Comput. & Indus. Engng Vol. 4, pp. 193–201
Pergamon Press Ltd., 1980. Printed in Great Britain

INFORMATION SYSTEM DESIGN FOR A PEDIATRIC CLINIC

S. SITHARAMA IYENGAR
Department of Computer Science, Louisiana State University, Baton Rouge, LA 70803, U.S.A.

(*Revised January 1980; received for publication 7 March 1980*)

Abstract—Although the "Information System" has pervaded nearly every aspect of modern industrial life, the pediatric clinic—replete with an overdose of information—lags behind in management and control. Although there are instances of the use of information technology in the medical system, these are usually limited to isolated processes, rather than to an integrated information system.

This paper presents a conceptual framework and the application of the framework when applied to the information system within a pediatric clinic. The design of the information system is discussed in terms of information processing, human factors and scheduling (patient) considerations. A prototype system for a pediatric clinic is presented.

INTRODUCTION

The most vital element in designing an organization is the information system design. Although the information system design concept has been applied to a wide variety of problem situations, these in general have been restricted to construction of conceptual frameworks. At this point, it is fair to state that although computer applications in the medical environment are increasing both in variety and extent, there has not been a general procedure to design an information system for a pediatric clinic. The purpose of this paper is to apply the concept of conceptual frameworks developed by Ackoff[1], Mason and Mitroff[4] and Lee[2] to a pediatric clinic information system. Padwal and Bandhopadhyay[5] have discussed an application of system design concept to a commercial bank. These results indicate that matching information with appropriate decisions, and cutting out redundant information can show very large cost savings. Rouse and Rouse[6] recently reported in their paper on "the design of a model-based on-line management information system for inter-library loan networks", that the data collection effort necessary for periodic updating in library operations can be quite expensive. An information system could eliminate redundant information and also would eliminate unnecessary cost involved in the operation.

BACKGROUND OF THE PROBLEM

In many areas of medical practice today, doctors are forming clinics, or joint practices, in an attempt to pool resources, energies and skills in order to provide better services to patients. Clinics such as these may have several doctors and several thousand active patients. Scheduling of patients can become a problem, especially in the case of a practice in which there are a large number of patients requesting service each day as, for example, is the case in a pediatric clinic.

Children typically "get sick" very quickly and unexpectedly, often, literally, overnight. Therefore, the volume of requests for appointments during a day can be large, indeed. When many parents attempt to secure an appointment at the same time, several persons can be kept very busy answering phones and scheduling visits. Since each person may receive phone requests for appointments with all of the doctors, and since it is practical to maintain only one appointment list, the mechanics of manually assigning appointments to patients can be complex, inefficient and time-consuming.

Imagine such a pediatric clinic with N doctors and M active patients. Suppose each doctor will schedule L ($4 \leq L < 6$) visits per hour for, say 9 hr per day (9 a.m.–6 p.m.). Assume further that appointments are regularly accepted for as many as K months into the future (including "Today"). Finally, suppose that the clinic wishes to maintain certain information (other than that recorded in the patient's medical folder) to be readily available to the staff when scheduling appointments, when checking patients into or out of the clinic, or when preparing periodic reports. Here is a situation in which a well-programmed computer would be of much assistance.

It is not impractical to consider such a clinic with, say, $N = 10$ doctors, $M = 20{,}000$ active patients, and in which doctors may choose to see from 4 to 6 (L) patients per hour for 8 or 9 hr per day. Patients may be scheduled up to $K = 4$ months ahead. The problem in general led the author to propose a scheduling program, a patient data base system and a physician's decision management system. This could be centralized into an on line/real-time management information system.

The approach suggested in this paper is to consider the design of the information system for a pediatric clinic on the following issues:

(1) Information processing considerations,

(2) Human factors considerations.

INFORMATION DEVELOPMENT PROCESS

The primary goal of an information system should be the improvement of the medical services available to the people of the community. This can best be accomplished by the integration of the decision making process by doctors and information technology. The system approach used here is a technique for the application of a scientific approach to complex problems. What is being stressed in our approach is an understanding of the information system environment, including interactions within the system. The description of the conceptual model will use Fig. 1 as an outline.

