Introduction to the Special Issue on Database Research

Martin L. Kersten
CWI
P.O. Box 4079, 1009 AB Amsterdam, The Netherlands

1 Introduction

The CWI Database research group was established in 1985 to study topics in the area of distributed information systems with a focus on the theory of datamodels and architectures for database management systems. In particular, those research issues stemming from requirements posed by information systems which are distributed over time and location, where the applications change frequently, and where the database is only guaranteed to be locally consistent. Such information systems can be found in office environments, (financial) trading environments, in design environments, and scientific environments. There, the kinds of data being collected can not be frozen for long periods and the way in which the data is being used to produce information changes frequently.

Since its inception, the group is organized around projects, which study the problems from complementary perspectives. Moreover, this organization maintains a balance between theoretical and experimental computer science research.

During the first six years, we have been involved in the design of an object-oriented datamodel based on Category Theory and the design and implementation of a Distributed Main-Memory Database Machine.

In the coming years, our research efforts will be further focussed on the object-oriented approach, such as the formalization of an object-oriented data- base design, the development and the evaluation of efficient algorithms for object-oriented databases in a distributed environment, and the storage management of replicated objects in a distributed system.

2 Research 1985-1990

Research in this period centered around several projects: PRISMA, TROPICS, and Floc. They are discussed shortly below, while an in-depth description of PRISMA can be found elsewhere in this issue.

2.1 PRISMA and TROPICS

The PRISMA project (1986-1990) was a large-scale research effort (30 persons) in the design and implementation of a highly parallel machine for data and knowledge processing. It was organized as a nationwide Dutch research activity with combined forces from four universities, a governmental research institute, and Philips Research Laboratories.

During these years considerable effort was spent by our members in the PRISMA team on the design and prototyping of the database system, which resulted in ca. 45K lines of documented POOL code developed together with our partners at Twente University. By the end of 1988 a stable interpreter for the programming language POOL became available, which was used to finalize the prototype PRISMA/DB in the spring of 1989. This first implementation round was finished with a two-day workshop to evaluate the code produced and to assess future enhancements.

Awaiting the delivery of the POOL compiler for the multi-processor machine, we spent our time on research issues related to the PRISMA database system, such as an evaluation of the programming language POOL for its expressiveness of relational database concepts at the level of the One-fragment manager, database recovery, and the semantics of nested queries in SQL. In 1990, we had to rework major portions to make the system run on the 100-node multiprocessor.

The ESPRIT-II project TROPICS ran from 1989 till June 1990. One of the major themes in this project was to enhance the PRISMA database system to become a viable platform for carthographic and office systems. Therefore, our group has developed an extended version of SQL and implemented a prototype for feasibility studies. After termination of this project, we refocussed our effort on active databases.

2.2 An object-oriented datamodel

The PhD project Floc ran from 1985 till 1989. It aimed at the formalization of object-oriented datamodels, in particular the structural parts of these models. As category theory is the mathematical theory of structures, it was chosen as vehicle for the development of the formalism called Floc. During this project, category theory proved to be more than adequate to formalise both the static and the dynamic structure, including constraints, of object-oriented datamodels. Moreover, it allowed for the inclusion of negative and disjunctive information as well as the development of a, deductive, query language for Floc.

The main theme for the design theory was on structural equivalence, for which category theory proved to be, again, indispensable. Briefly, this allows a database designer to switch between alternative, but equivalent, models of the same Universe of Discourse. For example, it was shown that traditional constraints such as functional, multi-valued and join-dependencies can also be modelled by a suitable type construction and a (simple) dynamic constraint.

3 Current and Future Activities

The projects currently underway extend the results on database theory and architectures of parallel database systems.

3.1 The ISDF project

The Integrated Systems Design Framework (ISDF) project (1990-1993) is a research collaboration between several Dutch universities and funded by NFI. The overall goal of this project is the development of an integrated framework for the description of information systems and the analysis of information systems descriptions within this framework. The importance of such a framework lies in the outlook it offers for the development of truly integrated information systems design methodologies, which apply hitherto largely unused mathematical theories for the analysis of their design. In particular, it provides a sound theoretical basis for such methodologies to formulate requirements and logical designs, and it forms a reference point for the accompanying software development tools.

The task of our group is, in close collaboration with Twente University, to develop a formalism which is suitable for both:

1) the expression of the design concepts within the framework, such as specialization and generalization;

2) the analysis of a design with regard to structure, dynamics and constraints, for problems such as structure equivalences, lifeness/deadlock and consistency, as well as the interaction between these areas.

This project is elucidated with a paper in this issue. The design-theory for object oriented databases is now focussed on the normalization of class-hierarchies.

3.2 The Starfish project

As computers become less expensive and more numerous, many organizations find that they have many computers that need to co-operate. This requires a single, uniform, operating system that allows all the computers to work together in a seamless way. Processes on any machine must be able to communicate with remote processes the same way they communicate with local ones, in order to make it possible to reconfigure the system.

Similarly, there should be a single way to access a file, whether that file happens to be on the local machine, on a file server in the same building, or in a distant city. In general, the physical location of processes, data, and all resources should be *transparent*, with access handled automatically and efficiently by the operating system.

Within the Starfish project, a collaborative action of CWI, Free University of Amsterdam, University of Twente and University of Amsterdam, we believe that transparency should be achieved by running the same operating system, possibly on different kinds of hardware. The system chosen is *Amoeba*, a distributed operating system designed and implemented at the Free University of Amsterdam and CWI.

On top of this operating system we need several generic application systems, that provide the required functionality and that exploit the distributed platform efficiently. Therefore, a prime objective is to develop an *extensible complex object server*, called Goblin, which provides the user a datamodel and query language to describe and manipulate graph-like objects. The task of the object server is to efficiently implement this model under the *Amoeba* system, shielding the complexity of maintaining the fragmented representation through clustering, indexing, and query evaluation strategies.

The research actions focus on the software architecture of an extensible complex object server with an emphasis on dynamic query load balancing and active database support. To illustrate, an active database system is characterized by a set of event-condition-action pairs, which describe actions to be taken upon encountering an event in a particular database state. The correctness criterion for transaction management, i.e. serializability, is a serious handicap in achieving both a more efficient system and to limit the modelling of the cooperative behavior of autonomous systems. Well-founded execution models and efficient algorithms are needed in both research areas mentioned.

In addition, research questions related to efficient WAN communication protocols, fault tolerance, data distribution, language interworking, and application of the system in robotics are being addressed by the partners.

3.3 The future

Although the two afore-mentioned projects will run for a few years to come, the long term research plan of the database research group is taking shape. It centers around interoperability problems of information systems, performance quality assessment and prediction of advanced database systems, and the application of database technology to other fields of research, such as the requirements posed by scientific database systems on database programming languages. Funding for these strands of research are being sought within the context of ESPRIT and national programmes.