Managing Organizational Data Resources: Quality Dimensions

VICTORIA Y. YOON, University of Maryland Baltimore County, USA
PETER AIKEN, Virginia Commonwealth University & Defense Information Systems Agency, USA
TOR GUIMARAES, Tennessee Technological University, USA

The importance of data resources as a major component of Information Resources Management has long been recognized (Guimaraes, 1988). Many organizations are discovering that imperfect data in information systems negatively affects their business operations and can be extremely costly. Results from a survey indicate 50 percent of IS managers reported losing valuable data in the last two years and at least 20 percent with losses costing $1 million or more (Panettieri, 1995). Another survey reports 70 percent of the IS managers having their business processes interrupted at least once due to imperfect data (Wilson, 1992). Still another study showed that the nature of the problems associated with defective data ranges widely, from damaged files and lost data accounting for 23 percent of the responses, cost overruns (17 percent), conflicting reports (16 percent), improper regulatory reporting (13 percent), improper billing (9 percent), poor decisions (7 percent), delivery delays or errors (6 percent), and others (9 percent) (Knight, 1992).

We believe imperfect data can result from practice-oriented and structure-oriented causes. Practice-oriented causes result in systems capturing or manipulating imperfect data (i.e., not designing proper edit checking into data capturing methods, or allowing imprecise/incorrect data to be collected when requirements call for more precise or more accurate data). Operational in nature, practice-oriented causes are diagnosed bottom up and typically can be addressed by the imposition of more rigorous data-handling methods. Structure-oriented causes of imperfect data occur when there exists a mismatch between user requirements and the physical data implementation designed to meet the requirements. The imperfections are inadvertently designed into the implementation. Correcting structural causes more often requires fundamental changes to the data structures and is typically implemented top down. Structural problems result when a user cannot obtain desired results due to lack of access and/or lack of understanding of data structure, as opposed to getting an incorrect value or representation.

Adopting an organization-wide perspective to data quality engineering integrates development activities using data architecture. Failure to develop systems as coordinated architecture components results in fragmented data resources whose definitions apply at best within system boundaries. One additional consequence is that data interchange among company systems and those of partner organizations is more difficult. Structurally defective data results in unfavorable outcomes such as: 1) providing the correct response but the wrong data to a user query because the user did not comprehend the system data structure; 2) organizational maintenance
of inconsistent data used by redundant systems; or 3) data not supplied at all due to deletion anomalies (i.e., storing multiple facts in the same physical entity).

Previous studies of data quality have addressed practice-oriented causes of imperfect data with data quality engineering methods such as those reported by English (1996) and Broussard (1994). Less guidance has been available to organizations interested in addressing the problems creating structurally defective data and how it relates to the comprehensive dimensions of data quality engineering. With the strong trend toward more integrated systems within and among organizations on a global scale, clearly defined data resources and management guidelines are increasingly required for situations where data crosses system boundaries. Many researchers have contributed to the evolution of a data life cycle model. We seek to build on previous work illustrating how a better understanding of the data life cycle results in better matches of data quality engineering techniques with life cycle phases.

Similarly, previous studies on data quality have identified the dimensions necessary to ensure data quality within system boundaries. Collectively the research work has resulted in a data quality model with three dimensions (data model, data value, and data representation), as reported by several authors such as Reingruber and Gregory (1994), and Fox, Levitin, and Redman (1994). As mentioned earlier, attempts to define data quality engineering methods have focused on correction of operational problems, addressing these three quality dimensions and directing attention to practice-oriented data imperfections.

The objective of this paper is to present an expanded data quality model that addresses practice-oriented as well as structure-oriented causes of imperfect data. The expanded data life cycle model proposed here enables us to identify links between cycle phases and data quality engineering dimensions. Expanding the data life cycle model and the dimensions of data quality will enable organizations to more effectively implement the inter- as well as intra-system use of their data resources, as well as better coordinate the development and application of their data quality engineering methods.

The next section of the paper defines the theoretical foundation for the paper. That is followed by a proposal to extend the existing conceptual model for data management with a data life cycle model consisting of eight phases: metadata creation, metadata structuring, metadata refinement, data creation, data utilization, data manipulation, data assessment, and data refinement. In turn, that is followed by a section outlining an expanded view of data quality engineering as encompassing four dimensions: data representation, data value, data model and data architecture, each with their specific set of attributes necessary to ensure data quality. The last section contains a short summary and some final conclusions for managers in this increasingly important area.

THE THEORETICAL FRAMEWORK

Semantically, data are a combination of facts and meanings (Appleton, 1984). When implemented, the logical label ‘meaning’ can be replaced with the physical implementation term ‘data entity structure’ (DES). The physical implementation of a DES is an entity/attribute combination. A data value is a combination of a fact and a DES specifying an entity/attribute combination—Tsichritzis and Lochofsky (1982) labeled this structure a triple. Based on present practice within most organizations, triples can have organization-wide scope; but, system managers consider themselves fortunate to have them consistently applied within a system and spend considerable time trying to manage multiple triple variations within a single system.

Based on a widely accepted definition, when data are supplied in response to a user request, they become information. For example, a DES associates a fact (23 beds) with a specific meaning (average occupancy of Ward C for Quarter 2). As a triple, this is provided in response to a hospital manager request inquiring as to the average number of beds occupied during the second quarter. The same triple is re-used to respond to other information requests: How effective was the advertising? What was the perceived product quality? Can we measure market penetration? If technology didn’t permit association of individual facts with multiple meanings, the data maintenance required to supply requested information would require more resources. Re-using DESs permits organizations to provide a relatively wide range/large amount of information by managing a smaller amount of data.

Also widely accepted is the importance of metadata describing specific data characteristics. Facts describing organizational data quality are one type of metadata. One instance of data quality metadata is the association among data model entities sharing common keys (model metadata). Data model metadata describes structured DES components used to represent user requirements. Data models represent these associations of respective triples with correct representation of user requirements and physical implementations. Another type of metadata important to data quality is the association among organizational data models (architectural metadata) which represent a major component for organizational data architecture. It includes information on the relevant entities and attributes, such as their names, definitions, a purpose statement describing why the organization is maintaining information about this business concept, their sources, logical structures, value encoding, stewardship requirements, business rules, models associations, file designs, data uses, specifications, repositories, etc. This architecture is a critical framework facilitating communication, thoughts, and actions among developers and data resources users. It works as the blueprint or master plan guiding and promoting data sharing by providing common organizational and industry-wide data