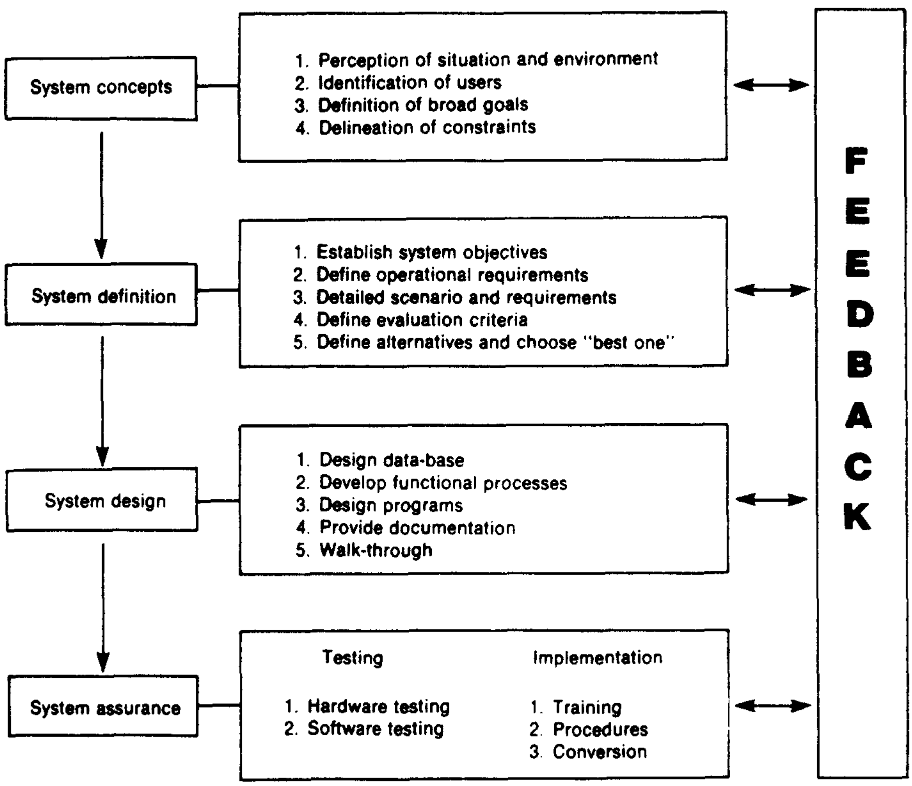

Fig. 1. Information system development process.

To discuss our proposed design, we will define the following.

System definition. A pediatric clinic information system is a combination of people, computer system with related programs, a reliable data base and institutional procedures interacting in a prescribed pattern.

System design. System designers must be responsible for the overall design of all System elements related to the System operation, whether they are implicitly or explicitly stated. Principally it involves the following:

(1) Design data-base

(2) Develop functional processes

(3) Design programs
(4) Provide documentation
(5) Structured programming.

Details on these factors are available in Iyengar and Alia's paper [3].

System assurance. The system assurance concept is a vital element in designing an information system. The purpose of system assurance is to verify the proper operation of the system and to verify the user's satisfaction with the new system. An integrated system test program is a significant undertaking and should be planned just as carefully as the system design. A methodical approach to system testing would include: test planning, a testing strategy, test data, function testing and acceptance testing. Further discussion can be obtained from Ref. [3].

CONCEPTUAL FRAME WORK DESIGN OF A PEDIATRIC CLINIC INFORMATION SYSTEM

Figure 2 shows the sketch of the information system for a pediatric clinic. Basically the sketch shows an overview of the entire scheduling system. Figure 3 shows the relationships between the various subsystems which make up the system. In addition to scheduling, information will be recorded concerning laboratory work performed, diagnosis, doctors visited and fees assessed. Details of the subsystem design, file structure and several components of the information system will be discussed in the design section of our paper.

The objective of our paper is to illustrate the design concepts for a medical clinic with respect to the following issues.

