Reprinted from: Proceedings of the ASIS National Convention, 1971

THE ORGANIZATION OF RESEARCH DATA BANKS: EXPERIENCE WITH DIRAC-BASED INFORMATION SYSTEMS

J. Vallee*, J. Hynek**, G. Ray† and P. Wolf††

Abstract

This article presents the findings of a new kind of information retrieval effort at Stanford University, in the course of which scientific data banks were created, updated and interrogated remotely using a time-sharing computer. Experience with a direct-access network for astronomers is reviewed, also an automated processing system for bone marrow data and blood bank information, and an example of generalized programs in cancer research. Programmer intervention was minimized by reliance on a retrieval language named DIRAC-1. The study indicates that cost-effectiveness can be achieved in scientific information systems when language functions are sufficiently generalized to make file structures and access methods transparent to non-programmers. The impact the system has had on the work patterns of scientists using these data banks has been recorded. The article reports on the trends that were observed in terms of computer utilization and user behavior.

Introduction

During 1970 we have had the opportunity to create and use sizeable computer-based data files in time-sharing. These files contained actual research records and were applied to the analysis of problems of current scientific interest, notably in medicine and astronomy. An unusual aspect in the project was the fact that no programmer intervention took place, leading to costs considerably lower than could be expected under classical conditions. All files were made accessible under a uniform command system that was capable of interpreting statements typed at terminals rather than executing special programs in procedural fashion. This in turn made it possible to observe certain mechanisms in the interaction between users and their data that dedicated programming usually tends to obscure.

The principal difficulty in implementing scientific retrieval systems under classical languages such as PL/1, Fortran, and Cobol has arisen from the need to anticipate user behavior and "typical inquiries" at the time of system design. The current literature provides guidelines the data processing "expert" is invited to follow when he undertakes the implementation of a large data base and the search algorithm that will be used to process queries against that data base. Such guidelines have been successfully applied to airline reservation, banking, and other business systems. In the research environment, however, the enforcement of these rules leads to inefficiencies that tend to cumulate the problems already created by the rapid obsolescence of coding schemes and shifting concept structures.

Such a misunderstanding of the essential components of cost-effectiveness in research systems results in two observable defects: the first one is over-specialization (where we obtain very advanced systems that can serve only one field for a specific application); the second one is generalization at the wrong level, as shown in the spectacular failure of several large information systems, both in the business world and on campus. Sometimes, promising retrieval languages fail to completely satisfy researchers because they lock them inside a narrow set of commands, and prevent them from establishing communication between the records they have selected and statistical or mathematical routines that are available on the same computer.

* Stanford Computing Company, Inc., 770 Welch Road, Suite 327, Palo Alto, California 94304

** Director, Dearborn Observatory, Northwestern University, Evanston, Illinois 60201

† Division of Radiation Therapy, Department of Radiology, Stanford University Medical Center

†† Director, Clinical Laboratory, Stanford University Medical Center, Stanford, California 94305

Certain techniques of generalized programming, first introduced in non-scientific applications, seem to be useful here, and our work has been a preliminary exploration of this concept with real-life problems as test cases.

The DIRAC Concept

Generalized programming, a major trend in current software development, is not an altogether recent concept. It was recognized long ago by systems designers who applied it to input-output control routines, and the idea of extending this convenience to wider areas has been a natural one. When the Business community became aware of the hidden costs of dedicated programming and demanded of the software specialists increased capabilities for the on-line control of large data files, a number of information-oriented languages appeared on the market (1,2). All these languages attempt to relieve the user of the burden of creating and maintaining his files by special-purpose programs; they offer flexible interrogation tools that can be "interpreted" (rather than "compiled") for immediate execution.

Late in 1969 we began planning a series of information-oriented experiments using the Stanford time-sharing system, that runs on an IBM 360/67 computer, and a new retrieval language developed by one of us (3). The purpose of the experiment was to apply the non-procedural concept to scientific problems in order to assess its usefulness in a wide range of research areas. We hoped to be able to present some statement on the future feasibility of scientific information systems and also to arrive at some guidelines for a reduction of programming costs and delays in research involving the use of computers.

