SIGMOD INSTITUTIONAL SPONSORS†

Electronics and Telecommunications Research Institute
Post Office Box 8, Daedog
Danji Daejoen
Korea 305-606

Hewlett-Packard Laboratories
1501 Page Mill Road
Palo Alto, CA 94304

Lotus Development Corporation
One Rogers Street
Cambridge, MA 02141

Microelectronics & Computer Technology Corporation (MCC)
3500 West Balcones Center Drive
Austin, TX 78759

Mitsubishi Electric Company
Computer Works
325 Kamimachiya
Kamakura, Kanagawa
247 Japan

† Benefits of sponsorship include two subscriptions to the SIGMOD Newsletter "SIGMOD RECORD" and the privilege of sending employees of your organization to conferences and workshops sponsored by SIGMOD at reduced member rates. In addition, your organization will be listed in the SIGMOD Newsletter as an Institutional Sponsor.
Editor's Notes

The main feature of this issue is a special section on Advanced User Interfaces for Database Systems edited by Yannis Ioannidis.

Following the special section are three submitted articles. The first is a paper by M. David on the advanced capabilities of the outer join as proposed by the SQL2 standard. The second paper by P. Mishra and M. Eich proposes a definition of functional completeness in the context of Object-Oriented Databases, and the third paper by D. Jonsson shows how to represent database states as well-formed formulas in formal logic.

A report on the first annual meeting of the EXPRESS Users Group is presented by P. R. Wilson, and V. Soloviev provides a comparative overview of three commercial Object-Oriented DBMSs. Next, there are two bibliographies on database security by G. Pernul and G. Luef, and on Object-Orientation and Deduction by S. Conrad and M. Gogolla.

The regular invited sections include the following material. The section on Database Research Centers presents two reports, the first on database research at IPSI, and the second on database research activities at the University of Massachusetts, Amherst. The section on Database Research Funding by Marianne Winslett includes information of interest to researchers and developers in the database area.

This issue is concluded with calls for papers and participation including a summary of upcoming database conferences prepared by Felix Saltor, and a set of pictures from 1991 SIGMOD conference prepared by Fred Spingsteel.

As a final note, I call the attention of the readers to the new SIGMOD Innovation Award whose details are given next page.

Arie Segev
January 1992
At the end of 1991, ACM SIGMOD introduced its "SIGMOD Innovations Award." This award is envisioned to be the first in a new SIGMOD Awards program. This note is intended to inform the database community of the existence of this annual award, and to encourage the nomination of candidates for the award.

DETAILS OF THE AWARD

1. Name: SIGMOD Innovations Award.
2. Given: Annually (if there is at least one qualified candidate).
3. Award: $1,000 plus a plaque per person (or group, if it is for a group).
4. For What:
   Innovative contribution to the development or use of database systems and databases; the contribution must have been reduced to practice and adopted widely in significant use.
5. Administration:
   Administered by the SIGMOD Awards Committee.
6. Nomination/Evaluation Procedures:
   - Anyone in the field can nominate one or more persons or group.
   - Nominations should include a 200-500 word statement to justify the nomination.
   - Contributions should date back no more than 10 years.
   - The nomination procedure, etc., will be published in SIGMOD RECORD once or twice a year.
   - The Awards Committee will evaluate all nominations and decide on zero or more winners.
   - Nominations must be received by March 1st to be considered for that year's award.
7. Recipient:
   The recipient will receive the award during the opening ceremony at the annual ACM SIGMOD/PODS Conference; he/she will give a short speech (5-10 minutes).

WHERE TO SEND NOMINATIONS

Nominations should be submitted to (either via e-mail or postal mail) to one or more members of the SIGMOD Awards Committee, and copied to the ACM SIGMOD chair.