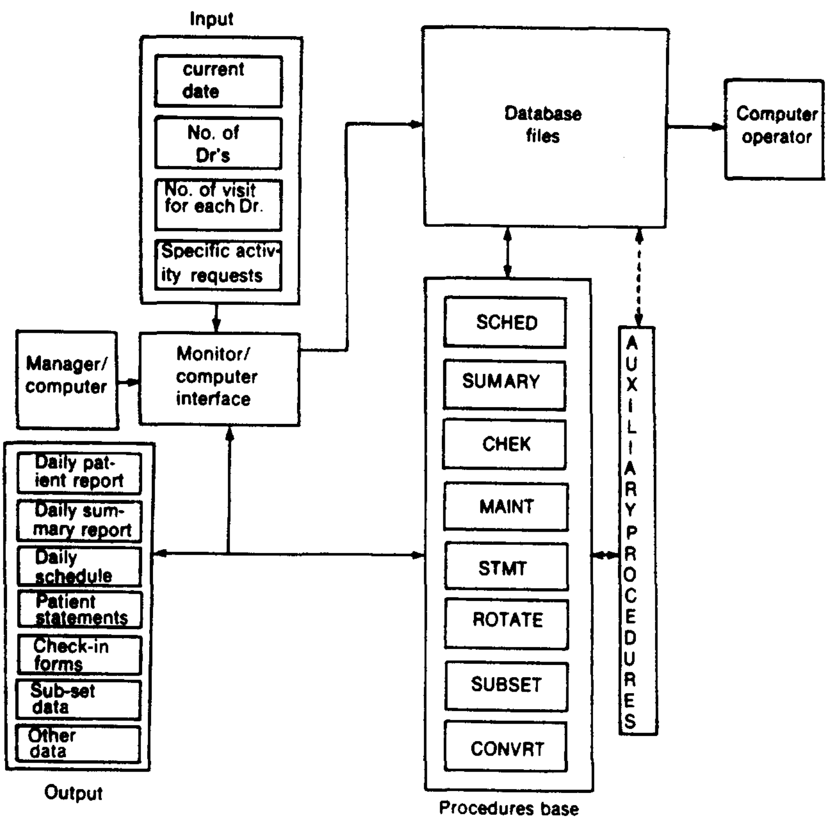

Fig. 2. Sketch of the information system for a pediatric clinic.

INFORMATION PROCESSING CONSIDERATIONS

Both this section and the next follow the lead of Rouse and Rouse [6]. As Rouse and Rouse point out, of the several design decisions to be made when developing a technique for information processing in a management information system (MIS), two stand out: "design of the data base" and design of "efficient execution procedures".

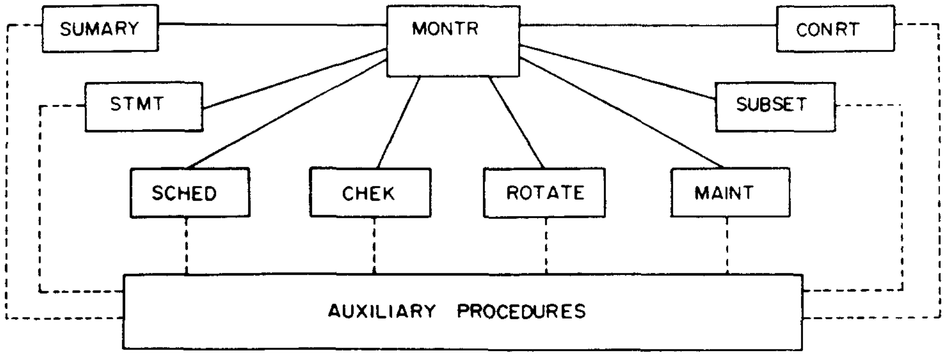

Fig. 3. Relationship of the sub-system.

The system monitor (MONITR) and the several procedures in the system serve as an interface to assist the user to communicate with the local operating system. The monitor allows the user to call up the desired procedures, and each procedure assists the user in the various manipulations of the data. Further, the procedures permit the user to transfer control back to the monitor for further transfer elsewhere, if that is desired. By designing the system so that it consists of separate, external subsystem procedures, sufficient memory space to accommodate all of the procedures at the same time is not required. Thus, all of the procedures are available to the user, but they need not all be resident in main memory at all times.

DATA BASE DESIGN CONCEPT

A data-base is a centralized collection of files stored for use by related subsystems, and consists of three prinary elements: physical storage structure, contents of data-base, logical relationships among data. Data base design involves developing efficient data storage and retrieval procedures. As Rouse and Rouse [6] states, that efficiency can be measured with respect to the following objectives of information processing:

(1) Minimize the number of disk accesses
(2) Minimize storage requirements
(3) Minimize response time or processing time
(4) Minimize vulnerability to system failure
(5) Accomodate access of data for multiple purposes.

Unfortunately, meeting all of these objectives is not always possible. For example, decreasing the number of disk accesses usually results in an increase in storage requirements.

