A Functionality-Oriented Criteria Set for Evaluating Information Resource Dictionary Systems

Bijoy Bordoloi, Sumit Sircar, Craig Slinkman
University of Texas at Arlington

Nitant Cakranarayan
Database Consultants, Inc.

This paper develops a detailed set of criteria with which the functional capabilities of different Information Resource Dictionary System (IRDS) products can be compared. The proposed criteria are primarily based on the ANSI and ISO standards for IRDS and related published literature in the field. The criteria set can be used as a guideline by vendors to streamline their products, or as a framework by potential adopters to compare and select an IRDS from multiple offerings.

In recent years, it has come to be recognized and accepted that data is as valuable a corporate resource as any other corporate resource such as money, material, or personnel, and that, like these resources, data must also be administered and controlled to ensure proper handling, proper access and proper utilization.

Data administration is the function that assists the organization in the management and control of data. It includes a human facility, the data administrator, and an automated facility called the Information Resource Dictionary System (IRDS), some common implementations of which are known as the Data Dictionary/Directory System, the Repository, or the Encyclopedia (Cashin, 1988). The data administrator acts as a manager of the corporate data resource. The IRDS incorporates a centralized “repository” of information about data relevant to the organization. The repository is an administrative database that allows storage and management of all the database and related information system definitions, referred to as metadata. It contains the attributes, domains, definitions, usage and relationships of the data in an organization. The major benefits of a central repository include:

- Establishing a single source of metadata to be shared and reused by various users and tools throughout the lifecycle of an application or information system
- Reducing or eliminating duplication of effort in creating and managing shared data
- Promoting and enforcing consistent definitions across interrelated application areas

The concept of storing data about an organization’s data and its information systems originated over three decades ago, with the use of COBOL copy statements to provide consistent data declarations that could be referenced by applications (Gillenson, 1993). This was followed by a succession of products, such as data dictionaries, and catalogs for relational DBMS. As a result of a proliferation of work in this area,
both the American National Standards Institute (ANSI) and what is now the National Institute of Standards and Technology (NIST) began to work on data dictionary standards in the early 1980s. They named the software “Information Resource Dictionary Systems,” and formulated a set of standards (X3.138) in 1988 (ANSI, 1989). These standards were recently extended in 1992 (ANSI, 1992).

Because of its growing importance, IRDS implementation has been a major software endeavor for many software companies, large and small. For instance, IBM has announced a product called Repository for DB2, its relational DBMS. Furthermore, the concept of a repository has been most successfully implemented in ICASE (Integrated Computer Assisted Software/System Engineering) technology. In fact, the repository or encyclopedia is the cornerstone of all the well-known ICASE packages on the market. Some examples are: ORACLE corporation’s CASE®Dictionary; Texas Instruments’ IEF (Integrated Engineering Facility), and Knowledgeware’s ADW (Application Development Workbench).

Several third party vendors have also introduced their repository products, and this market seems to be constantly expanding. The problem, however, is that there is a dearth, in the published literature, of a well-defined set of criteria to evaluate the functional capabilities of these products. This, needless to say, makes it extremely difficult for IS management to evaluate and select an IRDS product. This is the problem that we address in this paper as we develop a set of criteria with which the functional capabilities of IRDS products can be compared.

Several authors (Bruce, Fuller, & Moriarty, 1989; Narayan, 1988; Plotkin, 1992) have suggested criteria to select an IRDS, but they lean towards the business decision-making aspect of choosing an IRDS for an organization. Their criteria, for example, include marketing-oriented features such as vendor support and vendor reputation. The criteria of these authors do not focus in detail on the functional power of the IRDS and the overall functionality of the IRDS. Building on the works of these authors, we present a more detailed set of functionality-oriented criteria with which the capabilities of IRDSs can be compared (ignoring business related criteria such as cost, vendor reputation or vendor support).

Our proposed criteria are based on published literature in the area of IRDS, the ANSI and ISO standards proposed for IRDSs (primarily ANSI), and inputs from several practitioners in the field. We assume that the reader is familiar with the basic concepts and terminology of database management, particularly the concepts of database schema, metadata, the E/R model and the relational data model.

The rest of the paper is organized as follows. First we provide a summary discussion of some basic IRDS concepts and IRDS database architecture. The discussion presented in this section forms the basis for our proposed criteria which we discuss in the next section. We then provide some methodological guidelines for using the proposed criteria set, followed by concluding remarks.

IRDS Concepts and Database Architecture

Before we discuss the architecture and desirable functional features of IRDS, it is perhaps useful to review some basic IRDS concepts and the evolution of IRDSs from simple manual systems to complex automated ones.

Evolution of IRDS

In the early days of computing, data were stored in the form of small isolated islands, and were controlled by independent programs, so programmers and system developers needed a reference list for the names and definitions of the variables used in their programs. The need for sharing data amongst different applications was apparent even then. It was necessary to keep track of the data, their meaning and the relationship between the data and programs. Gillenson and Frost ([Gillenson & Frost, 1993] have traced the evolution of the concept of storing and using data about data, i.e. metadata. They found the following overlapping stages in the development of this concept: passive data dictionary, active data dictionary, relational catalog, hybrid relational data dictionary, American National Standards Institute (ANSI) Information Resource Dictionary System (IRDS), repository, and OODBMS catalog.

Passive data dictionaries were the earliest form of data dictionary. They are called passive, because they are used independently of the running of the DBMS, typically as system documentation tools. Well-known examples of such products for specific DBMS are University Computing’s UCC TEN, IBM’s DB/DC and Cincom’s Data Dictionary. Active data dictionaries represented an enhancement of the data dictionary concept. They are called “active” because the metadata they