

PANEL: User Interfaces and Database Management Systems

T Lougenia Anderson

Tektronix, Inc
Computer Research Laboratory

Gad Ariav

New York University
Graduate School of Business Administration

MOTIVATION

Database management has in the past dealt primarily with the storage and retrieval of data. However, with the advent of high speed graphics display devices, natural language front ends, voice recognition technology, video disc technology, and so on, DBMS users are beginning to expect more from their DBMS interfaces. Of course, these demands on the user interface are also application driven. New applications such as cartography, computer aided design, and office automation have contributed to the rise in user expectations. An obvious question is, of course, how will database management systems evolve to handle these new user interface requirements.

Equally important, however, is the problem of dealing with existing (less exotic) DBMS user interfaces. For example, it has been estimated that more than half of the code in current commercial applications deals with user interface definition. This is the code that handles such mundane tasks as data layout, formatting, forms definition, validation, and representation conversions. This is also the part of applications code that typically requires the most maintenance. Even in DBMS-based applications, this code is replicated many times over, for data sharing stops at the level of the user interface descrip-

tion. Much more might be shared if user interface specifications were migrated to the DBMS, where they could be re-used and easily modified. Inclusion of the user interface specification in the DBMS could reduce redundancy in another way, since constraint information could be used by the interface to provide a more complete and responsive interface to the user and by the system itself to maintain data consistency. A second question, then, is whether database management systems should evolve to handle these "old" user interface requirements, and, if so, what tools will be needed.

The emphasis of the panel will be primarily on emerging principles underlying DBMS user interface design and implementation. Tangentially, examples drawn from the latest whiz-bang DBMS user interface may be used by panelists to illustrate these principles.

TOPICS

Applying General Principles of User Interface Design to DBMS

A user interface is any mechanism used to mediate between users and applications. The mechanism can range in complexity from I/O formatting details to natural language front ends. What are some of the principles of "good" user interface design that most, if not all, of these mechanisms must satisfy and how can they be effectively applied to DBMS applications? What evaluation techniques can be used to measure the effectiveness of a particular DBMS user interface?

For example, customization is believed to be one of the most important features of a successful user interface. That is, users themselves should be able to specify the particular ways in which they will interface with their applications. Can customization capability be effectively added to DBMS applications? How should this best be done?

DBMS Support for New Display Technologies and New Applications

DBMS support for new display technologies, such as graphics or video discs, is still in its infancy. Record and table-based schemas map nicely to the typical report format required by conventional data processing user interfaces. However, these record structures may not map so directly to the complex display objects possible with emerging technology. Also, at both the query language level and at the access method level, there is considerable difference between asking for an employee record by content ("Give me all employees who make over \$30,000 per year") and asking for video disc frames by content ("Give me all frames that contain some view of building XYZ")

What schema extensions will be required to support a more complex user interface embodying graphics objects, video images, or voice? Will new access methods be required? How will the mode of user interaction be affected by this new display technology (e.g., Does browsing in a conventional database differ from browsing in an image database? Do graphics really convey more information than tables?)

Data Modeling and the User Interface

Relational technology provides a facile abstraction of secondary storage, there is no analogous abstraction for user interfaces. With the advent of object-oriented database systems that support complex objects and multiple connectivity, a fixed format for displaying the results of queries seems no longer adequate.

Is there a formalism that supports declarative specification of displays on database objects, which also permits such desirable facilities as modular design, automatic generation of displays from specifications, and update through displays? What data modeling tools are necessary to represent forms, formatting information, the user's interaction behavior itself, and so on?

The DBSSG User Facilities Task Group

The ANSI-SPARC Data Base Systems Study Group recently established the User Facilities Task Group. The charter of this new group is to define a reference model for describing DBMS user interfaces. The reference model will define major components of the user interface as well as common terms and functions. The User Facilities Task Group's overall goals and results to date will be discussed by the panel.

PANEL PARTICIPANTS

Dr. Nancy McDonald
President, Computer Technology Planning
Co-Chair, DBSSG User Facilities Task Group

Prof. David Meier
Oregon Graduate Center

Prof. Ben Shneiderman
University of Maryland