Of the five items listed above, we have concentrated to date on implementing item (2). A description of files is included later in this paper, but we take this opportunity to present an example from the Patient Master File. This file contains such permanent information as patient name, responsible party information, primary physician, etc. In our original of the file, we included the name of a patient's primary physician (15 bytes) and the responsible party's full address (30 bytes). By utilizing a 1-digit code for each primary physician and a 1-digit code for each city of residence, the physician and address fields have been reduced to 1-byte and 24-bytes respectively. This represents a savings of 20 bytes per record. For example a data base of, say, 20,000 records (active patients), a saving of 400,000 bytes can be expected. This savings seems well worth the effort in record design. Of course, a directory of doctors' names and one of cities' names is now required, but they together consume only about 1100 bytes (on a full-scale system). So a net savings of 398,900 bytes of disk memory could be realized at the expense of some additional processing time (when names must be located via the appropriate directory).

HUMAN FACTORS CONSIDERATIONS

In an MIS where extensive amounts of data are to be entered, maximizing the entry rate while minimizing errors is of major concern. In the system under consideration here, the volume of data to be entered, once the data base has been initialized, will not usually exceed one line per transaction (listing an appointment or recording a visit). Therefore, speed vs error rate is not of major concern. Rather, the ease with which the system can be learned and the simplicity of making data entries to or data requests of the base are of significance here.

Each time an entry is requested of the operator, an appropriate prompt will be displayed. Once the system is learned, these prompts should be sufficient to provide smooth operation for the user. However, should the user desire it, assistance will be available at many data entry points of the program. The user need only type AID to receive additional assistance at these points. The additional assistance may consist of the list of legal responses which may be made and/or a reference to the user's manual provided.

Provision is also made to allow the user to return to the monitor at any point in the program where data may be entered. Further, when an activity (for example, an update to a master file) is requested of the system the user may continue to perform that activity until a new activity is requested. The information system discussed in this paper is being designed to accept "natural" data entries. That is, names, addresses and other related data items, for the most part, be typed into the system in the same way that they would be recorded with pen and paper.

It is hoped that this system will require only a few changes in the thinking and activities of the users.

A PROTOTYPE DESIGN

Due to space limitation, prototype design discussed for the scheduling program uses the following reduced values for the key parameters. (This program was originaly designed to run under MUSIC as implemented at Jackson State University.)

$$N = 3$$
$$M = 50$$
$$K = 0.07$$
$$4 \le L \le 6$$

where N refers to the number of doctors, M refers to the number of active patients, L refers to the number of visits per hour and K refers to the number of months.

The above numerical values are chosen to reduce the storage burden on our computer.

Music is a time-sharing operating system developed by McGill University System for the interactive computing group of McGill University. This has been implemented at the Jackson State University computer center with a IBM-370/145 computer system.

STRUCTURE OF THE FILE SYSTEM

The following files are included in the system:

(1) Patient Master File
(2) Excess Master File
(3) Daily patient log
(4) Daily schedules
(5) City directory
(6) Lab directory
(7) Diagnosis directory
(8) Physician directory.

Depending on facilities available at JSU, these files will be maintained on disk, or the program will be written to "think" of them as being on disk. A brief description of each file and its record structure follows.

Patient Master File

This is the main online source of information concerning active patients ("Active" patients are those who have visited the clinic within the last k months. The user defines the value of k.) The following records are maintained for each patient:

One personal information record containing patient's medical record number (MRNO), patient's last and first (and middle name or initial, if space permits) names, patient's date of birth (DOB), patient's sex, responsible party's last (R-LAST) and first (R-FIRST) names, address

(R-ADDR), area code and phone number of responsible party, patient's primary physician (code), and patient's current balance.

Two records containing information on visits and financial transactions are kept for each patient. These records provide information for the patient's statement.

Master Schedule File

This file consists of daily schedules for today and tomorrow. Each daily schedule consists of hourly schedules (9 a.m.–6 p.m., plus 2 "extra" hours), and a heading for each doctor. Each doctor may choose to schedule from 4 to 6 (NVPH) patient visits per hour. The first record for each doctor contains heading information and three hourly schedules. The remaining two records in a doctor's daily schedule contain the remaining hourly schelules. The extra (Use Code E) slots may be used for comments if desired. Each doctor's schedule is signalled by an H (Use Code) in the first byte of its first record. The date for that doctor's schedule is given in Julian form in bytes 3–5 of that first record. Schedules may be of two types:

R indicates regular daily visit schedules, and *N* indicates evening visits.

CODING SPECIFICATION WITH STORAGE ALLOCATION

This section of the paper will indicate the coding specification of the design.