The file schema under which DIRAC-1 operates is that of a simple hierarchy of fields within an arbitrary number of records. Each record represents an entity in the real world. For instance, if we have defined a certain alphanumeric field in a catalogue of supernovae as the "cluster", and if we have defined another field, "Vs", as an integer representing the recession velocity of the star, it is now possible to type at the terminal the request:

Cluster Contains Virgo
and VS(<=2000 and >=1000).

The burden of generating an appropriate procedure to identify and retrieve those stars for which the information exists and matches the request (in this case, all supernovae in the Virgo cluster for which the velocity is between 1000 and 2000 km/s) is now placed on the system itself, and the user need not be concerned with the details of file operations that take place when this internally-generated procedure is executed.

After a three-month planning period during which the DIRAC prototype was mounted and tested on the Stanford computer, we began creating and updating files of medical and astronomical data.

```
DIRAC-1          STATUS REPORT FOR FILE A200    (CLASS 1)
                         17/APR/1970

      Dearborn Observatory Bright Star Catalogue

CREATION DATE     :  10/APR/1970   FILE NAME    :  Dearborn
RECORD NOTATION   :  $N$            CREATED BY   :  Northwestern
FIELD NOTATION    :  GF             REC. LENGTH  :  32
DISPOSITION       :  PUBLIC         NO. FIELDS   :  23
NO. OF RECORDS    :  8933          UPDATED LAST  :  15/APR/1970

  I D E N T I F I C A T I O N   &   S T A T I S T I C S
```

No	Name	Field Description	T&M	Nec?	Size	Existence	
1	Star	Star identification	A S	YES	4	8933	100.00%
2	Alpha	Right Ascension	R S	YES	4	8933	100.00%
3	Delta	Declination	R S	YES	4	8933	100.00%
4	Mag	Apparent Magnitude	R S	YES	4	8933	100.00%
5	Spelum	Spectrum/Luminosity	I M		3	8933	100.00%
6	Amotion	Proper motion alpha	I S		4	8773	98.21%
7	Dmotion	Proper motion delta	I S		4	8773	98.21%
8	Tpar	Trigon.parallax	I S		3	2159	24.17%
9	Spar	Spectr.parallax	I S		3	3673	41.12%
10	Dpar	Dynamic parallax	I S		3	515	5.77%
11	Apar	Adopted parallax	I S		3	5444	60.94%
12	Radvel	Radial velocity	R S		4	6759	75.66%
13	Rota	Rotation	I M		1	6856	76.75%
14	Mul	Multiplicity Code	C S	YES	3	8933	100.00%
15	Dm	Delta.m	I M		2	2066	23.15%
16	Sep	Separation	I M		3	2184	24.45%
17	Per	Period	I M		4	376	4.21%
18	Remark	Remarks	C M		1	245	2.74%
19	HR	HR Region	C S		1	5444	60.94%

Figure 1. Short form of a DIRAC-1 Status Report. It shows types and multiplicities of data for a file of nearly 9,000 stellar systems. Such reports are typed at a remote terminal or displayed on a scope.

Experiments with an Astronomical Information Network

Since the advent of time-sharing in the mid sixties, considerable interest has centered on the idea of a documentation service used simultaneously by many specialists. Systems of this type have been designed, and some have been implemented partially, to explore applications in literature searching. There exists, on the other hand, a class of users whose need for interactive data retrieval was largely ignored by these early systems: in the course of many research activities, professional scientists do not really need access to literature references as much as they need the capability to interact meaningfully with observational records, standard tables, and private files of instrumental data. Large institutions devote their attention to the preparation, publication, and maintenance of scientific records, but they themselves do not have at their disposal an adequate computing tool of any generality. Clearly this is an aspect of data management that lends itself to automation, yet computing facilities have so far been unable to respond creatively to these needs.

Such a situation is encountered in astronomy, where specialists are well aware both of the potential usefulness and of the serious shortcomings of the computing machinery as an information processing tool. The Astronomical Society of the Pacific, for instance, expressed concern over the problem and stated in 1969:

> The need for a means of recovering various data on individual stars, galaxies, clusters, or other objects, is becoming obvious, and techniques for compiling, storing and distributing such data have been developed in recent years.

