RESEARCH IN KNOWLEDGE BASE MANAGEMENT SYSTEMS

Gio Wiederhold
S. Jerrold Kaplan
Stanford University

Daniel Sagalowicz
SRI International

I INTRODUCTION

The Knowledge Based Management Systems (KBMS) Project * addresses the problems of intelligent processing in large databases. Many research projects, commercial enterprises, and government agencies maintain large databases. The existence of these databases is often due to operational requirements like the monitoring of project progress, the management of inventories of the resources for the enterprise, and the need for assessing the alternative ways that resources can be used in the future. High levels of reliability of computers and communication, adequate high level languages, and comprehensive operating systems have made operational use of databases nearly routine. The use of the knowledge contained within data as a high level resource in management and planning has not yet been routinely achieved. In order to advance the state of the art we are turning to techniques which have proven themselves in artificial intelligence systems, and are developing these techniques for very large databases.

---

* This research has been partially supported by the Defense Advanced Research Projects Agency under MDA903-77-C-0322 and N000-39-80-G-0132. The views and conclusions contained in this document are those of the authors and should not be interpreted as representative of the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency of the U.S. Government.

Figure 1 presents the architecture of a candidate KBMS. The various modules in this architecture correspond to the various research topics that we are interested in. This system structure is hence not intended as a modularization for a database implementation, but rather as a framework to define the scope and interfaces of topics that show promise for research and analysis. The next sections present those ideas which have been or are currently being investigated.

II TASK MODELS

Certain entities in a database -- be they fields, attributes, relations, or more complicated constructs -- have certain a priori probabilities of being required in an interrogation of the database. These probabilities are dependent on the particular task a user may be performing, and his or her focus and interests. By incorporating a task model into the KBMS, these characteristics can be used in a variety of ways to improve the utility of the system. Four main uses of task models are being explored in a KBMS: (a) inclusion of relevant information not explicitly requested in the response to a query; (b) organizing responses so that the more "interesting" items are presented first; (c) checking for semantic irregularities in the performance of the task; and (d) prefetching of items and fields that have not yet been requested but are likely to be in the near future.

III SEMANTIC QUERY OPTIMIZATION

A request for information can often be formulated in more than one way, depending on knowledge about the subject area and ingenuity in determining the best access path to the desired information. A question about all ships currently carrying iron ore, for example, can be answered by only looking at information about bulk ore carriers, assuming that it is known that only bulk ore carriers carry iron ore.

Semantic query optimization is an approach tc query optimization that uses such domain knowledge, often referred to as semantic integrity constraints. The objective is to transform a query into a semantically equivalent one, one that produces the same answer, but that can be processed more efficiently.

A key issue in designing a query optimization system that uses semantic query optimization is the control of the inference of constraints based on semantic rules. There may well be many ways to transform a query semantically, but it is essential that only the most promising possibilities be tried. Otherwise, the inference process itself may cause the entire optimization and retrieval activity to be inefficient. Knowledge to guide the inference process includes the description of the physical structure of the database and the factors

that contribute to the cost of processing queries, as gleaned from the study of conventional optimization techniques. The essential idea is that some changes in the set of query constraints bring about changes in the set of processing operations that can or must be performed. It is crucial to recognize the connection between changes in the two sets, particularly when there is a significant effect on processing costs, without repeatedly resorting to detailed and expensive cost estimation procedures.

Thus, semantic query optimization can be viewed as a search at two levels of abstraction. At one level, there is a search in a space of semantically equivalent nonprocedural queries. The search is guided by a set of heuristics that represents a distillation or abstraction of the query processing expertise and detailed cost models developed in conventional optimization research. The result of this search is a group of candidate semantically equivalent queries. For each candidate, there is a search in a (lower level) space of possible processing methods and sequences using the detailed cost models; in other words, the conventional query optimization process.

