
Research Areas

Database query languages, database design, and object-oriented programming.

Research Description

My main current research interest is in creating a unified database front-end system that provides easy-to-use yet powerful common language to access varying types of relational data management system (RDBMS), and shields the complexity of underlying RDBMS.

My system called GLAD II (Graphics Language for Accessing Database) allows users to interact with different relational DBMS by providing a common graphic language called DFQL (Data Flow Query Language) that is based on a data flow diagram. The system automatically translates a user specified DFQL query into the equivalent query statements recognized by the connected backend DBMS. At present, the prototype connects to Oracle and translates the DFQL queries into the Oracle's SQL statements.

My work is different from other similar-looking work and commercial products in that theirs only support a SQL connectivity, in that theirs do not shield users from the complexity of SQL. In contrast, DFQL provides a more logical, higher-level, and consistent query language. Users of DFQL do not have to bother with the poorly designed language features in SQL. In other words, theirs do not eliminate the semantics problem associated with data retrieval.

DFQL is a graphic query language based on relational algebra. It has been designed with sufficient expressive power and functionality to allow the user to easily express database queries. DFQL is relationally complete and includes an implementation of aggregate functions. An object-oriented implementation allows programmers to easily create their own DFQL operators from the primitive and other existing user-defined operations. This extensibility of query language is unique: No other query language allows such extensibility. The overall intention is to provide the user with a simple-to-use, yet powerful and extensible tool to implement database queries. A human-factors analysis comparing DFQL and SQL showed DFQL was statistically better than SQL.

Relevance To DoD/DoN

In the U.S. Navy environment, diverse types of databases exist. These databases are implemented with many different types of database management systems. Some of them are in the older hierarchical and network DBMS. Newer ones are being implemented in the relational DBMS, commonly Sybase or Oracle. Fully exploiting the information kept in these diverse databases is the key to the effective and efficient management of a downsized Navy. Although new databases are implemented with the relational DBMS, we are far, far away from the effective use of information. One major technical reason is that there is no single tool for the users to access those databases. To access databases, one must learn idiosyncrasies of individual database management systems. This situation of

underutilized information motivated my research of creating a unified front-end system GLAD II.

Recent Publications

C. T. Wu and T. A. Norman, *Introduction to Programming: Object-Oriented Approach with C++*. Manning Publishing division of Richard D. Irwin, Inc., 1996.

C. T. Wu and G. J. Clark, "DFQL: Dataflow Query Language for Relational Databases". *Information and Management*, 27 (1994), 1 - 15.

C. T. Wu, "Protocol vs. multiple inheritance". *J. of Object-Oriented Programming*, 5, No 4, July/August 1992, 72-75.

C. T. Wu, "Development of a Visual Database Interface: An Object-Oriented Approach". Chapter 4 in *Applications of Object-Oriented Programming*, L. Pinson and R. Wiener (eds.), Addison-Wesley, 1991, 101-138.

See: <http://www.cs.nps.navy.mil/~ctwu/>.