Recommending that such techniques be surveyed, the Society appointed a committee, chaired by Dr. Helmut Abt, to estimate the storage and retrieval needs of astronomers. While this study was conducted, Stanford and Northwestern were engaged in an experiment involving the use of DIRAC from remote terminals at Dearborn Observatory (near Chicago) and in the Bay Area, to query an astronomical data-base. Users could also create new files and update existing files in that data-base without programmer intervention.

The main active part of the experiment lasted from April 2 to May 26, 1970. Three basic catalogues had been stored under DIRAC: they were the Warsaw Catalogue of Supernovae (that was punched cover to cover for the purpose of the study), an expanded version of the Yale Bright Star Catalogue, with a volume of approximately ten million bits, and the Catalogue of Bright Galaxies of G. deVaucouleurs. Figure 1 shows the status of the Bright Star file, printed by the automatic report generator included as a subset of DIRAC. Eight astronomers used the system and we were able to record the parameters of 121 time-sharing sessions. These led to estimates of a number of factors that had not been previously accessible, such as the expected distribution of terminal sessions during the training period and during normal use of the facility; the statistical distribution of holding times; the level of interaction and growth rates of the files.

In a typical application of the system, a user in Illinois interrogated the Galaxy Catalogue to extract a sublist of all irregular galaxies for which radial velocities were not available, and used it the same day to prepare an observing schedule for the observatory's 40-inch telescope. This particular project would normally have required writing, compiling and debugging a special program. In another use of DIRAC a staff astronomer at Northwestern, Mr. Wackerling, generated the statistical distributions shown in Figure 2, again without special-purpose programming.

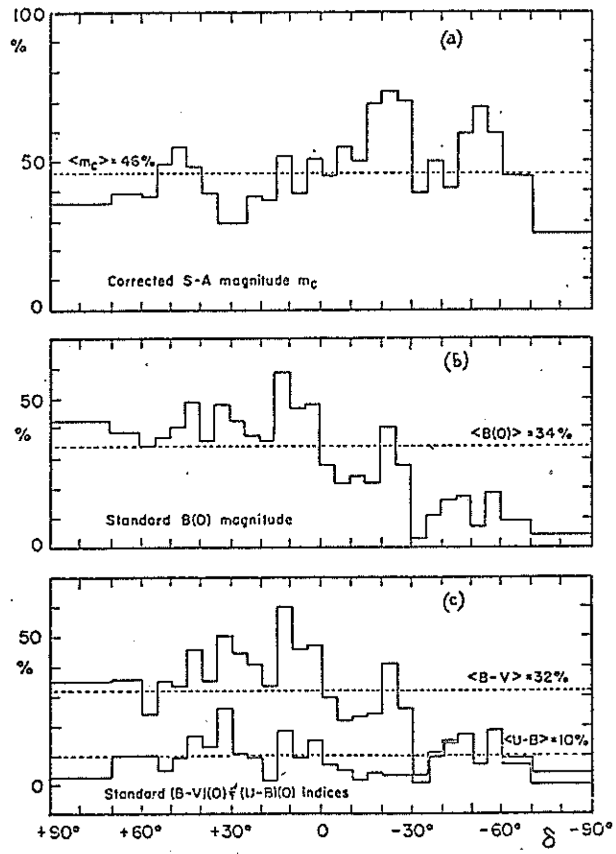

Figure 2. Relative frequencies of galaxies as a function of declination computed by DIRAC-1:
(a) galaxies of known corrected S-A magnitude
(b) galaxies of known standard B(O) magnitude
(c) galaxies of known standard indices.
Courtesy of G. DeVaucouleurs and L. Wackerling

We also evaluated the cost of extracting and printing a number of copies of a significant subset of the Bright Star Catalogue from a remote terminal. The problem was for the Illinois users to produce on a high-speed printer in California a catalogue of F-type stars that were primaries of a Southern visual pair. It took a single DIRAC command to generate this subset:

```
Delta < 0 AND mul(2)¬=0
AND Spelum(1) (>=300 AND <400)
```

This command reflects a coding scheme that we have described in detail elsewhere (4). The retrieval system found 192 such stars. We now evoke the text-editor to make this list of objects available for punching, editing, or off-line printing. We generated 25 copies of this special catalogue with a final unit cost of 50 cents. Total time spent at the terminal had been six minutes, and we had used 1.4 minutes of computer time.

