EXTENDED HARDWARE SEQUENTIAL SEARCH SYSTEM

C.R.Watters
School of Computer Science
Technical University of Nova Scotia

ABSTRACT

A fast sequential search system is proposed that provides interactive query facilities for access to bibliographic databases. The system combines software with a special purpose hardware sequential search unit to provide both user interaction and search speed. The user follows online search strategies using data collected during the sequential search of the database. The system involves no storage overhead nor does it require extensive manipulation of the database.

The proposed retrieval model attempts to improve the dynamics of searching when inverted file access is either not possible or not practical. The combination of special purpose hardware for search speed and software for user interaction moves the system functionally closer to traditional online retrieval.

EXTENDED SEQUENTIAL SEARCH

INTRODUCTION

An Extended Sequential Search (ESS) system is proposed that provides the user with online retrieval capabilities in a sequential search environment. A special purpose hardware component for searching and software components for user interaction are combined for sequential retrieval from a semi-structured database, in particular a bibliographic database. A software prototype of this extended sequential model has been implemented to illustrate the interactive facilities available with the extended model.

Since traditional or von Neumann computer architectures are not well suited to non-numerical processing, much work has been done to design alternative, non-conventional, non-von Neumann architectures that are better suited to character processing (Hsaio, 1979) (Hollaar, 1979). A great deal of work has been done in recent years in developing non-conventional architectures for fast retrieval of large structured databases, particularly for DBMS databases. Examples of systems for DBMS using non-conventional hardware for sequential searches include: CASSM (Su, 1979), RAP (Schuster et al, 1979), ASSM (Hurson, 1981), STARAN (Rudolph, 1972), and the Data Base Computer (Banerjee et al, 1979). Semi-structured databases are those with well defined fields but loosely defined field domains. Bibliographic databases are good examples with well defined fields such as title and authors, but little structure within each field in terms of repetitions, length, or format. Systems using non-conventional architectures for less highly structured and full text databases include: the Utah Text Retrieval Project (Hollaar, 1983, 1984), the High Speed Text Search System (Mayper et al, 1978), and the DAP Retrieval Project (Pogue and Willett, 1984). Grundke and Heaps (1983) proposed using an intelligent peripheral device to rapidly filter a sequential database to reduce to amount of data processed by software.

EXTENDED SEQUENTIAL SEARCH MODEL

The conventional sequential retrieval model accepts as input a preset query with fixed logical connections between the query terms. The online retrieval model accepts a user query that may range in format from Boolean to vector, processes the query, returns a set of hits, and allows the user to continue this interactive iteration until the user is satisfied with the results. Most currently available commercial information retrieval systems for bibliographic and full text data, including DIALOG, ORBIT, MEDLARS, New

York Times Information Bank, and Mead Data LEXIS system, have been implemented using inverted files (Salton and McGill, 1983).

The effect of a precise interpretation of Boolean logic is likely to be more pronounced with a sequential search than with an online search. In an online search session new terms can be added and logical connections revised with little effort. In a sequential search session the cost of revising a precisely stated Boolean query is to repeat the search over the entire database.

ESS is a model for sequential retrieval that extends the conventional sequential model closer to current models for online retrieval.

The ESS sequential retrieval model differs from the conventional sequential model by shifting the query formulation from a preset query determined before the sequential search to an interactive query formulation after the completion of the sequential search. This permits the user to formulate precise queries after the sequential search and allows the user to take advantage of data collected during the sequential search of the database.

SEQUENTIAL SEARCH

Fast sequential searching becomes an attractive alternative to indexing a database in some circumstances. Although indexing a database reduces the time required for online searching of the data base, several factors may make indexing of the database impractical:

i) As database sizes increase, the storage requirements of indices may require as much storage as the original data base.

ii) A high frequency of updates to a database may make the cost of updating the indices prohibitive.

iii) The use made of the database for query retrieval may simply be too low to warrant the expense of the index.

iv) The facility and means for collecting data may make the inclusion of an index impractical, especially if the data is collected in real time such as in satellite communications.

v) The ready availability of unindexed databases as a by-product of other operations such as typesetting of journals and newspapers, preparation of pictorial data, and collection of numeric data makes a fast sequential search capability more attractive.

