THE BELL SYSTEM
TECHNICAL JOURNAL
DEVOTED TO THE SCIENTIFIC AND ENGINEERING
ASPECTS OF ELECTRICAL COMMUNICATION

Volume 52 December 1973 Number 10

Information Management System:

Interactive Information Management Systems

By D. T. CHAI and J. M. WIER

(Manuscript received October 5, 1972)

This paper and the following three describe computer systems to store, retrieve, and manipulate information. These have all utilized time-shared computer systems. All have evolved toward a system constructed of modular component parts and having a high degree of user interaction. Considerable attention has been given to implementation in a form suitable for simple transfer to systems of adequate capability with minimal programming effort. The data bases involved are all hierarchical in organization. The major parts are a language facility, a data base manager, a processing package, and numerous coordinated administration functions. The parts are currently assembled into a package which can be applied to an arbitrary hierarchically structured data base with little user effort. The component parts are also available for integration into more tailored systems for special applications.

I. INTRODUCTION

This paper and the three that follow it discuss various aspects of the problem of using computers to store, retrieve, and manipulate

information. In particular they describe computer systems for carrying out important parts of such work. These parts have been integrated into a system for handling information. The system described in these papers has been designed so that a user and the computer system can interact heavily in reaching the solution to a problem posed by the user.

Systems generically related to the ones described here have appeared in great numbers in the past decade.[1-6] In general they all use a computer to store, process, and provide results from information contained in a "data base" controlled by the computer. However, this deceptively simple description hides the many differences between the systems which make them less generally applicable than would seem immediately evident. No attempt will be made in the following to be complete in categorizing such systems. However, enough information will be given to place the present work in perspective with respect to important requirements placed on such systems in various applications.

To circumscribe the work reported here and its potential field of application, let us characterize information systems according to the properties indicated in Table I.

The systems which have been implemented using the tools reported here generally are most useful in applications corresponding to the earlier-given of the choices in the various categories. The amount of information contained in the data bases served is generally less than 50,000,000 characters. The information is heavily structured into a hierarchical format. The users are typically not highly skilled in the use of computers. A typical request placed on the system will require fewer than ten seconds of processing. Finally, the user will always expect an answer in less than ten minutes, often in less than one minute, and occasionally in less than ten seconds.

These figures are dictated by the uses to which the systems are usually put, tempered by economic and computer limitations. Relatively small packets of information are supplied to the system in any one transaction. Further, requests to provide information and processing are simple since the user employs on-line composition of requests and interpretation of the results delivered.

The properties implied by this method of interaction cause the resulting system to be somewhat specialized in order to carry out such operations to the satisfaction of the potential users. The following are a few cases where the decision to handle processing in the manner indicated may adversely affect the applicability of the system to other uses.

In order that response time to a given request be short, the system tailors its operations to deal with a spectrum of requests assumed known at the time of system origination. Thus, requests for large

TABLE I—CHARACTERIZATION OF INFORMATION SYSTEMS

Amount of Information:
Up to 100,000 characters
100,000 characters to 50,000,000 characters
50,000,000 characters and up
Structure of Information:
Hierarchical
Network
List
Users:
Non-computer skilled
Computer skilled
Size of Transaction:
Less than 10 seconds of processing
Greater than 10 seconds of processing
Time Scale:
Less than 1 minute
1 minute to 10 minutes
Greater than 10 minutes

amounts of output, complex or lengthy processing, or data stored in some order much different from that assumed may result in poor service. Specifically, mass business data processing is frequently not well handled in this way.

Since the system is designed to serve an interactive user as well as is feasible, the data base may be more difficult to update or set up in the first place than one specifically designed to be processed as a whole. In the same vein, restart procedures are generally more difficult to incorporate as such operations take time and thus cause poorer time response.

The decision to utilize a hierarchically structured data base means that other organizations will be unavailable, except as they can be mapped onto a hierarchy.

The concentration on serving users who are perhaps not skilled in the use of computers limits the complexity of potential operations.

The exact degree of difficulty for other applications caused by each of these choices varies. The positive benefits obtained have been adjudged sufficient rewards in the thriving areas where the system to be described is used.

II. SYSTEM DESCRIPTION

All of the elements of the total system to be described in these papers have been implemented on a time-shared computer system. The computer system thus takes care of many of the details involved in serving

many users. Some of the more obvious and important of these are:

(*i*) Provision of an interface to a communication facility.
(*ii*) Provision for separating users into categories and keeping them apart.
(*iii*) Provision of a flexible charging structure.
(*iv*) Provision for physical storage allocation.

The parts of the information management system are assembled in a modular fashion. Between each of them is a well-defined interface for exchanging information. The components are put together as shown in Fig. 1.

In this figure the users are shown impinging on the system at the left. This contact takes place via the switched telephone network. One or more users can be connected to the information system described at any time. Each user interfaces with the Natural Dialogue System (NDS). The Natural Dialogue System is described more fully by Puerling and Roberto.[7] It provides the ability to carry on a relatively simple interactive pseudo-English conversation with the user in order to ascertain his needs.

