needed by the various applications. It can be extended by the applications as well. For example, OLAP metadata is created and used in several applications that are frequently used together: SAS/Warehouse Administrator, SAS/EIS software, and SAS/MDDB server. Version 2 of the SAS/Warehouse Administrator software provides an excellent interface for definition of OLAP metadata, such as column roles and hierarchies. These definitions can be used by the MDDB Server, and read by SAS EIS for reporting. SAS/MDDB Server can provide OLAP for OLE DB metadata to the MDDB procedure.

The applications are both using and contributing to a rich and dynamic model which defines the content of the repository. Neither the repository nor the model is limited to the CMM defined architecture. Applications can and do create their own metadata. The model provides the common elements across applications and is designed to be flexible enough to address a wide range of specialized metadata requirements for object types or persistence. For example, updates to metadata on the LDAP server may be required for global user and server information, Microsoft Repository might be good for application metadata.

COMMON METADATA API

In Version 8 of The SAS System, the presentation layer may be a client applications written in either C or Screen Control Language (SCL). The Common Metadata architecture is implemented in C for efficiency. Client applications written in C may access all methods and instantiate objects from C. Support is in place for C client applications to define new classes in C as well.

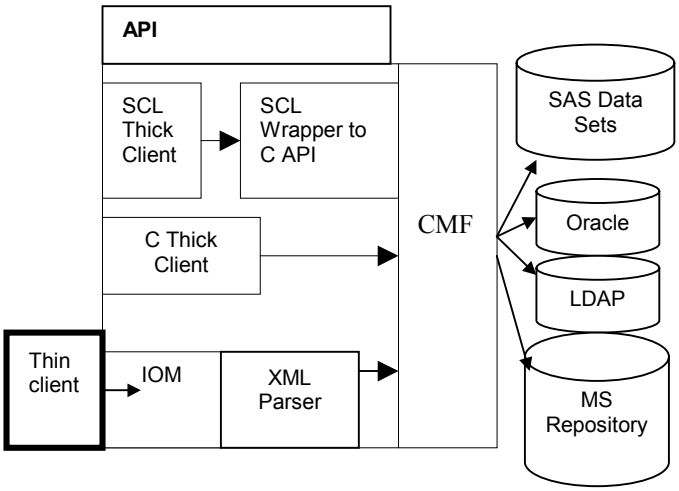

Figure 2 - Focus on the API within the SAS Metadata Architecture

SAS Software and customer applications both have and extensive code base in SCL. This architecture provides a wrapper interface to the methods that support the Common Metadata Facility and the persistence layer.

For a thin-client applications, the ability to separate the presentation from the application logic is especially critical. A supporting API is in development to process XML requests for retrieving and updating metadata. The XML stream moves to the SAS Server via IOM objects. The XML Parser translates the XML requests into method calls and then translates the results of the method back to the XML stream and returns the results to the thin client.

THE PHYSICAL REPOSITORY

The Common Metadata Facility can manage metadata stores in many different physical formats and on many platforms. Ultimately, the architecture is represented by a physical metadata store, but it may reside in a variety of forms or on several platforms. Just as data warehouse tables may be in any number of formats across an enterprise, metadata can be as well. As architectures become more complex, metadata versatility and flexible deployment will be as important as those characteristics are in data warehouse architecture.

"HUB AND SPOKE" DISTRIBUTION

In many organizations, having a single, central, shared repository of metadata not ideal due to large numbers or wide distribution of users, network performance, or application usage requirements. The SAS Metadata Architecture allows growth from central metadata, to simple hub and spoke designs, to a complex, distributed metadata environment.

Spokes can be created as department-specific hubs, user-specific hubs or application-specific hubs depending on usage requirements. Spokes can be other SAS repositories or non-SAS metadata stores.

The overall repository architecture is built from a central repository (the hub) and uses copy management to populate or distribute metadata to any number of client repositories (spokes.) Hub to spoke copy tools can convert metadata to a different persistent model (SAS or non-SAS) as in Figure 3 - Hub and Spoke Architecture. The Hub and Spoke diagram depicts repositories acting as both a Hub and a Spoke.