In sum, semantic knowledge about the database can be used in a new way to achieve efficient data retrieval. The method supplements conventional query optimization methods. It draws on semantic, structural, and processing knowledge, and makes use of the heuristics of query processing developed in prior research. A system has been implemented to explore this approach. Improvements have been

demonstrated in a range of query types, by means of transformations that add or delete constraints on attributes or even entire files, as warranted by the database and query structure. In other cases, the system detects unsatisfiable query conditions without inspecting the database, or aborts inference when it determines that no improvement can be expected. Analysis overhead is low in all cases.

IV A METHODOLOGY FOR THE DESIGN OF INTEGRATED USER MODELS

This area concerns research to improve the techniques of database design and modelling. The Structural Model, formally defined within the KBMS project, captures those semantics that are of importance in the design of the physical structure of the database. In addition to relations and tuples, inter-relation ownership, reference, and subset connections are modelled. In this sense the structural model can be considered as a formalization of an important aspect of the semantic models, for instance the Entity-Relationship models, that are used to capture the requirements of the users.

In order to generate a comprehensive database model many data models of individual applications may have to be integrated, since the problem of designing a large database involves many potential users, each having his own perception of what the database should look like. The model supports the integration of many such views into a database model to support all users. Included in the model is a formalism for the clear expression of structural integrity constraints in each user's view of the application area and rules for resolving these constraints relative to all users.

V THE DESIGN OF HIGH PERFORMANCE DATABASES FROM INTEGRATED USER MODELS

The integrated database also has to satisfy the performance constraints of the individual users and make efficient aggregate use of the computer system resources. This latter area is now being addressed. We recognize that the mapping from model relations to effective physical files is much more complex than generally presented in simple analyses. Tuples from several conceptually distinct, but related relations may be placed into the same file space, and, especially in distributed systems, the converse may need to occur. A methodology to guide this mapping, while preserving semantic integrity, is the goal of this research.

For the initial specification of an operational database we envisage the collection of usage estimates applied to database submodels, the transformations of these estimates onto the integrated database, and the application of this information to make successive physical design choices. At the highest level we evaluate whether the physical binding of relational, hierarchical, or network database management systems is appropriate. At the next level we will make decisions in regard to clustering and linkage implementation, corresponding to the connections which have been defined in the structural model.

VI ACCESS PATH ORGANIZATION

The KBMS access path organization module will examine the current physical representation of the database, to determine how to process each (ad hoc) user request in a way that is likely to be efficient.

A responsive database should be self-organizing; that is, the decision as to which auxiliary files and access paths should exist, and how they should be represented, should be made by a database management system, with the user interface remaining invariant. The structural model provides a basis for a consistent interface, while permitting many alternative physical implementations. Use of dynamically changing statistical knowledge of the activity applied to the file structures can then improve the systemwide performance. Hence certain of the database administrator's traditional functions in database tuning (that is, establishing bindings between relations and files based on estimates of access frequencies) may be partially assumed by the system.

Reuse of previous responses and intermediate results can reduce access time. We have written a program that recognizes when a stored temporary is a subexpression of a new query. This will be used for improving processing of queries submitted in an ad hoc interactive environment. We are also studying how transactions can be optimized using this tool.

VII NATURAL LANGUAGE DATABASE UPDATE

Although considerable research has studied the problem of processing queries expressed in natural language, the possibilities for performing natural language database updates have not been explored. The primary difficulty is that the casual user of a natural language system does not understand the details of the underlying database, and so may make requests that either (a) are reasonable given their view of the domain but are not possible in the underlying database; (b) are ambiguous with respect to the underlying database; (c) have unanticipated side effects on the responses to earlier questions or on alternative views of the database. A theory detecting the particular user's view of the database and determining the legality, ambiguity, and potential side effects of updates expressed in natural language is being developed. A formal analysis of the problems associated with updates expressed on views (data sub-models) is central to this work. The expected result is a system that will process natural language updates, explaining problems or options to the user in terms that s/he can understand, and affecting the changes to the underlying database with the minimal disruption of other views.