SIGMOD Awards Committee:

Prof. Hector Garcia-Molina
Dept. of Computer Science
Stanford University
Stanford, Calif. 94305
hector@earth.stanford.edu

Prof. Dennis McCleod
Dept. of Computer Science
University of Southern California
Los Angeles, Calif. 90007
mcleod@vaxa.isi.edu

Prof. Peter Scheuermann
Dept. of Electrical Engineering and Computer Science
Northwestern University
Evanston, Illinois 60201
peters@eecs.nwu.edu

SIGMOD Chair:

Dr. Won Kim
UniSQL, Inc.
9390 Research Blvd.
Austin, Texas 78759
executive@unisql!kim@cs.utexas.edu
ADVANCED USER INTERFACES FOR DATABASE SYSTEMS
Letter from the special issue editor

It is often stated that the three most important factors that determine the success or failure of a database system are performance, performance, performance! The experience of the last twenty years with relational systems has shown that at least one of these three references to performance implies that of end-users when interacting with the system to access data, i.e., user productivity. Although declarative query languages like SQL and QUEL represent major improvements over procedural programming languages like COBOL, the overall consensus is that they are too complex for many users. The need for more intuitive and easier to learn and use interfaces to database systems is always current.

In fact, in the past few years, this need has become even more pressing for three reasons. First, database systems that are based on more complex data models than the relational, e.g., object-oriented, have become very popular and several such prototypes and commercial products have appeared in the scene. Given the underlying richness of these models, declarative languages for them tend to be even more complex than SQL. Second, even within the relational world, more complex types of data are being stored in the database, e.g., rules or CAD designs, and more complex queries are asked of them, e.g., recursive queries. The result is again an increase in the complexity of the necessary query languages. Third, the community of potential users of database systems has grown well beyond the traditional administrative business professional, and now includes people that are even less familiar with the standard ways of approaching data organization and data access, e.g., researchers from other sciences. For most of them, twisting their way of thinking so that it fits that of the available systems is simply something on which they are not willing to spend time.

For all the above reasons, it has become evident that more advanced interfaces are imperative for database systems if they are to successfully face the future demands. The papers in this issue deal with some of the key research ideas that are currently being pursued in this direction. They all describe prototype or commercial systems that are currently available or under development. As expected, graphics play a very important role in most of these systems, but several other orthogonal approaches or issues are being discussed as well. I hope that this collection of papers will spur the interest of others as well to identify interesting problems and work on advancing the state of the art on user interfaces for database systems. Below I give a brief overview of the essence of each paper.

Rowe sets the stage by giving a tour through four application development systems with which he has been involved for the past fifteen years, describing how the limitations of each one led to the next.

The next four papers deal with object-oriented database systems. King and Novak present an overview of Facekit, a toolkit for designing graphical interfaces to object-oriented database systems, and focus on the techniques used in that system for specifying graphical representations of database objects that are stored in the database itself as methods.

Flynn and Maier take a similar approach in the Object Display Definition System, where display information for complex objects is a complex object itself that is stored in the database and managed by that system instead of the individual applications.

Paredaens et al. describe the GOOD system, which offers graph-based database manipulations for object-oriented systems based on graph-pattern matchings, and its future extensions based on the Tarski data model.

Borras et al. discuss two user interfaces that have been developed for the O2 database system, the O2Look commercial product, whose emphasis is on creating and customizing graphical representations of complex objects, and the ToonMaker prototype, which allows such customization to happen interactively.

Consens et al. summarize the key contributions of the G+ and GraphLog systems, which offer the ability to both present labeled graphs and ask recursive queries on them graphically, and then discuss some of the foundations of extending these ideas to developing visual languages.

Ioannidis et al. describe their efforts to build a database system to provide support for the management of scientific experiments and data, focusing on the graphical user interface of that system.

Motro presents a user interface to a relational database system that qualifies its answers by providing to the user additional useful information.

Finally, Kuntz discusses the GIUKU user interface, which provides several features that help knowledgeable users in their interaction with the database system.

The authors of all these papers have put much effort in preparing them. I want to thank them for the excellent work that they did to make this issue a reality.

Yannis E. Ioannidis
University of Wisconsin - Madison
January 1992