When an adequate amount of information is available to define the user service request, the Natural Dialogue System passes information sufficient to define the request to a processor. The processor chosen is determined by the user-NDS dialogue. The processor then uses its input data to make calls on Master Links (ML) to provide specified information from the associated data base or send some to it. Master Links, using facilities described by Gibson and Stockhausen,[8] carries out the operations required on the data base and returns the data needed. The chosen processor then formats the response and sends it to the user. The whole sequence may be reinstituted by the user by placing a new request before the system or the user may actively (by signing off) or passively (by hanging up) abandon his quest.

In this system the processors are one of two types:

(*i*) Job-specific ones that have been specially programmed for an application.
(*ii*) General-purpose ones that have been found to be useful in numerous applications and that thus are provided to all users.

In addition to these elements there exist a number of auxiliary capabilities which are necessary to the smooth and complete operation of such systems. These capabilities are provided by numerous programming packages. They, among other tasks, take care of loading

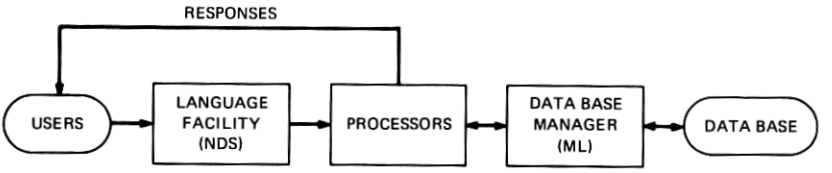

Fig. 1—Components of an information management system.

bulk data, checking its validity, auditing the system for efficiency and completeness, rearranging the data into a different order, and taking statistics on usage. These will not be described.

III. INTERACTIVE INFORMATION PROCESSING

As was mentioned in the introduction, the systems described in this series of papers concentrate on the provision of a highly interactive contact between the user and the information management system. The importance of this type of interaction was dictated by the applications which led to the design. This section discusses some of the considerations leading to the specific design decisions made.

In a very practical sense, an understanding of the system is not possible without examining the environment in which it works. The job of solving problems involving a data base contained in an interactive information system is jointly shared by the user and the system. Each does what "he" can do best.

The information system takes care of data storage, data processing, and information display in addition to a number of housekeeping chores. The user brings in the problem, formulates the solution in the form of a sequence of requests placed before the system, and guides the work of the information system as it progresses.

These operations would appear to be identical with those carried out in classically programmed data processing. The user (a programmer) translates a problem into a sequence of data processing steps which the computer is given to carry out. There is, however, an important difference which makes the interactive process much better for some applications.

That difference stems from the fact that it is not possible to program a computer to provide some solution unless an algorithm exists for doing it. When working with complex data bases, it is frequently necessary to find out a great deal about the data just to be able to write a suitable program. This process of "finding out about the data"

is frequently best done by going into the data base on some exploratory trips. It is here that interactive data base processing is very useful.

The user not only can collect the data, but he can also exclude vast areas where it is not worth taking computer time to look. This is possible because he gets a "feel" for the data, the limits of its range, its empty spots, its peculiarities. These allow him to reduce search times, to try simplified models that are "apparent" from looking, and to avoid wasting time and effort. All of these are simple for the user to employ while provided with immediate response from the data base. They are frequently difficult to program. Recognition of patterns is one of man's strong points. Generation of all possible patterns to be explored is not.

In order to provide this interactive capability it is necessary to smooth the communication between the user and the interactive computer system. This process is not a simple one. Basically, it involves a smooth translation from a form which is "natural" and unambiguous to the user to one which the computer can use on input. On output the process is reversed.

Numerous studies have been made of the use of English as a communication medium for talking with a computer.[3,5,6,9–11] Unconstrained English serves this purpose poorly, not only because of implementation difficulties, but because of the heavy use of context and alogical constructions. Even the REL[5] system which has progressed a long way toward natural language usage requires a rather disciplined approach to construction and meaning. Montgomery[11] has collected numerous telling examples which clearly illustrate the difficulties. These problems have led the designers of this system to adopt a pseudo-English language based on independent phrases, each of which begins with a specified keyword. The use of keywords greatly reduces the ambiguities of the user request and, at the same time, reduces the parsing or analyzing time by the computer. The paper by Puerling and Roberto[7] describes the keyword style of languages that is available through the use of the Natural Dialogue System. The paper by Heindel and Roberto[12] describes one implementation of a keyword language for general-purpose retrievals.

The choice of accepting independent phrases in a request also materially simplifies another computer-user interaction process. Economics and the state of technology strongly recommend a keyboard input mechanism (other choices cost too much or are not well developed technically). Unfortunately, typing, particularly facile

typing, is not a universally available skill. Thus input, for many potential users, is clumsy and is often a source of errors. The time delays and annoyances in this process often put off potential users and reduce the value of a system. The use of a phrase-type grammar provides some help in the system described by reducing retyping on errors to the level of the phrase rather than the sentence. In actual use the quantity of typing is further reduced by providing editing facilities which preserve common material already placed in the system from interaction to interaction.