VIII DESCRIPTIVE RESPONSES TO DATABASE QUERIES

The typical response to a database query is to list the set of items from the database that satify the conditions presented in the query. This list can be excessively long, and consequently may be inappropriate for a conversational system. Often, a more appropriate response to such queries is a description of the set, rather than a listing of its elements. For example, the response "All corporate officers" may be more concise and informative in response to "Which employees profit share?" than a list of 1,000 names. Practical techniques for producing a significant class of such responses from existing database systems without a using a separate world model or knowledge base have been been implemented.

IX NATURAL LANGUAGE

A natural language interface to the KBMS project database has been implemented using software developed at SRI International. The interface consists of a LIFER grammar, and database access comomponents borrowed from the LADDER natural language system. This facility is supporting research in other areas of the KBMS project.

X LEXICON MANAGEMENT

For natural language systems to provide practical access for database users, they must be capable of handling realistic databases. Such databases are often quite large, and may be subject to frequent updates. Both of these characteristics render impractical the encoding and maintenance of a fixed, in-core lexicon. We have developed and implemented a technique for reducing the number of lexical ambiguities for unknown terms by deferring lexical decisions as long as possible, and using a simple cost model to select an appropriate method for resolving remaining ambiguities.

XI FILE ACCESS SYSTEM

A file access system that uses symbolic keys to access variable length records based in PASCAL and supporting several host languages, including PASCAL, INTERLISP, and FORTRAN has been developed and is now being tested. The services that this system, named FLASH, expects from the underlying operating system are limited to directory management for named segments of secondary storage, and access to fixed size blocks or pages of these segments. In a multi-user environment some locking facilities are also needed. Since this subproject may become the basis for long-range database system development, reliabilty and efficiency have been major design and implementation objectives. It is specifically designed to provide strong and symmetric support facilities for databases, so that powerful database systems can become easier to implement than they are using conventional files, designed with only programmer's needs in mind. The underlying structure uses B+ trees for storage of both primary and secondary keys. This system will be used to study various dynamic storage and retrieval strategies. The experience of implementing FLASH is already being used to better define the Input-Output package for the ADA language.

XII MAINTENANCE OF A LARGE CODASYL BASED DATABASE OF SHIPS AND PORTS

A substantial database of ship, cargoes, ports, etc, is being maintained as a support activity to allow testing on problems of a realistic scale. This database was designed using the Structural Model, and is implemented using DBMS-20 on the SRI International DEC KL-10. A conventional database system provides a realistic basis for comparision for advanced techniques.

Data included are merchant ships, ports, cargoes, shipping lanes, and voyages.

XIII EXPERIMENTS WITH DATABASE TECHNOLOGY TO SUPPORT VLSI DESIGN

The complexity of large VLSI or device designs impacts the design cycle. Single designer approches take long or utilize chip or card area poorly, and multi-designer teams require much communication and task scheduling. Current design automation aids use task specific files without automatic feedback of analysis results or decisions made by the designers. We consider databases a potential communication tool where multiple designers work on different aspects of the same device or system. Logic and circuit design information of two devices, a ALU and a DEC-11 CPU, and the component library required to build them, have been placed into a CODASYL (DBMS-20) database. The lower level data elements can be automatically generated using an existing design automation program. Initial performance measurements indicate only a factor 2 performance degradation versus use of specialized design files. Modification of lower level elements during the design process is signalled automatically, using a height-first algorithm, to the related parent levels, so that this detailed knowledge can be incorporated in the higher level abstraction, when these are accessed during succesive design iterations. Current work in progress is developing access techniques to design data that are partially stored, and if not stored, generated from higher level circuit or logic specifications. The volume

of lower level elements in a VLSI design can be greater than is managable by the largest storage devices now available, so that automatic methods for VLSI design automation will need access to a mix of generatable, regular elements and instantiated, irregular elements if they are to handle large circuits that are not totally regular.