Use of Interactive Retrieval in Cancer Research

The flexibility inherent in DIRAC and its facilitation of computer-user interaction in an on-line system is illustrated by its application to a problem in clinical cancer research. Using DIRAC for both data processing and analysis the Division of Radiation Therapy, Department of Radiology of the Stanford University Medical Center reviewed its fourteen years experience during which 307 consecutive cases of localized carcinoma of the prostate were treated with curative intent using the Stanford Medical Linear Accelerator as the external radiation source (5).

More than 300 individual items were abstracted from each patient chart and stored, necessitating large individual records with as many as 70 separate multiple fields. This massive data base demanded separate field interrogation, correlations and cross-correlations, as well as the ability to analyze certain subsets of data using statistical routines available to the user but outside of DIRAC.

The physicians were interested in such questions as the following:

-How many patients had disease limited by the prostatic capsule? -Of these, how many had solitary nodules occupying less than 25 percent of one lobe? -What was the survival of this group of patients five, ten, and fifteen years following completion of treatment? -What did statistical analysis of these groups' survival show when compared to other groups of patients? -Were there significant differences in sequellae between various rotational techniques?

Figure 3 shows survival curves for a selected subset of patients. Survival statistics such as these were displayed by a special routine that operated upon subsets of the file extracted by DIRAC. It was thus possible for the researcher to select potentially interesting subsets by successive approximation, to trigger the production of survival statistics on his terminal, and to return either to previously selected subsets or to the original file.

In a typical sequence of interrogations, an investigator had thus obtained survival statistics for patients with cancer extending beyond the prostatic capsule to immediately adjacent structures. The survival for this group was less than had been expected, and prompted the researcher to further interrogate the files of these patients looking for variables which might have contributed to the poor survival. This led to the identification of a separate group of patients who had extension into the bladder and whose survival was significantly ($p < 0.05$) poorer than the entire group so as to adversely change the slope of the survival curve. With the solution of this problem, the physician could then release the subset of data and return to interrogate the entire file of 307 patients.

Although this problem could not have been foreseen originally, therefore excluding its preprogramming as required by more conventional procedural languages, the non-procedural nature of DIRAC allowed further interrogation and solution of the problem without time consuming and costly re-programming.

During this study of the cancer file, we introduced new commands in DIRAC-1 to allow certain operations that had not been foreseen when the language was initially designed. Of special interest to physicians was the on-line display of correlation tables. If field 79 had the values A and D for a given subset of the file, and if field 80 had the values A,C, and D, then the command

Corr 79 and 80

produced the table:

FIELD NO. 80

				--	A	C	D
79	69.6%	16	--	0	2	4	10
	26.1%	6	A	6	0	0	0
	4.3%	1	D	1	.0	0	0
		23		7	2	4	10
				30.4	8.7	17.4	43.5

indicating for instance that ten records, out of 23 in the current subset, had the combination of a D in field 80 while field 79 was empty. Row and column totals, as well as percentages, were also generated.

```
  :    Q60 CONTAINS Y END
 ________________________
    14 RECORDS SELECTED

ACTION
  :    EXTRACT 1 DT6 END
 19620212  19681119  ALIVE  B     6.8 YEAR
 19581103  19640301   DEAD  D     5.3 YEAR
 19651018  19680318   DEAD  D     2.8 YEAR
 19650501  19680618   DEAD  D     3.0 YEAR
 19611013  19671127   DEAD  C     6.1 YEAR
 19640922  19690508  ALIVE  D     4.6 YEAR
 19571002  19690306  ALIVE  D    11.4 YEAR
 19680329  19700709  ALIVE  C     2.3 YEAR
 19661006  19691125   DEAD  D     3.1 YEAR
 19640730  19650222   DEAD  D     0.6 YEAR
 19651122  19670110   DEAD  D     1.1 YEAR
 19650930  19700526  ALIVE  D     4.7 YEAR
 19570730  19661209   DEAD  D     9.4 YEAR
 19620426  19680601                        6.1  LOST TO FOLLOW-UP

COMPUTATION OF SURVIVAL RATES
STATUS OF PATIENTS AT THE END OF EVERY 6-MONTH PERIOD SINCE FIRST TREATMENT

     YR    A    D  T.D   W  LST           % ALIVE               % DEAD          l'x    qx    px    Px             SURVIVAL CURVE

 1  0.0  14.   0.   0.   0.   0.  XXXXXXXXXXXXXXXXXXXX |                        14.00  0.0   1.00  1.00  -----------------------------*
 2  0.5  14.   1.   0.   0.   0.  XXXXXXXXXXXXXXXXXXXX |                        14.00  0.07  0.93  0.93  --------------------------*
 3  1.0  13.   1.   1.   0.   0.    XXXXXXXXXXXXXXXXXX | X                      13.00  0.08  0.92  0.86  ------------------------*
 4  1.5  12.   0.   2.   0.   0.     XXXXXXXXXXXXXXXXX |.XX                     12.00  0.0   1.00  0.86  ------------------------*
 5  2.0  12.   0.   2.   1.   0.     XXXXXXXXXXXXXXXXX | XX                     11.50  0.0   1.00  0.86  ------------------------*
 6  2.5  11.   1.   2.   0.   0.       XXXXXXXXXXXXXXX | XX                     11.00  0.09  0.91  0.78  ----------------------*
 7  3.0  10.   2.   3.   0.   0.        XXXXXXXXXXXXXX | XXXX                   10.00  0.20  0.80  0.62  -----------------*
 8  3.5   8.   0.   5.   0.   0.           XXXXXXXXXXX | XXXXXXX                 8.00  0.0   1.00  0.62  -----------------*
 9  4.0   8.   0.   5.   0.   0.           XXXXXXXXXXX | XXXXXXX                 8.00  0.0   1.00  0.62  -----------------*
10  4.5   8.   0.   5.   2.   0.           XXXXXXXXXXX | XXXXXXX                 7.00  0.0   1.00  0.62  -----------------*
11  5.0   6.   1.   5.   0.   0.              XXXXXXXX | XXXXXXX                 6.00  0.17  0.85  0.52  --------------*
12  5.5   5.   0.   6.   0.   0.               XXXXXXX | XXXXXXXX                5.00  0.0   1.00  0.52  --------------*
13  6.0   5.   1.   6.   0.   0.               XXXXXXX | XXXXXXXX                4.50  0.22  0.78  0.40  -----------*
14  6.5   3.   0.   7.   1.   1.                  XXXX | XXXXXXXXXX              2.50  0.0   1.00  0.40  -----------*
15  7.0   2.   0.   7.   0.   1.                    XX | XXXXXXXXXX              2.00  0.0   1.00  0.40  -----------*
16  7.5   2.   0.   7.   0.   1.                    XX | XXXXXXXXXX              2.00  0.0   1.00  0.40  -----------*
17  8.0   2.   0.   7.   0.   1.                    XX | XXXXXXXXXX              2.00  0.0   1.00  0.40  -----------*
18  8.5   2.   0.   7.   0.   1.                    XX | XXXXXXXXXX              2.00  0.0   1.00  0.40  -----------*
19  9.0   2.   1.   7.   0.   1.                    XX | XXXXXXXXXX              2.00  0.50  0.50  0.20  -----*
20  9.5   1.   0.   8.   0.   1.                     X | XXXXXXXXXXX             1.00  0.0   1.00  0.20  -----*
21 10.0   1.   0.   8.   0.   1.                     X | XXXXXXXXXXX             1.00  0.0   1.00  0.20  -----*
22 10.5   1.   0.   8.   0.   1.                     X | XXXXXXXXXXX             1.00  0.0   1.00  0.20  -----*
23 11.0   1.   0.   8.   1.   1.                     X | XXXXXXXXXXX             0.50  0.0   1.00  0.20  -----*
24 11.5   0.   0.   8.   0.   1.                       | XXXXXXXXXXX             0.0 ******************  -
25 12.0   0.   0.   8.   0.   1.                       | XXXXXXXXXXX             0.0 ******************  -
26 12.5   0.   0.   8.   0.   1.                       | XXXXXXXXXXX             0.0 ******************  -
27 13.0   0.   0.   8.   0.   1.                       | XXXXXXXXXXX             0.0 ******************  -
28 13.5   0.   0.   8.   0.   1.                       | XXXXXXXXXXX             0.0 ******************  -
29 14.0   0.   0.   8.   0.   1.                       | XXXXXXXXXXX             0.0 ******************  -

INSTANTANEOUS DEATH RATE =    0.06


ACTION
  :    RELEASE
 ________________________
    139 RECORDS SELECTED

ACTION
  :    Q43(1) CONTAINS Y AND Q43(5) CONTAINS Y END
 ________________________
    4 RECORDS SELECTED

ACTION
  :    Birthdate < 1920 AND Birthdate > 1910 END
 ________________________
    21 RECORDS SELECTED
```