A second communications barrier which can exist in an interactive system is that of response time. If the user is employing the system in an interactive way in the pursuit of a solution to his problem, he finds that excessive delays in delivering replies to his requests create gaps in the continuity of his thoughts on the solution. They distract him and, more seriously, they affect his ability to note patterns in the output. They thus reduce his effectiveness in solving the problem. They also bore him and waste his time, both of which reduce the probability that a proper and prompt solution will be forthcoming.

Because of the effect on user acceptance and user effectiveness, the systems to be described have been implemented with response time a major criterion of merit. This criterion has shaped the system in at least the following ways:

(*i*) The complexity of a request is reduced by making simple requests easier to formulate than complex ones.

(*ii*) The Master Links data base management system provides numerous tools for tailoring a data base to the requirements of its potential users.

(*iii*) The languages are designed to reduce search time in the data base by simplifying the specification of data base delimiters.

(*iv*) Monitors have been provided for noting the state of the data base and the usage by the system clientele.

(*v*) Dialogue is retained from request to request to reduce the typing burden.

(*vi*) Numerous detectors of errors are employed and extensive helpful (not critical) diagnostics are provided.

IV. SOME COMMENTS ON PERFORMANCE

As has been mentioned, the systems described have been designed to deliver prompt response in an interactive environment. In addition

to pursuing the goals just mentioned, the software has been designed to perform well in an absolute sense as well. To measure the performance actually achieved, extensive unit testing has been employed. In traffic situations, simulations have been run on the performance in the presence of various levels of load. Overall tests of system performance have been designed and run. A model for evaluating system performance as a function of the processing to be done in the data base has been developed. Such tests and models have been most helpful in comparing different system implementations and algorithms. The knowledge so gained has also been used in updating designs and optimizing system use.

The systems described have been used in various applications with data bases containing up to a few tens of millions of characters of data. These have all been hierarchical in organization and generally did not employ more than ten levels in the hierarchy. By using the various tuning facilities, the time to return answers to typical requests can often be reduced below ten seconds. More complex ones occasionally run to a few tens of seconds, but these employ less commonly used facilities. In general, requests requiring extensive data searches are more time-consuming than those requiring less information.

The key to good performance lies in matching the information management system to the needs of the application. In most applications the system can be tailored to provide adequately prompt service for the spectrum of common requests, sometimes at the expense of less important functions. These latter can usually be handled, less expeditiously, without creating an operational problem as they occur less frequently. In the current state of the art, no economic solution has been found which does not require this compromise for the larger and structurally more complex data bases. In all of the latter it is always possible to find pathological interactions with the data base which force data base searches in a very poor order.

V. SUMMARY

The work done in designing, testing, and applying the systems described has indicated the following:

(*i*) Interactive information management systems of acceptable performance are feasible and economically attractive in the current state of the art.

(*ii*) The hierarchical data base organization has been no handicap in providing information management in most applications tested.

(*iii*) It is desirable to match an information management system to the application in order to get prompt responses from it.

REFERENCES

1. Cuadra, C. A. (editor), *Annual Review of Information Science and Technology*, Interscience Publishers, 1966, 1967; Encyclopaedia Brittanica Co., 1968, 1969, 1971.
2. Senko, M. E., "Information Storage and Retrieval," in *Advances in Information Systems Science*, 2 (ed. by J. T. Tou), Plenum Press, 1969, pp. 229–281.
3. Salton, G., and Lesk, M. E., "The SMART Automatic Document Retrieval System—an Illustration," Comm. ACM, *8*, No. 6 (June 1965), pp. 391–398.
4. Sinowitz, N. R., "DATAPLUS—A Language for Real Time Information Retrieval from Hierarchical Data Bases," Proc. AFIPS, *32* (SJCC 1968), pp. 395–401.
5. Dostert, B. H., "REL—An Information System for a Dynamic Environment," REL Report 3, California Institute of Technology, December 1971.
6. Chai, D. T., "An Information Retrieval System Using Keyword Dialog," Information Storage and Retrieval, *9*, No. 7 (July 1973), pp. 373–387.
7. Puerling, B. W., and Roberto, J. T., "The Natural Dialogue System," B.S.T.J., this issue, pp. 1725–1741.
8. Gibson, T. A., and Stockhausen, P. F., "MASTER LINKS—A Hierarchical Data System," B.S.T.J., this issue, pp. 1691–1724.
9. Woods, W. A., "Procedural Semantics for Question Answering," Proc. AFIPS, *33* (FJCC 1968), pp. 457–471.
10. Kellogg, C. H., "A Natural Language Compiler for On-Line Data Management," Proc. AFIPS, *33* (FJCC 1968), pp. 473–492.
11. Montgomery, C. A., "Is Natural Language an Unnatural Query Language?" Proc. ACM, *25* (August 1972), pp. 1075–1078.
12. Heindel, L. E., and Roberto, J. T., "The Off-The-Shelf System—A Packaged Information Management System," B.S.T.J., this issue, pp. 1743–1763.