XIV DATABASE MACHINES

One focus of research activity within the KBMS project has involved the design and analysis of alternative hardware machine architectures oriented toward the very rapid execution of the relational operations which arise frequently in large-scale database management applications. During the past year, this research has yielded certain surprising and potentially important results which may ultimately prove useful in designing cost-effective, extremely high-performance database machines.

These results are manifested in a "high-level" design for a specialized non-von Neumann machine, portions of which would be implemented in VLSI, which would support the highly efficient evaluation of the most computationally complex operators of a relational algebra, specifically projection with elimination of duplicate tuples and the equi-join. Based on a hierarchy of associative storage devices, this architecture permits an O(log n) improvement in time complexity over the best known evaluation methods for these operators on a conventional computer system, without the use of redundant storage, and using currently available and potentially competitive technology. In many cases of practical import, the proposed architecture should also permit a very significant improvement (by a factor roughly proportional to the size of the primary associative storage device) over the performance of

previously implemented or proposed database machine architectures based on associative secondary storage devices.

XV DISTRIBUTED DATABASES

We expect that future databases will often be distributed. In widely distributed systems we cannot expect that each node will have a complete directory, and we believe that the contract-net paradigm, developed for control of distributed computing in artificial inteligence networks can provide a method for inter-node knowledge passing.

The management of redundant, distributed data can seriously affect system performance. We have analyzed and developed algorithms for integrity maintenance. The use of 'hole-lists', to inform nodes of update status of transaction while passing a minimal amount of information, has been shown to be effective. The analyses has also shown that it is difficult to better the performance of centralized control methods, if sufficient backup can be provided in the responsible nodes.

The separation of read-transactions into three classes, namely those that require no or minimal consistency, auditable consistency, and time-critical consistency, can improve aggregate system performance. The problems arising from serving transactions that support planning functions, which require access to large granules of the database, can be greatly reduced by lowering their consistency demands.

XVI CONCLUSION

At the current time, the project is about two years old, and work is continuing in most of the areas listed above, both at the theoretical and implementational levels. Some highlights of the implementation work to date are:

A query processing program that selects appropriate fields and orders response tuples in accordance with a simple model of the users needs, so that information most likely to be of interest is presented first.

A program (QUIST) that uses simple rules about the semantics of the domain of an underlying database and a cost model of query processing to perform semantic query optimization.

A program (SDDL) that facilitates the design of databases by interactively soliciting and crosschecking records, fields, and connections.

A program (DESCRIBE) that provides descriptive responses to database queries. The responses are formulated by inspecting likely fields for relevant generalizations.

FLASH, a "portable" file access system that uses symbolic keys to access variable length records based in PASCAL and supporting several host languages.

A lexical processing technique that allows natural language front ends to remain current on databases subject to updates.

In addition to continuing many of the existing efforts, several new projects are envisioned for the near future:

A natural language system that will perform updates to the underlying database.

