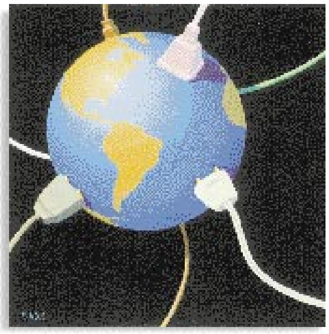
## Information Technology Demystified

A Report from the UniForum Technical Steering Committee

# Metadata Tracks a Moving Target



Each month, the TSC examines a key emerging technology or its use. This time, we look at a major aspect of data warehousing.

Today, not only is business changing rapidly, so are the demands on operational IT systems. Industry experts contend that large IS organizations spend up to 80 percent of every programming dollar just on maintaining operational systems. As IT continues to evolve and its uses to expand, there is little chance that this cost will decrease.

Much of this maintenance expense entails changes to operational applications and databases. Therefore, the “hot” application of the moment—the data warehouse—is built on shifting sands. If a data warehouse team does not build a maintenance strategy into its architectural design, the cost of maintaining the warehouse is likely to rival the cost of maintaining operational systems.

The key to avoiding this maintenance burden lies in the area of metadata management. *Metadata* is simply data about data: what fields constitute a record definition, the characteristics of each field, where and how the data defined by the record definition is stored and other characteristics. In many organizations the task of uncovering the metadata that defines the operational systems and their interdependencies can constitute a formidable task for the warehouse design team.

Metadata is typically stored in different locations, often in diverse, incompatible formats. In fact, some necessary metadata may not be available in any readily accessible way, but buried in application code that exists in data interface programs which link related operational databases.

Add to this the fact that production databases are rarely rebuilt through a change in the database schema; the systems in question simply cannot be taken down long enough for this to be done. As a result, many versions of the schema are only implicit within a particular database. (For example, if field A contains “1,” then the following fields represent X; otherwise Y.)

Employee turnover also has an impact. Individuals who implemented changes in the first place may no longer be available to help anticipate problems or provide insights on past change rationales to help solve the new problems. The overall result may be that no standard format exists across company databases.

## Common Types of Metadata

The complexity of this problem becomes even more evident when one considers the variety of types of metadata required to provide IS users with the information they need to make intelligent judgments about modifying existing systems. In the event that modifications are made, this metadata must be accessible in a form that allows the warehouse maintenance team to analyze proactively and react quickly to minimize the impacts of those changes.

A fundamental type of metadata is the *definition of the databases* being maintained under each database management system or file system. As noted, it is not uncommon for this information to be stored only within the application code. The database schema may have been redefined at some point to change the meaning of some fields, while preserving the previous field boundaries and data types to avoid rebuilding the entire database. This can be one of the more difficult types of metadata to manage, because unless a staff member recalls the schema redefinition, the correct metadata can be discovered only through trial and error.

Equally important is information about the *relationships between the data elements* stored under different data access systems. Today, a significant part of acquiring metadata may be automated by software tools. But this does not help when the relationships between databases are not recorded electronically. For example, in an employee identification field “EMPLOYEE-ID” in one database may be equivalent to “SOCSEC#” in another.

*Data values* often are semantically inconsistent, or even if semantically equivalent they can be represented differently so that some type of transformation is required before the data can be correlated. Sometimes these semantic inconsistencies may require considerable conditional logic to transform the source values into the appropriate form for the target system. Moreover, because the designers of operational systems were motivated to save disk space and CPU cycles, much of the data that resides in operational databases is not appropriate for use by end users (for example, symbolic fields like city names are often represented as integers, or some data is in binary form). As a result, building a warehouse entails the creation of numerous, sometimes complex business rules.

Another key type of metadata is *data primacy*. It addresses which database should be considered the *database of record* for a replicated data value. This is an example of metadata that is rarely recorded electronically; users of the system often consider data primacy to be a matter of common sense. For example, a

By Katherine Hammer

customer's address may be stored in multiple databases, but in case of a conflict in the record content, the user is likely to consider the address in the billings database to be the most accurate. In other cases, it may be difficult to determine the database of record.

## Iterative Management

As indicated above, one of the most important reasons for keeping metadata in a centralized location is *impact analysis* when change occurs in the operational environment. As a result, it is important that an organization's metadata management strategy include a mechanism for versioning of metadata, so what one learns about the various versions of schema that underlay most operational databases can be captured for use by later projects, rather than having to be rediscovered each time.

Because different types of metadata are distributed throughout heterogeneous systems, an organization may not be able to document its metadata fully before undertaking a strategic IS project. On the other hand, to escape spiraling maintenance costs an organization must try to gather and maintain as much of this information as possible for all existing projects. It must also fully integrate this activity into its development methodology. By practicing this philosophy, the metadata acquired by one project can be reused by others.

In short, IS organizations should strive for the appropriate mix of methodologies and tools to support iterative and incremental development. Using this approach, each element of the project is subject to managed change. It is important to consider what metadata would be helpful in reducing the time required for analysis and/or implementation for each of the types of changes that are likely to be made. Once the probable change scenarios are identified, an evaluation of software tools can be based on how well they capture and support the use of this information. Likewise throughout the project, the methodology should be regularly reassessed, tuned and distributed across the organization to maximize its benefit.

## Tools and Standards

As one might expect, the software industry has risen to the metadata management challenge with a variety of solutions. Many data dictionary products and repositories seek to provide a centralized facility for managing metadata. Additionally, CASE and design tools not only capture metadata but support the effective design of new (usually relational) database schemas, as well as generating some straightforward database applications. Data extraction, integration and warehouse products also use the same type of metadata to automatically generate data interface programs.

To enable enterprise data management, these different tools must be able to easily exchange the metadata created by other tools and stored in a variety of storage facilities. The rapid proliferation of these tools has resulted in almost as many different treatments of metadata as there are tools. The only way to enable the exchange of metadata between tools is to establish at least a minimum common denominator of interchange standards and guidelines.

In response to this problem, a number of initiatives have been launched to develop a simple interchange format. Some of these initiatives are vendor-specific, like those of IBM and Oracle, but other vendor-independent efforts have been organized as well.

While the prospect of locating, compiling and managing enterprise metadata may seem overwhelming, the combination of a sound, iterative methodology, today's tools and tomorrow's standards is expected to make the management of metadata less difficult. Metadata interchange standards should enable IS managers to select what they perceive as a best-of-breed configuration of tools to build a support infrastructure that fits their unique needs.

***Katherine Hammer*** *is cofounder, president and CEO of Evolutionary Technologies in Austin, TX. She can be reached at kay@evtech.com.*