In the sequential search environment, it is not practical to execute a new search for each different iteration that the user requires to produce satisfactory results. The ESS system has been designed to combine the speed of hardware sequential text searching with the flexibility of software interfaces to improve the effectiveness of searching in a sequential environment.

ESS SYSTEM OVERVIEW

The ESS system provides an interactive environment for the user similar to traditional online interactive query processing whereby the user formulates and reformulates the query until the results are satisfactory. The ESS system allows the user to take advantage of the semi-structured form of most bibliographic databases by linking search terms with particular fields of the database record, such as author or title.

The ESS system has the following three components: the pre-search component to define the terms of the pre-search query that is used in the sequential search, the sequential search to select the potential hit records on the basis of the pre-search query, and the post-search component to interact with the user to determine a list of satisfactory records.

From the user°s perspective, the ESS system is a two stage retrieval system: an input stage before the sequential search and an online interaction after the sequential search.

The user inputs a pre-search query before the sequential search is performed. This pre-search query is intended to provide the basis for a broad sweep through the database. The query syntax is essentially °OR°s of term-attribute pairs with °HAS° (must contain) and °NOT° (must not contain) logic for occasional use.

The user proceeds with the second stage of interaction after the sequential search has been completed.

The post-search component interacts with the user in an online iterative manner similar to traditional online retrieval to narrow the sequential search results to a set of database records that satisfy the user°s requirements. The post-search component first evaluates and ranks the pre-search query terms in order of potential effectiveness for the user. The user generates a Boolean query using only terms from the pre-search query. The query is processed against those records satisfying the pre-search query by using the results of the sequential search. The resulting

hits are ranked and the number of hits presented to the user. The user may choose to review some or all of the relevant records, which are then retrieved from the database. The user may continue this iterative process until satisfied with the results.

The post-search component takes advantage of the term frequency data that were collected during the sequential scan of the database for ranking of both terms and results.

The post-search component can be used to rank the results at each iteration according to the exactness with which the record satisfies the query logic and by using term frequency data. Those records that satisfy the query logic exactly are presented first. Within this group of records term frequency data are used to order the records. Those records that partially satisfy the query logic would be presented later in the ranking order. The result of this ranking order is similar to the ranking order produced by Salton's extended Boolean scheme (Salton, 1984)(Salton et al, 1983). This would allow the user to examine records which, although not perfect matches to the query, have some terms in common with the query.

HARDWARE SEQUENTIAL SEARCH COMPONENT

The hardware sequential search component must process the database sequentially against the pre-search query. Results of the sequential search are passed to the software post-search component for further interaction with the user. The speed of the search component, which is critical to the effectiveness of the sequential search system, must be comparable to the transfer rate of the disk.

In a sequential search environment the most costly component is the sequential search of the database. The incremental cost for collecting additional data about terms is minimal since each character of the database passes through the term match processor at the data transfer rate no matter how many terms are being compared and no matter what term data is collected. Consequently, the sequential search can be used to both screen the database for potential hit records and to collect data for each of these potential hits with regard to frequency of occurrence of each of the query terms. This frequency data can be used subsequently to assist the user to formulate queries using the more effective terms and to rank the results.

The search component must process the entire database searching for term-attribute matches as specified in the pre-search query. Those records satisfying the pre-search query are identified as potential hit records. Data such as term frequencies per document and term frequencies for the entire database can be collected during the sequential

search for each of the terms in the pre-search query.

A special purpose hardware sequential search unit must be able to accept characters from the data stream at the disk transfer rate, compare these characters with those in all of the query terms, collect any matches for each record, accummulate counts for matches, and resolve any query syntax for each record. A pipeline architecture of processors (Heaps, 1983), each doing one function per clock cycle, is suitable for such a task.