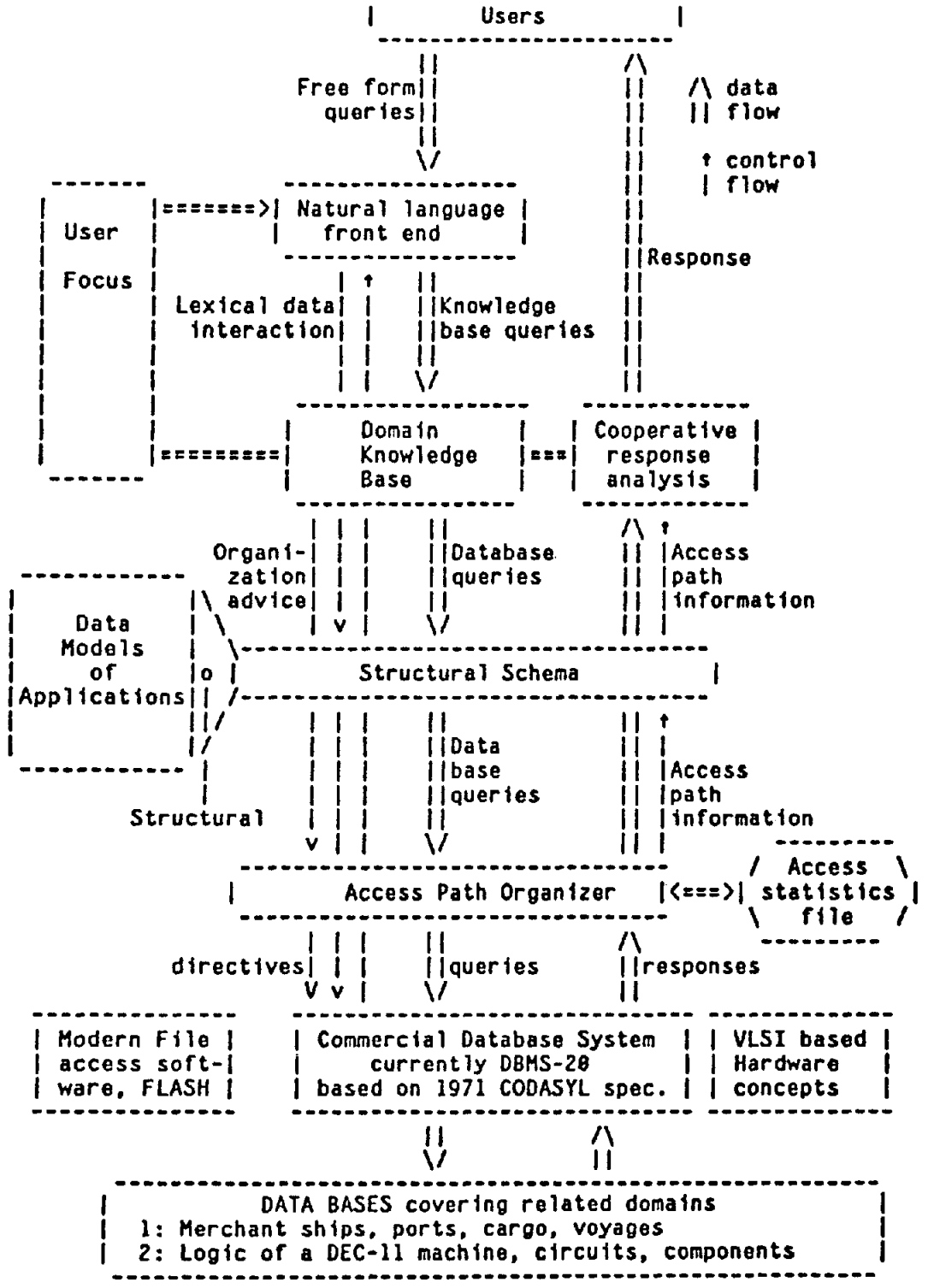
A program for the automatic selection of physical storage structures given a Structural Model definition of a database, and a model of expected access paths and frequencies.

The KBMS project is providing us with the opportunity to research various ideas in the combined use of artificial intelligence and more traditional DBMS techniques for facilitating the interactive use of databases. This short paper has presented the main directions of the project. More detailed information may be found in the various articles and reports mentioned in the bibliography.

XVII ACKNOWLEDGEMENTS

The work reported in this paper is the result of efforts by many individuals. The project is being directed by Stanford with cooperation from SRI International. Current Stanford participants are Mike Anderson, Jim Davidson, Sheldon Finkelstein, Jerry Kaplan, Arthur Keller, Jonathan King, Neil Rowe, Garrett Short, Kyu-young Whang and Professor Gio Wiederhold, Principal Investigator of the project. SRI participants are Barbara Grosz, Norm Haas, and Daniel Sagalowicz, Associate Investigator. Past contributors include James Allchin, Ramez El-Masri, Hector Garcia-Molina, Thomas Rogers, Earl Sacerdoti, and David Shaw. In some instances, work was performed using computing resources originally funded by NSF and NIH, and in cooperation with the Stanford portion of the S-1 project.

