

DATABASE GROUP RESEARCH ACTIVITIES

"I have no data yet. It is a capital mistake to theorize before one has data."

Arthur Conan Doyle

Adventures of Sherlock Holmes

Adventure 1. — A Scandal in Bohemia

The Database Group at the University of Waterloo is one of the largest such groups with eight core faculty, six associated faculty and over thirty graduate students. It is also a very productive group; since 2001 members of the group have authored 99 refereed publications. Of these, 25 have appeared in journals, and 74 in conferences, symposia, and workshops. In addition there are 5 book chapters, encyclopedia entries and handbook chapters. An increasing number of the refereed publications have been appearing in top ranked venues: for this period, 43% have appeared in these venues, but the ratio increases to 52% and 60% in 2004 and 2005, respectively. During this period, group members have given six keynote addresses, five tutorials at conferences, have been applicants of eight patents (three granted) and edited four books and proceedings. A total of 27 graduate students have completed their studies since 2001 (8 PhD, 19 MMath) and have started their careers at various universities (Hong Kong University of Science and Technology, Indiana University, University of Manitoba) and industry (Microsoft, iAnywhere, IBM, Alcatel, Samsung, MKS, Siebel Systems, and others).

Group members have also been active members of the scientific community. They have served on the editorial boards of 6 journals (including one as an Editor-in-Chief), chaired one conference, acted as the program committee chair or vice-chair of nine international conferences and have served on a total of 98 program committees. In addition, our members have served on the Board of Computer Research Association Board, chaired the ACM Special Interest Group on Management of Data (SIGMOD), as trustee of the VLDB Endowment, and as member of the ACM Publications Board as well as participating in several research management committees, research award committees, and similar committees.

An identifying characteristic of the group is its broad-based research program addressing both extending and enhancing database management systems (DBMS), and data management for new domains. The research focuses on data management issues as they apply to complex objects, geometric and spatial data, video and audio data, historical data, structured documents and text, and real-time data as well as to traditional relational database systems. This wide coverage mirrors the expanding role of database systems in managing very large amounts of diverse data. The research also has wide coverage of many aspects of database system functionality. Consequently, there are ongoing projects that explore storage structures, indexing methods, query languages and optimizers, transaction management, and distributed data management.

A second identifying characteristic of our work is its collaborative nature. Ongoing collaborations span academic institutions (Hong Kong University of Science and Technol-

ogy, National University of Singapore, New Jersey Institute of Technology, Purdue University, University of Illinois at Urbana-Champaign, Université de Nantes, University of Toronto, Queen's University, Free University of Bozen-Bolzano, University of Roma La Sapienza, Technical University of Dresden, University of Calabria, University at Buffalo, University of British Columbia, University of Illinois at Urbana-Champaign, Purdue University, University of Toronto, Carnegie Mellon University, University of Jyväskylä), research laboratories (IBM Almaden Research Laboratory, INRIA, National Informatics Institute of Japan, AT&T Research), and industry (IBM Toronto Laboratory, Sun Microsystems, Nortel Networks, Open Text Corporation, Sybase, Hewlett-Packard). Further information about the group is available at <http://db.uwaterloo.ca>.

Existing work in the group can be categorized along two dimensions: extending and enhancing database systems (advanced query processing, adaptive and self-managing DBMS, security in databases, distributed data management), and data management in new domains (embedded database systems, Internet-scale distribution, XML data management, spatial, temporal and multimedia systems). Our work in these areas are summarized below.

1. **Advanced Query Processing** (Ihab F. Ilyas)

A key feature of a DBMS is its ability to process high-level declarative queries, specifying *what* the desired query result should be. Emerging applications that require database support (e.g., multimedia systems, Web databases) incorporate notions of proximity, ranking, and user feedback that have serious impact on the query model, which cannot be adequately handled by current query processing approaches embodied in traditional DBMSs. Our work focuses on enriching query processors/optimizers to handle Information Retrieval (IR) style queries that are common in these applications. These queries do not have exact answers, requiring the ranking of the results according to their suitability as an answer. Some examples of these types of queries are the following. The goal of this line of research is to develop a formal and generalized ranked-retrieval framework that adapts to non-traditional processing environments and is applicable to a variety of data models. We are primarily interested in developing algorithms and frameworks to leverage the current capabilities of database systems to handle ranked retrieval efficiently and adaptively.

2. **Adaptive and Self-Managing DBMS** (Ashraf Aboulnaga, Ihab F. Ilyas, Kenneth Salem)

As DBMSs have become more sophisticated, and as modern computing environments have become more dynamic, experiencing data fluctuations and unexpected frequent delays, it has become more difficult to tune them for particular environments. This has increased the interest in the development of adaptive and self-managing DBMSs that can better cope with the environmental changes and simplify database administration.

We address a number of aspects of this problem. One project aims to improve the query optimizer statistics and cost estimates by observing the estimation errors and learning from it. We use feedback from query execution to improve statistics and cost estimation. We concentrate on quantifying confidence in the statistics and the estimates that they produce, and on using this confidence to improve query optimization.