The hardware unit proposed for the ESS system is a pipeline that has multiple processors each operating on the data. The processors of the pipeline are as follows: coder, term comparator, attribute resolver, match accummulator and term counter, logic resolver, and record counter.

SEARCH SPEED AND DATABASE SIZE

The speed of the ESS system resides in the speed of the sequential search component. Given a hardware implementation of the sequential search component, the time required for the sequential search depends on the transfer rate of the data from the disk and the size of the database file. Search times of 4-6 minutes for a 300 megabyte disk have been shown for comparable hardware search devices (Hollaar, 1983). Using a sequential search for infrequent retrieval reduces the overhead of both storage and maintenance. Using conventional inverted file software a typical inverted file may require as much storage as half of the sequential database. In the case of a large system disk drive of 450 megabyte capacity, the ESS system would allow retrieval from 450,000 records of the 1 kilobyte size. If an inverted file was used for interactive retrieval only 300,000 records could be stored on a 450 megabyte disk.

CONCLUSION

The ESS system could permit interactive full field retrieval for any data file with minimal pre-processing or pre-formatting of the data file. Each field or attribute must be flagged for access by the use of attribute descriptors. For access without attribute specificity only the end of each record must be marked. Thus, the ESS system could be used as the retrieval system for many very different files on a user's system.

The proposed ESS retrieval model attempts to improve the dynamics of searching when inverted file access to a database is either not possible or not practical. The combination of special purpose hardware for search speed and software for user interaction moves the system functionally closer to the traditional online model.

REFERENCES

BANERJEE J., HSAIO D.K., KANNAN K. "DBC - A database computer for very large databases". IEEE Transactions on Computers. Vol. c-28, No.6, June 1979. 414-429.

GRUNDKE E, HEAPS H.S. "Rapid search of textual data bases by intelligent peripherals". Proceedings of the 11th Annual CAIS Conference. Halifax, 1983.

HEAPS H.S. "A pipeline Processor for bibliographic retrieval". Technical University of Nova Scotia. Unpublished. 1983

HOLLAAR L.A. "Unconventional computer architectures for information retrieval". Annual Review of Information Science and Technology. ed. M.Williams 1979.

--------. "Hardware systems for text information retreival." Proceedings of 6th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Vol 17, Number 4. 1983.

--------. "The Utah text retrieval project- a status report". Proceedings of the Third Joint BCS and ACM Symposium Research and Development in Information Retrieval. Cambridge 2-6 July 1984.

HSAIO D.K. "Data base computers". Advances in Computers. Vol 19. Academic Press. 1980.

HURSON A. "An Associative Backend Machine for data base management". IEEE Computer Society Workshop on Computer Architecture for Pattern Analysis and Image Database Management. Nov 1981.

MAYPER V.Jr, MICHELS L.S., NAGY A.L. "A practical text search system for unindexed data". Proceedings of COMPSAC 78 Computer Software and Applications Conference Chicago Nov. 1978. p 710-715.

POGUE C., WILLETT P. "An evaluation of document retrieval from serial files using the ICL Distributed Array Processor". Online Review. Dec. 1984.

SALTON G. "Extented Boolean information retreival - an outline." Proceedings of the fifth National Online Meeting. New York, April 10-12, 1984. pp 339-45.

--------, MCGILL M. Introduction to Modern Information Retrieval. McGraw-Hill Book Co. 1983.

--------, FOX E.A., WU H. "Extended Boolean information retrieval". Communications of the ACM. Vol 26, No 11. November 1983. pp 1022-1036.

SCHUSTER S.A., SMITH K.C., NGYEN H.B., OZKARAHAN. "RAP.2 - An associative processor for databases and its applications". IEEE Transactions on Computers. vol c-28. no 6. June 1979.

SU S.Y.W., NGUYEN L.H., EMAM A., LIPOVSKI G.J. "The architectural features and implementation techniques of the multicell CASSM". IEEE Transactions on Computers. vol c-28. no.6. June 1979. p 430-445.