Figure 1: Conceptual Interactions of KBMS Components



REFERENCES

Allchin, J., A. Keller, and G. Wiederhold: "FLASH: A Language-Independent Portable File Access System"; Proceedings of ACM SIGMOD Conference, May 1988, pp. 151-156.

Barr, A. and J. Davidson: "Representation of Knowledge"; Stanford University CS Report CS-88-793, March 1988.

Davidson, J. and S.J. Kaplan: "Parsing in the Absence of a Complete Lexicon"; Proceedings of the 18th Annual Meeting of the Association for Computational Linguistics, Philadelphia, June 19-22, 1988.

El-Masri, R. and G. Wiederhold: "Database Model Integration Using the Structural Model"; Proceedings of the the ACM SIGMOD Conference, Boston, MA., June 1979, pp 191-198.

El-Masri, R. and G. Wiederhold: "Properties of Relationships and Their Representation"; Proceedings of the 1979 NCC, AFIPS Vol. 49, August 1979, pp. 319-326. El-Masri, R.: "On the Design, Use and Integration of Data Models"; Ph.D. dissertation, Stanford CS Report CS-88-801, June 1988.

Garcia-Molina, H. and G. Wiederhold: "Application of the Contract Net Protocol to Distributed Data Bases"; Stanford University Heuristic Programming Project paper HPP-77-21, April 1977.

Garcia-Molina, H.: "Distributed Database Coupling"; Stanford University Heuristic Programming Project paper HPP-78-4, March 1978, also appears in the Third USA-Japan Conference Proceedings, AFIPS, San Francisco, October, 1978, pp. 75-79.

Garcia-Molina, H.: "Performance Comparison of Update Algorithms for Distributed Databases, Part I"; Technical Note 143, Stanford University, Computer Systems Laboratory, Departments of Electrical Engineering and Computer Science, June 1978.

Garcia-Molina, H.: "Crash Recovery in the Centralized Locking Algorithm"; Technical Note 153, Stanford University, Digital Systems Laboratory, Departments of Electrical Engineering and Computer Science, November 1978.

Garcia-Molina, H.: "Performance Comparison of Two Update Algorithms for Distributed Databases"; Proceedings of the 3rd Berkeley Conference on Distributed Data Management and Computer Networks, August 1978, pp.108-119

Garcia-Molina, H.: "Performance Comparison of Update Algorithms for Distributed Databases, Part II"; Technical Note 146, Stanford University, Computer Systems Laboratory, December 1978.

Garcia-Molina, H: "Distributed Database Couplings"; Stanford University Heuristic Programming Project paper HPP-78-4, March 1978.

Garcia-Molina, H.: "Restricted Update Transactions and Read Only Transactions"; Technical Note 154, Stanford University, Computer Systems Laboratory, January 1979.

Garcia-Molina, H.: "Partitioned Data, Multiple Controllers, and Transactions with an Initially Unspecified Base Set"; Technical Note 155, Stanford University, Computer System Laboratory, February 1979.

Garcia-Molina, H.: "A Concurrency Control Mechanism for Distributed Databases Which Uses Centralized Locking Controllers"; Proceedings of the 4th Berkeley Conference on Distributed Data Management and Computer Networks, August 1979, pp.113-124.

Garcia-Molina, H.: "Centralized Control Update Algorithms for Distributed Databases"; Proceedings of the 1st International Conference on Distributed Processing Systems, October 1979.

Garcia-Molina, H.: "Performance of Update Algorithms for Replicated Data in a Distributed Database"; Ph.D. dissertation, Stanford University, Computer Science Department report CS-79-744, 1979.

Garcia-Molina, H. and Wiederhold, G.: "Read Only Transactions"; Stanford University Computer Science Department report CS-80-797, April 1980.