A related project studies ways of enhancing the capabilities for current query optimization and execution techniques to cope with the continuously changing computing environments. In particular, we investigate developing query optimization techniques that are highly adaptive to changes in the computing environment and to lack of accurate and up-to-date statistics.

A novel aspect of our work in this area involves the relationship between DBMSs and storage systems. Database management systems have traditionally stored data on locally attached storage, which they manage either directly or through a local file system. Modern storage systems provide *storage virtualization*, which hides physical storage devices behind an abstraction layer. Building a DBMS on top of this virtualization, while providing significant benefits, poses a number of challenges, which are the subject of study in this project.

3. **Distributed Data Management** (Ihab F. Ilyas, M. Tamer Özsu, Kenneth Salem)

Database systems have been addressing the challenges of distributed computing from the very early days. The distributed computing environment has changed significantly over the years. Client/server systems are still prevalent, but cluster-based parallel environments and very wide-scale distributed systems have become prevalent spurred by the expansion of Internet (in particular the peer-to-peer (P2P) systems). Our group focuses on several topics that arise in managing data in each of these environments.

One project focuses on building scalable database systems using replication. To scale a system up, data are replicated and the growing workload is distributed over the replicas. However, unless the replicas are somehow synchronized, applications may not see a consistent view of the database, or they may see a stale view. We approach the replica synchronization problem using the notion of application *sessions*, which are sequences of related database requests. Our approach provides strong consistency and freshness guarantees within a session, but weaker guarantees across sessions.

A second project studies methods to optimize database client/server interactions. Relational database applications establish database server connections through which they issue streams of query and update requests and fetch the results of those requests. It is common for application request streams to include many small requests. For each such request, there is significant latency and overhead associated with the client/server interconnection network and the layers of system interface and communications software at both ends of the connection. The overhead can be significant, in many cases dominating the total cost of the application request. The goal of this work is to reduce this overhead to improve client/server performance.

In several projects we are investigating the data management issues in Internet-scale (i.e., very large scale) distributed environments. One of these projects focuses on the management of *stream data*. A growing list of emerging applications receives and process data as a sequence (stream) of items (e.g., financial tickers and other on-line Web information sources, Internet traffic measurement). These applications have database requirements that are significantly different than traditional DBMSs in that instead of executing (generally) transient queries on persistent data, they execute persistent (continuous) queries over transient data in real time. The objective of this project is to study query languages and query processing issues over data streams.

Peer-to-Peer (P2P) networks emerge as the next-generation Internet infrastructure, which are special in their scale, dynamism, and heterogeneity. These characteristics force the reconsideration of many of the data management problems and their solutions. We are working on an interesting and important problem within this context, namely rank-aware query processing (i.e., top-k queries) in P2P systems.

4. **Embedded DBMSs** (David Toman, Grant Weddell)

Many devices these days manage significant amounts of data (e.g., telecommunications equipment, PDAs, etc). Current practice is to design and develop special purpose data management software using low-level languages such as C, ignoring database management technology that has proven to reduce the “cost” of data management in traditional applications. This research program is guided by the hypothesis that savings in the cost of developing and maintaining embedded software, would also ensue from using database technology. However, important enhancements are necessary in current database technology. We focus on topics that underlie three of these areas. First, there is a need for improving query optimization. As suggested, query plans generated by query optimizers must be comparable in performance to low level code written directly by expert programmers. Second, the unique properties of embedded systems that affect the indexing and encoding of data have to be taken into consideration via enhancement in schema languages and reasoning about integrity constraints. Third, there is a need for alternative transaction models. For embedded software, subsystems implementing different features can have undesirable interactions that are not prevented by following traditional concurrency control protocols. To this end, we concentrate on advanced concurrency protocols, and on incorporating dynamic knowledge in database schema.

5. **Security in Databases** (Kenneth Salem, Frank W. Tompa)

Despite their rapidly increasing importance, XML and object DBMSs do not have a standardized and well-understood access control model. This reduces their practicality in the commercial world. Access controls determine which users are allowed to query and update the information that is stored in a database. Our goal is to develop a well-defined, efficiently implementable, fine-grained (i.e., at the element level) access control model for XML and object-oriented data against which a variety of operations or methods are to be applied. We are investigating both the methods to specify access controls for such data, and techniques for efficiently enforcing and maintaining the controls in the presence of thousands of simultaneous users on very large data collections.

6. **Spatial, Temporal and Multimedia Databases** (Edward Chan, Ihab F. Ilyas, M. Tamer Özsu, David Toman)

Traditional databases store relatively simple data and are snapshots of data at a certain point in time. Large classes of applications (e.g., geographic information systems, location-based services, graphic and simulation systems, historical data warehouses, multimedia systems) require the management of data about objects that have more sophisticated characteristics: they are spatial (location), they evolve over time (temporal), the objects move, and objects may consist of multiple media types (video,

audio, text, images). Our research addresses the development of database technology to fulfill these requirements.