Figure 3. Example of survival statistics for a subfile of cancer patients. Such special reports could be generated by users during a DIRAC run.

Data Management in the Clinical Laboratory

Early in the development of our project, the Clinical Laboratory at Stanford Hospital identified several areas where introduction of interactive data management appeared desirable.

Physicians concerned with the treatment of hematologic disease had long been interested in obtaining a simple and, if possible, inexpensive computer storage and retrieval method, in view of the rapidly expanding volume of bone marrow examinations they had to maintain. A primary objective of our project in this area was to produce a simple bone marrow report format. This file would then serve as an update source for DIRAC under control of the physician.

It is important to note that input

files to DIRAC were stored and managed under the unique text-editing system called WYLBUR developed at Stanford by J. Borgelt and R. Fajman. Under this text-editor, not only was it possible to interactively modify the contents of a data-set, but listings of interest to researchers could be generated either at the terminal or on high-speed printers.

Use of the system in this environment provided the physician with information at three levels: 1) the standard clinical diagnosis, 2) an improved bone marrow report, available for checking and editing, and 3) a retrieval and display language capable of answering unpredictable queries in conversational mode. This proved helpful in evaluating effects of new treatment methods on hematologic diseases as well as bone marrow and peripheral blood morphology, and in correlating bone marrow smears with bone marrow sections.

Another important area where DIRAC was used as a modeling tool was the blood bank. In view of the rapidly expanding volume of blood handled by a modern bank, and in view of the increased sophistication of the services it must provide, the design of computer-based systems has been recognized as imperative. Both rapid identification of donors and blood inventory control are important, but other administrative services can also be made more manageable by use of a computer.

The transfusion center prototype that was designed under DIRAC took advantage of the generality of the language to place the creation and maintenance of files under the control of the medical personnel. When the inventory is controlled by computer, the matching process takes the form of an interrogation of the parameters of units in storage. These parameters are displayed on a scope that the clerk can consult easily. Similarly, the computer can identify specifically matched donors and recipients with regard to platelets and cryoprecipitates. As blood units are used, the blood bank clerk triggers an update sequence that generates administrative reports, and also reflects the change in inventory that has taken place. When the inventory level falls below a certain threshold, a list of donors of the appropriate category is printed out. In emergencies, the system can be queried to produce the phone numbers of donors within a certain area (who can be called into the transfusion center within minutes) as shown on Figure 4.

```
:  Serum CONTAINS O END
-----------------------------------------------
512 RECORDS SELECTED
:  RETAIN
:  RHType CONTAINS NEG END
-----------------------------------------------
28 RECORDS SELECTED
:  City CONTAINS "Belmont" END
-----------------------------------------------
20 RECORDS SELECTED
:  DIFFERENCE (PresentDate - LastDate)
             > 60 END
-----------------------------------------------
18 RECORDS SELECTED
:  HomePH CONTAINS "593-" END
-----------------------------------------------
11 RECORDS SELECTED
:  TYPE Name Address City HomePH Bus.PH
   Serum RHType END
-----------------------------------------------
    23
Name           Best, John S.
Address        1721 Hiawatha Dr.
City           Belmont
HomePH         593-1789
Bus.PH         488-9161
Serum          O
RHType         NEG
-----------------------------------------------
    55
Name           Smith, Roger D.
Address        275 Fifth Ave.
  .              .
  .              .
  .              .
-----------------------------------------------
```

