The Information Repository: A Tool for Metadata Management

Noushin Ashrafi
University of Massachusetts at Boston

Jean-Pierre Kuilboer
University of Massachusetts at Boston

After a departure from the narrow view of data administration, a period of loose user involvement, and experimentation with stand-alone Computer Aided Software Environment many firms have come to realize that the promises of modern information systems have not been fulfilled. The failure to integrate competing models has limited overall productivity progress. There is a current renewal of interest in an integrated, but not centralized, modeling and data administration function supported by one or more information repositories. The following article is aimed at defining the modern repository, its value to the firm, and offering guidelines for the selection of the most appropriate tool.

The shortcomings of manual development of software have resulted in pressure to automate the process of software development. CASE (Computer Aided Software Engineering) tools help software developers to record and manipulate application requirements, data definitions, data flow, and procedural logic. These tools can have both textual and graphical interfaces. However, not all CASE products currently available in the market have the same capabilities in all areas. Most of the products tend to excel in some area and fall short in others. Therefore, the software developer, who is committed to using CASE tools, is in the position of either choosing a set of integrated tools, which may not meet their specific needs, or having to customize tool integration.

Developers who choose the latter, face the problem of bridging between CASE tools. Developing and maintaining bridge software imposes an extra burden to the already very difficult and time consuming process of software development. Few customers have the resources and knowledge to implement this approach. Most are limited to an incomplete multi-vendor solution or to a subpar single vendor.

In today’s multi-vendor, multi-platform CASE environment, managing corporate resources is more challenging than ever before. Since the reason behind using CASE tools is primarily to increase efficiency and improve productivity, it seems counter productive to spend the time and efforts for bridge building and maintenance of integrated tools. The solution to this problem is a single robust repository, which provides a single integrated model used by all CASE products.

A repository provides a supporting environment that facilitates the integration of tools, the standardization of software systems descriptions, and the sharing of information across applications, platforms, and system life-cycle phases. A repository is an organized group of information that supports business and data processing activities by providing a single logical point of control for management and the sharing of information. A consolidated view of the information model is more and more essential to the business unit as well as to the organization because, while past decisions were based on a stable information architecture and a centralized location of data, current decisions have to be made in a very dynamic and distributed environment.

The solution is attracting the attention of traditional MIS because of its control properties. Leading edge software developers are enticed by repository support for leading CASE methodologies and the possibility to adapt to future techniques. It is commonly acknowledged that the productivity revolution is not likely to happen based on a single product or technique. The concept of a repository as an integration vehicle is comparatively more attractive in a heterogeneous market. Users, even the ones burned by the unkept promises of
IBM AD/Cycle may reconsider the repository as a viable solution to inter–connectivity of disparate tools.

**Background and Definition**

Until recently, data administration has been instrumental for the management of the most important resource of any organization, information. Data administration has provided a supporting environment for all functional areas at all levels of the organization. Data administration’s functions center around (1) helping the organization to standardize its business structures and rules, (2) making sure that information is managed and used in a proper context. This system is primarily concerned with planning, controlling and administering metadata, data about the organization of data.

However, until recently the metadata concept was not fully understood nor were its resources completely accessed outside the realm of the data administration function. For example, the system development team is a group that should benefit most from this accumulated knowledge about the organization but has not done so due to its inaccessibility. An information repository facilitates the organization wide access to metadata. The main advantage of a repository is that it is accessible at the individual, business unit, and organization levels. The information itself can be stored on a heterogeneous confederation of media such as databases, and files. Metadata, however are stored and controlled in the repository.

The repository concept has evolved from the data dictionary and encyclopedia. Repository, datadictionary, and encyclopedia have much in common. The data dictionary in its most recent incarnation, IRDS, Information Resource Dictionary System, is defined as a four layer storage for Schema Description, Schema, Data, and Production Data (Hazzah, 1989). The data dictionary, the parent of metadata management, as it exists in the products marketed today, has far more abilities than simply documenting data structures. For this reason, it has been suggested that the term Data Dictionary must be dropped altogether because it does not reflect the characteristics of the conventional dictionary. The name data dictionary could even be detrimental to the products commercial success (Moriarty, 1993). Recognizing that data dictionary does not reflect an encompassing solution to the modeling of the organization, some authors and vendors extended and labelled their products encyclopedia.

The encyclopedia developed by Texas Instrument has data dictionary capabilities and more. Where data dictionary was supporting only the physical data structure, the encyclopedias supports the development process through integration of CASE tools. The objectives of the encyclopedia are more focused than a data dictionary (i.e. it serves as database for new applications development or as a support tool for specific CASE tool). A repository is a virgin encyclopedia without the company information (McGaughey & Gibson, 1993). A repository is a supporting tool, which is purchased without the enterprise specific metadata but offers complete check–in / check–out interfaces to many popular CASE tools and Database formats.

**Benefits**

Users of Computer Aided Software Engineering (CASE) and their associated environments have long recognized the importance of a storage facility for their data, relationships, models, and constraints. From preliminary systems study to the maintenance of existing systems, all phases can be assisted by the use of an efficient information repository. Decision support aspects are dealt with as well as the technical facets of the data driven function.

The strength of the repository based approach to development is its flexibility. Its benefits can be technical as well as managerial. Technical views of repository encompass data–oriented, process–oriented, and object–oriented views. The managerial benefit of a repository lies in its capabilities as a decision supporting facility.

**Data–oriented view**

The traditional data–oriented view of system development is effectively supported by many repositories. The database is the heart of the repository, therefore the data view is well supported. The user is able to store the entities, their attributes, and the relationships between entities. The underlying meta–model of many repository products is entity relationship based. The meta-model is the basis for tool integration and it allows sharing of both common data and methods that operate on the data. Virtually any real world situation can be presented using entity–relationship models.

**Process–oriented view**

The process driven development methodology is also supported by the repository. The repository can store Structured Analysis and Design models, their dictionaries, relationships, and documentation. In conjunction with the powerful tools of the future development workbenches, the repository has no difficulty in dealing with the procedural, functional, process based aspects of system development. A particularly useful characteristic can be the support for information versioning which can be applied in successive phases of system developments. From requirements to maintenance the version control capability is essential in keeping track of system changes.

**Object–oriented view**

The heart of the repository, namely the database where the repository engine stores the models and metadata can be implemented based on an object oriented metamodel. The traditional navigational data models such as the hierarchical or network model are ill-suited to store complex dynamic metadata. The relational model is an alternative but it lacks,
Related Content

Enhancing Database Access Control by Facilitating Non-Key Related Cover Stories
[www.igi-global.com/article/enhancing-database-access-control-facilitating/3334?camid=4v1a](www.igi-global.com/article/enhancing-database-access-control-facilitating/3334?camid=4v1a)

A Paradigm For Natural Language Explanation Of Database Queries: A Semantic Data Model Approach
[www.igi-global.com/article/paradigm-natural-language-explanation-database/51129?camid=4v1a](www.igi-global.com/article/paradigm-natural-language-explanation-database/51129?camid=4v1a)

Active Video Annotation: To Minimize Human Effort
[www.igi-global.com/chapter/active-video-annotation/28839?camid=4v1a](www.igi-global.com/chapter/active-video-annotation/28839?camid=4v1a)

Different Kinds of Hierarchies in Multidimensional Models
[www.igi-global.com/chapter/different-kinds-hierarchies-multidimensional-models/20687?camid=4v1a](www.igi-global.com/chapter/different-kinds-hierarchies-multidimensional-models/20687?camid=4v1a)