Database Research at the University of Twente

Principal researchers: P.M.G. Apers, H. Balsters, H.M. Blanken, R.A. de By, M.A.W. Houtsma\textsuperscript{1} (ed.), P.A.C. Verkoulen, A.N. Wilschut.

1 Introduction

The University of Twente is one of the leading technical universities in the Netherlands. It is the only Dutch university that has an independent computer science department. The database research group is comprised of approximately 15 researchers and graduate students. It is participating in a number of externally funded, national and international research projects, and has done so in the recent past.

The research interests of the database group are in the field of object-oriented, logical, and extended relational systems. The three main points of interest are: i) development of well-founded (object-oriented) data models and tools, ii) design of and optimization in database systems, with a strong emphasis on distribution aspects, and iii) database support for new (technical) applications.

In the following, we will give a brief description of our main research efforts, and mention some of the associated publications.

2 Data Models and Tools

Principal researchers: Balsters, de By.

Research on data models is done in the context of TM, an object-oriented model. Research on tools focuses on database design tools based on TM.

2.1 TM

TM is an object-oriented model that includes standard object-oriented features, such as arbitrarily nested attributes, methods, and inheritance. Specification of methods is done in a functional language, and set comprehension may be used for the description of, e.g., attributes. TM also allows for constraint specification in a first order logic like language, and constraints and methods are subject to inheritance, just like attributes.

The initial work on TM focused on providing a strong theoretical foundation for such an object-oriented model; it was done in the context of the ISDF-project. The theory is described in a formal language called FM, which is based on the Cardelli type system, set theory, and lambda calculus.

Recently, a logical query language called DTL has been developed for TM. It allows a declarative formulation of ad-hoc queries on a TM database specification.

A next step is to study the integration of LOTOS, a protocol specification language, and TM, with the goal of describing distributed transactions, e.g., in the context of groupware. This will be undertaken in the TRANSCOOP-project (ESPRIT III-BRA 8012). The TransCoop project aims at providing computer support for cooperative applications. In particular, the language based upon TM and LOTOS will be used to specify such cooperative transactions. The language will be mapped onto an existing transaction management system. Together with some additional tool support, this will give us an implementation of the TM/LOTOS language. Some specific applications like Design for Manufacturing (DFM) will be used to check whether the integrated language suits its purposes. The theory behind the integration of TM and LOTOS will be further studied in a project supported by NWO\textsuperscript{2}.

Rene Bal and Herman Balsters, “A Deductive and Typed Object-Oriented Language”, Proc. Third Int. Conf. on Deductive and Object-Oriented Databases, December 6-8, 1993, Scottsdale, Arizona.


Herman Balsters, Rolf A. de By, and Roberto Zicari, “Typed Sets as a Basis for Object-Oriented Database Schemas,” in Proc. European Conf. on Object Oriented Programming, July, 1993, LNCS #707 pp. 161–

\textsuperscript{1}The research of Maurice Houtsma has been made possible by a fellowship of the Royal Netherlands Academy of Arts and Sciences

\textsuperscript{2}A Dutch organization comparable to NSF in the USA


2.2 Tools

It is important that tools are developed that effectively support the users of the models. We are, therefore, focusing on the development and implementation of such tools.

A design tool for TM has been built that provides a graphical representation of TM specifications. This design tool is methodology independent, and hence gives a database designer the freedom to use his/her own favorite methodology. The tool is currently extended to include algorithms to analyze the specifications concerning internal consistency, safety, etc. We are also working on translating TM to existing platforms (OO databases, DBPLs).

Another notable tool is the so-called prototype generator (PG). The PG generates a quick-and-dirty prototype of a TM specification. It allows the user to fill the database with some test data and to perform constraint/method evaluations and analyze the results. The architecture of this tool is set up in such a way as to allow for prototyping in different target languages.

The aforementioned tools are some of the main contributions of our database group to the IMPRESS project (ESPRIT 6355). The IMPRESS project aims at obtaining a reliable, distributed object server for technical and multimedia applications.

3 Systems and Optimization

**Principal researchers:** Apers, Blanken, Houtsma, Wilschut

Research on systems and optimization has for a major part been done in the context of PRISMA, a parallel database project. Other optimization issues that have been studied are recursive queries, replicated data (in cooperation with the Favo-project at Stanford University), complex object support, and physical database design.

3.1 PRISMA

The PRISMA project has been a large-scale cooperation between several Dutch universities and Philips Research Laboratories. PRISMA is a relational database, implemented on a tightly-coupled distributed system (100 nodes). Goal of the PRISMA project has been to gain insight into the use of parallelism and main memory databases, and the applicability of query optimization algorithms in such an environment.

At the start of the project, the choice was made to develop a traditional relational database system and focus on parallelism and the use of main memory. Although an object-oriented language was used to program the system, the database itself is fully relational and its design was based on traditional distributed database system technology. Main idea was to have components called One-Fragment Managers as basic relational engines, locate these on separate processors, and then make use of parallelism for processing queries. Focus of the optimization was improvement of query response time.

A lot of effort was put into building as much flexibility as possible in the PRISMA prototype. This, to make it an experimental vehicle for studying a large variety of issues in the area of parallel query processing. An experimental study of parallel execution of simple relational joins between large relations, revealed that the main-memory character of the system significantly influences its parallel behavior. As main-memory processing speeds up local processing but not communication, the communication between processors becomes a bottleneck. As a result, the optimal degree of parallelism to be used for simple relational queries is lower than the number of processors in the system.