Figure 4

Some Technical Considerations

A large time-sharing system develops a well-defined pattern of behavior as a function of time, number of users and mix of jobs in its various partitions. It was important to understand the impact of a data-base system on this environment, to measure its essential parameters, and to carefully document failures. In the case of the applications that took place on Campus, the importance of failures was limited since users were already familiar with recovery procedures and were kept informed of current conditions at the computing site. In the case of the astronomy experiment, however, our users had no access to system levels below the text-editing commands, and failures created an especially critical situation. This also afforded an excellent test of the ability of scientists to solve the problems of data recovery, and led to suggestions of interest to systems engineers. In general, there were no great problems, except during two or three days when a major defect became apparent in the storage devices.

Careful accounting of the use of the system (text-editing, access time and computer time) was kept on a session-by-session basis, and these were cumulated on weekly intervals. The results are displayed in Figure 5 which provides some indication of the evolution of user behavior during the experiment. While the size of the data-base grew rapidly, actual use of the terminal was reduced, without as large a decrease in terms of computing time and editing time. We can also observe that users made progressively better use of the retrieval system, as the ratio of compute time to holding time evolved in a more economical direction.

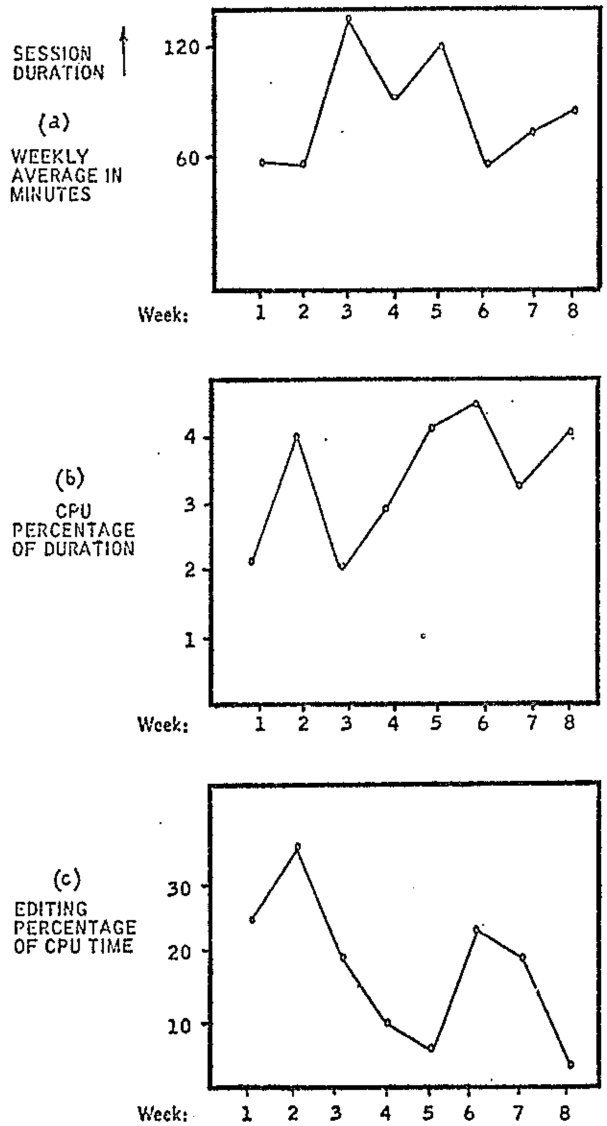

Figure 5

Conclusions

In an article published in 1966, (7), Lamson and Dimsdale defined an ideal system as one that would, among other requirements, "permit storage of documents and their retrieval by normal human processes in such a manner that the user could be unaware of the existence of the operative computer systems." Five years later, software designers are still far from this ideal goal. However the experiments summarized here clearly show that generalized programming can be as useful in the scientific environment as it promises to be in reducing the costs of industrial file handling. All our observations on the economics of the astronomical network have been gathered in a separate report (8), but our main conclusions - that also cover the other applications we have mentioned - can be summarized as follows:

1) The additional load such a language places on a time-sharing system is quite acceptable. This was verified in simulated conditions where large files were filled with data simultaneously from several remote terminals.

2) It was difficult to observe a "training" period or "learning curve" of any significance. Our users (who had never been exposed to time-sharing before) usually devoted their first session to learning basic commands, and began meaningful data retrieval on the next session. This was quite unexpected and caused the number of users to expand rapidly beyond our initial estimates.

3) The terminals proved to be valuable teaching aids when DIRAC could be used in connection with large files: Thus Northwestern astronomy students found it interesting to have to think about astronomical objects and the related literature in terms of information structures.

The cancer research experiment afforded an opportunity to observe information structures that were beyond the capability of most existing languages. These observations have led to the introduction of a versatile command language for the structures that will be available in DIRAC-2. Early tests of this new language with industrial data-bases indicate that significant improvements in cost-effectiveness have been achieved (12).

Our experience with DIRAC-1 indicates that a text-editor can be made to interface with a retrieval language to achieve considerable flexibility in the implementation of a potentially unlimited data-base. Networks using this concept can be created with the objective of reducing processing costs while bringing basic scientific information closer to specialists untrained in the art of programming. A major fraction of the operating cost then appears as pure computing activity (this figure was 42% in our case). Communications and connecting time costs are two items that are likely to decrease in coming years, making the concept of scientific data networks attractive to an increasing number of users.

References

1. McGee, W.C., "Generalized File Processing" in: Annual Review in Automatic Programming, vol. 5, Pergamon Press, 1968.

2. Olle, T.W., "Non-Procedural Languages - The Next Step in Automatic Programming" in: Annual Review in Automatic Programming, vol. 6, Pergamon Press, 1969.

3. Vallee, J.F., "DIRAC: An Interactive Retrieval Language with Computational Interface" in: Inform. Storage and Retrieval Journal, vol. 6, pp. 387-399, Dec. 1970.

4. Vallee, J.F. and Hynek, J.A., "An Automatic Question-Answering System for Stellar Astronomy" in: Publ. Astron. Soc. Pacific, no. 78, 463, Aug. 1966.

5. Ray, G.R., Cassady, J.R. and Bagshaw, M.A., "Prostatic Carcinoma Treated Definitively by External Megavoltage Irradiation: 14 Years Experience" in: Radiology, forthcoming.

6. Wolf, P. et al., "Progress Towards a Direct-Access Hematology Data-Base: Stanford's Experience with the DIRAC Language" in: Archives of Pathology, vol. 91, pp 542-549, June 1971.

7. Lamson, B.G. and Dimsdale, B., "A Natural-Language Information Retrieval System" in: Proceedings of the IEEE, vol. 54, no. 12, Dec. 1966.

8. Vallee, J.F., "Scientific Information Networks: A Case Study" in: Stanford Comp. Center, Inf. Systems Group, Report No. 2, Sept. 1970.

9. Survey of Generalized Data-Base Management Systems, CODASYL Systems Committee Report, 1969. (Distributed by the ACM)

10. "The Large Data-Base: Its Organization and User Interface", transcript of a panel discussion at the 1968 ACM Meeting, Las Vegas, Data-Base, no. 1, May 1969.

11. Hynek, J.A. and Wackerling, L.R., "Feasibility Study of an Astronomical Data Network", International Astronomical Union, Brighton (England) Triennal Meeting, Aug. 20, 1970.

12. "DIRAC-2: Concepts and Facilities" July 1971. available upon request.