Ghosh, S., A.F. Cardenas, I. Mijares, and G. Wiederhold: "Some Very Large Data Bases in Developing Countries"; 5th International Conference on Very Large Databases, Rio de Janeiro, Brazil, October 1979, pp. 173-182.

Kaplan, S.J.: "Cooperative Responses from A Portable Natural Language Data Base Query System"; Ph.D. dissertation, University of Pennsylvania, July 1979, also Stanford University Heuristic Programming Project paper HPP-79-19.

Kaplan, S. J., E. Mays, and A. K. Joshi: "A Technique for Managing the Lexicon in a Natural Language Interface to a Changing Data Base"; proceedings of the 5th International Joint Conference on Artificial Intelligence, Tokyo, Japan, August 1979.

Kaplan, S. J.: "Appropriate Responses to Inappropriate Questions"; to appear in "Formal Aspects of Language and Discourse", A. K. Joshi, I.A. Sag. and B. L. Webber, Eds., Cambridge University Press, 1988.

King, J.: "Exploring the Use of Domain Knowledge for Query Processing Efficiency"; Stanford University Heuristic Programming Project paper HPP-79-30, December, 1979.

King, J.: "Modelling Concepts for Reasoning about Access to Knowledge"; proceedings of the ACM Workshop on Data Abstraction, Data Bases, and Conceptual Modelling, Pingree Park, Co., June 23-26, 1980.

King, J.: "Intelligent Retrieval Planning"; Proceedings of the first National Conference on Artificial Intelligence, Stanford, CA., August 1980, pp. 243-245.

Martin, N., P. Friedland, J. King, and M.J. Stefik: "Knowledge Base Management for Experiment Planning"; Stanford University, Heuristic Programming Project paper HPP-77-19 Report, August 1977.

Shaw, D.: "A Hierarchical Associative Architecture for the Parallel Evaluation of Relational Algebraic Database Primitives"; Stanford Computer Science Department Report CS-79-778, October 1979.

Shaw, D.: "A Relational Database Machine Architecture"; Proceedings of the 1980 Workshop on Computer Architecture for Non-Numeric

Processing, Asilomar, CA, March, 1988, also SIGMOD vol. X, no. 4, and SIGIR vol XV, no. 2, April 1988, pp.84-95.

Shaw, D.: "Knowledge-Based Retrieval on a Relational Databased Machine"; PhD Dissertation, Stanford University, Computer Science Department report CS-88-823, September, 1988.

Wiederhold,G.: "Introducing Semantic Information into a Database Schema"; Proceedings of the CIPS Session '78, Canadian Information Processing Society, September, 1978, pp. 338-391.

Wiederhold, G.: "Management of Semantic Information for Databases"; HPP-78-12, Proceedings of the 3rd USA-Japan Conference, Session 18-2-1, San Francisco, October 1978.

Wiederhold, G. and R. El-Masri: "Structured Model for Database Systems"; Stanford University, Computer Science Department report CS-79-722, April 1979.

Wiederhold, G. and R. El-Masri: "The Structural Model for Database Design"; Proceedings of the International Conference on Entity-Relationship Approach to Systems Analysis and Design, North Holland Press, December 1979, pp 247-267.

Wiederhold, G.: "Databases for Economic Planning in India"; Management Sciences and the Development of Asean Economies, S. Torok (editor), Times Books International, Singapore, 1979.

Wiederhold, G: "Database Technology in Healthcare", to be published in the Journal of Medical Science, 1988.

Wiederhold, G.: "Databases in Health Care"; to be published in a compendium series on Technology in Healthcare, sponsored by the Healthcare Technology Center, University of Missouri, Columbia, MO, also Stanford CS Report 88-798, March 1988.

Wiederhold, G., Beetem, A., Short, G. : "A Database Approach to Communication in VLSI Design"; TR 196, Stanford University, Computer Systems Laboratory, 1988.