One project focuses on indexing structures and query evaluation algorithms for spatial databases. We focus on *buffer queries*, which find pairs of objects from two data sets whose “distance” is within a threshold, and *route queries* that ask routing queries over a large network. For buffer queries, we work on efficient spatial-join algorithms. For route queries, we focus on shortest path (SP) calculations between two points, in particular on disk-based SP algorithms due to the size of the network.

In related work we investigate generalized tools that assist implementation of spatial index structures. We proposed and implemented a spatial index generalization framework, SP-GiST. The framework identifies one core implementation of a wide class of spatial indexes and a simple user-defined interface to realize the individual structures.

In moving object databases, we focus on indexing issues as well, considering two types of databases: *historical*, which deal with the past movement or trajectory of objects, and *predictive*, where the location of a moving object is constantly updated to reflect its movement. For historical databases, we propose an index, called TSL-trees, that is scalable, update-efficient, and query-efficient for various query classes, including range and trajectory-based queries. For predictive databases, we are investigating ways to allow a large number of updates of the object locations as they move.

Related to historical databases, we are investigating the issue of similarity search where a database is searched to find and retrieve those trajectories that are “similar” to some sample query trajectory. This problem arises in an important class of applications. Our research focuses on two types of queries: *pattern existence queries* that retrieve trajectories that contain a query trajectory pattern regardless of the location of the pattern in the trajectory, and *shape match queries* in which trajectories that have a movement shape similar to the query pattern are retrieved (the location of the shape is important). We focus on two key issues, namely the design of a distance function and the development of techniques to improve the retrieval efficiency in the presence of data impurities (local time shifting and noise).

Our work in temporal databases focuses on the problem of data change over time and becoming out-of-date. In particular, despite advances in storage technology, the accumulation of large amounts of data over time can easily exceed the ability of the application or the application users/owners to store the accumulating data in perpetuity. Also, even if the growth of the actual physical storage systems for the data was able to keep up with the data accumulation, the sheer volume of data that needs to be examined by information requests (queries and updates) of an application can often lead to an unacceptable performance deterioration. The goal of our research is to study *logical approaches* to data expiration: approaches that, based on an analysis of information requests of applications, *automatically* detect what parts of the data set can be safely removed.

Finally, we have ongoing work on video databases. We investigate issues related to the modeling, representation and efficient retrieval of video objects.

7. **XML Data Management** (Ashraf Aboulnaga, Ihab F. Ilyas, M. Tamer Özsu, David Toman, Frank W. Tompa)

XML is rapidly being adopted as a standard syntax for describing the structure of data and for encoding structured or semistructured data in a manner that facilitates interchange. XML can be viewed simply as an exchange form, but our interest concentrates more heavily on studying XML as an encoding form for persistent data.

We have several projects that address XML data management issues from multiple perspectives. Most of our projects is driven primarily by *document-centric applications*, as opposed to conventional business (i.e., relational) data wrapped in XML. A long-term objective of this project is to support document storage and management by applying sound database principles to text management and by designing and prototyping suitable document database systems. The challenge is to discover how the complexity of text, with its intricate structure and diversity of expression, can be efficiently managed. One specific project along these lines is to provide information fragments as *virtual texts* (“text views”) that are assembled dynamically to meet various applications' needs.

Two other projects focus on optimizing the execution of XQuery and XPath, which have emerged as the de-facto standard query languages for querying and manipulating XML documents. There are two main approaches to this problem: by mapping XML data to a relational database and XQuery queries to SQL, and developing native XML DBMSs. Along the first dimension, we use a novel relational encoding of XML documents, called *dynamic interval* encoding, that allows a large fragment of XQuery, including queries that use arbitrarily-nested FLWR expressions, element constructors, many of XQuery's built-in functions, and structural comparisons to be efficiently handled using relational-style query execution engine. The technique enables (suitably enhanced) relational engines to produce predictably good query plans that do not restrict the use of algorithmically preferable query operators.

Along the second line of evaluating XPath and XQuery, we start from scratch in building a native XML database. We design storage structures for XML documents such that they are able to store terabytes of XML data, support efficient query evaluation, and are easy to update. We investigate efficient physical operators for answering XPath queries on this storage structure. We work on cardinality estimation, which is crucial to cost-based optimization. Cardinality estimation of path expressions is usually based on a graph-based synopsis structure that summarizes the XML tree and maintains sufficient statistics. We developed a synopsis structure, called XSeed, that is accurate and fast for cardinality estimation, using our novel estimation algorithm. We are investigating indexing techniques and the development of a full-fledged XQuery query optimizer.

Orthogonal to these three projects that focus on the efficient management of XML data, we also work on benchmarks for evaluating the performance of XML DBMSs. We have developed XBench, which is a family of benchmarks that capture different XML application characteristics.