The results of this work and of a theoretical study of the use of pipelined parallelism, are currently used in a study of parallel execution of multi-join queries. In this research, we study how the available computing power can be allocated to the individual joins in the query to minimize response time.


Peter M.G. Apers, Carol A. van den Berg, Jan Flostra, Paul W.P.J. Grefen, Martin L. Kensten and Annita N. Wilschut, “PRISMA/DB: A Parallel Main-Memory Relational DBMS” in *IEEE Transactions on*
3.2 Recursion

Initially, work in this project was centered around a rewriting strategy that transforms general recursive queries into regular Relational Algebra programs plus transitive closure operations. Later on, focus shifted to efficient use of parallelism to compute transitive closure problems such as shortest path and bill-of-material. Recently, an extensive survey on these topics has been published.

Important aspects of this project are the development of data fragmentation strategies, and effective use of small amounts of precomputed information to speed up query processing. This enables to transform a transitive closure operation on a huge amount of data, into a number of transitive closure operations, each on only a small fraction of the data. These can then be executed in parallel, after which a final join is performed to generate the final results.

Recently, experiments have been set up to test the developed approach (called Disconnection Set Approach) on the PRISMA database system. Results of these experiments are very promising, as a nearly linear speedup is achieved. Next step will be to implement other approaches, do a comparative study, and analyze the performance for several data characteristics. This will hopefully give insight into the applicability of the different approaches, for different data characteristics.


Maurice A.W. Houtsma, Peter M.G. Apers, and Stefano Ceri, “Complex transitive closure queries on a fragmented graph” in Proc. 3rd Int. Conf. on Database Theory, LNCS 470, Springer-Verlag, 1990, pp.470–484.

Maurice A.W. Houtsma, Peter M.G. Apers, and Stefano Ceri, “Disconnection Set approach” in Proc. 16th Int. Conf. on Very Large Data Bases, Brisbane, Australia, Aug. 1990, pp. 335–346.

3.3 Replicated Data

Work in this project stems from a cooperation with the Fauve project at Stanford University. Emphasis is on update behaviour in replicated databases. First of all, a classification of update strategies has been developed, that classifies existing algorithms according to aspects such as treatment of copies, propagation of updates, behaviour in case of failures, etc. Work has been done on a specific direction, so-called independent updates; replicas accept arbitrary updates, and only later the problem of bringing the database to a global consistent state is addressed. Currently, we are
investigating the application of independent updates in the area of mobile computing.


3.4 COD

The goal of the Complex Object Database project is to gain better insight into the representation, manipulation, and storage of complex objects—both in a central and in a distributed environment. The applications that have been studied include CAD/CAM, multimedia, and cartography. Emphasis has been on the implementation of a storage server for complex objects, and on experiments with joining such objects. For the implementation, the Amoeba distributed operating system has been used. Some of this work has been part of the Starfish project, supported by NWO.

Another aspect of this project concerns query optimization in complex object models. Work is being done on translation of an SQL-like query language for the object-oriented data model TM into an algebra for complex objects based on the $NF_2$ algebra, and on logical optimization of algebraic expressions. The emphasis is on translation and optimization of nested queries (involving, for example, quantifiers, iteration over set-valued attributes and nested results).

The general idea is to use the work done in (extended) relational query optimization whenever possible, and to extend and adapt it whenever necessary. We assumed that, also in the implementation of an object-oriented data model, a set-oriented query processing model is a valid model. Joins (possibly speeded up by physical pointers) are, also in complex object models, still considered to be important operations.

An important research question is how to handle joins involving set-valued attributes, without unnesting.


Hennie Steenhagen and Peter M.G. Apers, Implementation of the Object-Oriented Data Model TM, Dagstuhl Proceedings, Morgan Kaufmann.


3.5 Physical database design

Main emphasis in this project is on the development of tools and techniques for physical database design. DBMSs offer storage and access structures, like hashing and indexes, to allow storage of data and fast processing of queries and updates. Given, e.g., a query and a physical design, a DBMS-component called the optimizer will select a fast way of handling the query. During physical design a database administrator has to decide which structures to use for a specific database and a specific load. The aim is, of course, to achieve an efficient processing of the expected load on the database. Many choices have to be made, and usually heuristics are used to obtain a good physical design. Current research focuses on development of a tool that combines heuristic rules and interaction with the optimizer to achieve a good physical design. The tool is developed for a specific relational DBMS, however, attention will be paid to generality, so that the techniques may be useful for other DBMSs too.

Sunil R. Choenni, Elisa Bertino, Henk M. Blanken and Thiel Chang, “On the Selection of Optimal Index Configurations in Object-Oriented Databases,” in Proc. Int. Conf. on Data Engineering’94.

4 Database Applications

Principal researchers: Apers, Blanken, Houtsma

Besides traditional administrative environments, databases are being used more and more in different application environments. These environments demand new capabilities of database systems. Moreover, many design decisions that have been made in the development of database systems, have to be reconsidered to make these systems useful in new application areas.

New demands deal, for instance, with the introduction of new data types and more powerful query languages. Design decisions that need to be reconsidered include the traditional transaction concept and query optimization techniques. Because of the enormous need for computing power, we feel that distributed processing (parallelism) will be essential for these new application areas.

Currently, we are mainly focusing on Geographic Information Systems (GIS) as a relevant application area. Optimization of queries in this environment is one of the challenging problems here. Furthermore, we are taking into account possibilities in areas such as multimedia, interoperability, and data mining.


More information

More information on the work of the database group, a complete list of publications, or copies of specific