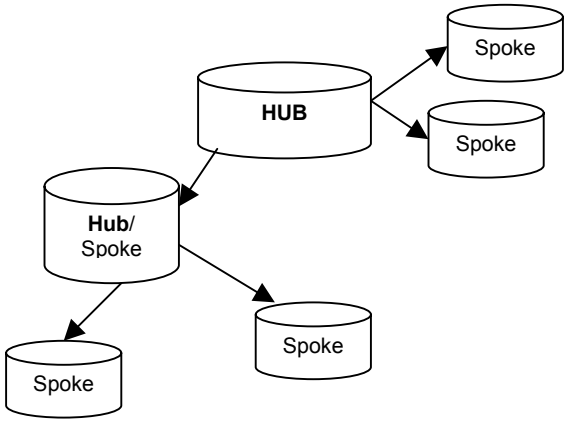

Figure 3 - Hub and Spoke Architecture

In Version 8 of The SAS System, Copy Management tools move metadata in a single direction, from hub to spoke, and is stored in SAS data sets. The intent of the SAS metadata architecture is to expand to support a variety of physical stores. This could include storage of metadata in RDBMS systems through Libname Engine support, LDAP, or Microsoft Repository support.

INDUSTRY STANDARDS INITIATIVES

The absence of metadata standards has been one of the biggest obstacles in pursuit of strong metadata management in any individual data warehouse. It is also a barrier to the development of tools to do the job. The Meta Data Coalition, a group of about 50 vendors including SAS Institute, was established in 1995 in an attempt to bring order to the situation. In 1997, they released the Meta Data Interchange Specification (MDIS) 1.1, a very basic approach using an ascii-based data interchange format. This generic work is easy to support, but not rich enough to serve the longer term needs, as it provides a

model for only the most basic data warehousing concepts.

The MDC focus has shifted to the Open Information Model, developed primarily by Microsoft, which is an effort toward a model of broader scope, beyond data warehousing. It includes models for business engineering, knowledge management, objects and components, and software analysis and design. These various initiatives point out that the need for standard models we experience in data warehousing are not unique. Common means of describing problems, components, data interchange formats, and syntax are critical to a robust development environment. This specification enables growth in content and scope for data warehousing efforts, which can be crucial to the future value of our data warehouses. Knowledge Management and Business Engineering information are a natural companion resource in decision support and a reasonable growth path for data warehousing initiatives. Table 2 provides a glimpse of the OIM initiatives.

OIM MODEL	DESCRIPTION
Business Engineering	Storage and interchange specifications for the many modeling and diagramming techniques used to describe business processes.
Knowledge Management	Organization of disparate types of information assets. Catalog structures, business terminology, semantic relationships, mapping to storage structures.
Database and Data Warehousing	Schema management and reuse, other logical database concepts. Multidimensional (OLAP) support. Data transformation metadata supports design, data quality, impact analysis, source-target relationships
Objects and Components Model	Component-based development for sharing, reuse, interoperability. Metadata tracks life-cycle of components from design to enhancement
Analysis/Design	Object-oriented modeling elements. Means for referring to elements outside the model.

Table 2 - Open Information Model (OIM) Metadata Standards Initiatives

Metadata standards are also being proposed by Object Management Group, spearheaded by IBM, Oracle, NCR, and Unisys, among others to produce a standard called Common Warehousing Metadata Interchange (CWMI). This initiative is also broader than just data warehousing concerns. There is, at least, a reasonable amount of lip service and organizational "cross- pollination" being given to ensure that these two standards are not wildly divergent.

Historically, some of the vendor organizations and individuals are arch-competitors and these parallel standards may have more to overcome than just the content itself if a common working environment is ever to be achieved. On an official level, some collaboration has been negotiated. OMG and MDC have each joined the other's membership. The process of achieving convergence of the OIM and CWMI will be an interesting study of industry politics. SAS Institute remains actively involved with the intention of supporting the industry best practices and emerging standards continuously.

CONCLUSION

The SAS Metadata Architecture provides support for the large base of existing applications, and paves the way for the wide variety of emerging development. Aggressive participation in the development of standards ensures that SAS Institute solutions will function in concert with the data warehousing industry. Delivering power for decisions in diverse technologic environments is SAS Institute's heritage. Whatever the confusion in the industry over metadata standards, SAS Institute in a unique position to overcome metadata integration issues.

We are the premier providers of end to end data warehousing technology. It could not be more important to anyone else.

REFERENCES

Kimball, Ralph; Reeves, Laura; Ross, Margy; Thornthwaite, Warren. *The Data Warehouse Lifecycle Toolkit: Expert Methods for Designing, Developing, and Deploying Data Warehouses* (1999) John Wiley & Sons, Inc.

ACKNOWLEDGMENTS

Thanks to the following members of SAS Institute in Cary, NC for help in preparing this paper:

Tony Fisher
Craig Rubendall
Lynette Dromsky

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Vernée Stevens
SAS Institute Inc.
100 SAS Campus Drive
Cary, NC 27513
919 677-8000
v.stevens@sas.com