City directory

A file of 7-character city names and 1-character codes. (10 cities, 80 bytes)

Lab directory

A file of 15-character lab test names and 1-character codes. (10 tests, 160 bytes)

Diagnosis directory

A file of 12-character diagnosis terms and 1-charcter codes. (10 terms, 130 bytes)

Doctor directory

A file of doctor's names and other information as follows:

1-character code
1-character NVPH code (number of visits/hour)
20-character doctor name

---

22 bytes/doctor.

Daily patient log

A temporary file (cleared daily) to store the following information on patients seen during a particular day:

5-character MRNO
1-digit code for doctor seen
1-digit verification code (See CHEK PROCEDURE)
1-digit diagnosis code
5-character fee entry

---

13 bytes per entry (Total 2184 bytes).

Excess Master File

A temporary file (cleared daily) to hold Patient Master File statement information entries for patients seen on a given day who do not then have space in their file entries, or for patients seen on that day who do not have a Patient Master File personal information record (new patients). Information from this file is entered into Patient Master File via the procedure MAINT. (432 bytes)

DESCRIPTION OF THE SUBSYSTEMS/PROCEDURES

Figure 3 shows the relationships between the various subsystem which make up the system. This section of the paper answers the question "What does this system do?" A brief description of each subsystem is provided.

MONITR

This procedure serves as the system monitor. It is the first point of arrival for a user entering the system. MONITR directs the flow of control to the various subsystems. In switching from one subsystem to another, a user must pass through MONITR. This procedure is also the one from which a user leaves the system.

Note. At any point at which the system requests input from a user, that user may request immediate return to MONITR.

SCHED

One of two major subsystems, SCHED performs tasks related to building and maintaining daily schedules (including today's schedule). The user may request activities to be performed on today's schedule or on that of some future date (MM/DD OR MM/DD/YY). Once requested, the daily schedule is called up and any of the following activities may be performed on that schedule:

(1) *ADDSCH.* Add a patient to a particular time slot in a given doctor's daily schedule.

(2) *CHGSCH.* Delete a patient from a particular time slot on a given doctor's daily schedule, or insert a new patient into such a given slot (by retyping the medical record number, MRNO).

(3) *VEWSCH.* View a chosen daily schedule entirely or

(4) *PRTSCH.* Print partially by doctor, by doctor and time, or by time (for all doctors)

(5) *DMPSCH.* Print the entire contents of the Master Schedule File or contents between and including specified dates.

(6) Capability may be given to view/print all open time slots for all doctors from some TIME 1 to a TIME 2 for a given day.

MAINT

The second of two major subsystems, MAINT provides facilities for maintaining all files (except schedules) by performing the following activities:

(1) *ADDMNT.* Add record(s) to a file

(2) *CHGMNT.* Change record(s) (by field) in a file, including deletion (clearing) of record(s)

(3) *VEWMNT.* View } any record(s) on file
(4) *PRTMNT.* Print }

(5) *DMPMNT.* Print the entire contents of file.

Concerning Patient Master File

MAINT handles the daily updates. For Patients whose Patient Master File transaction entries are all filled, transaction entries have been made in the Excess Master File. MAINT prints a hard copy of the transactions currently in the patient's record (for inclusion in the patient's medical folder), clears the patient's transaction records, and enters the new transaction record(s) from the Excess Master File. The just-transferred record is then cleared from the Excess Master File. For new patients, those without a personal information record in the patient Master File, MAINT permits their data to be inserted into the file. (Personal information must be collected manually at check-in or at check-out, and a MRNO must be assigned.) MAINT also permits the purging of records of patients (after printing of hard copy of those records) who have not visited the clinic in the last K months (k is defined at time of purge).

CHEK

This procedure performs the check-in/check-out functions of the system.

(1) *IN.* When a patient checks in, the patient's MRNO is added to a Daily Patient Log along with the 1-character doctor code of the doctor who is to see the patient. The additonal verification character I is added to the string to indicate that the patient checked in. On check-in, a form is printed for presentation to the doctor, it contains information useful in handling the medical and accounting records of the patient.

(2) *OT.* (out) When a patient checks out, the form received on check-in is presented to the receptionist or cashier who records on the terminal the MRNO, code for doctor seen, diagnosis and lab work codes, charges for services, and any payment received. The information concerning lab work, diagnosis, doctor seen and charges/payments is recorded as a transaction on the patient's Patient Master File records. The doctor code for the doctor seen is compared

with that on the Daily Patient Log and, if the doctor actually seen differs from the one listed on the Log, the doctor actually seen is listed on the Log. If the patient has checked in, the verification code is changed to B (checked both in and out). If the patient did not check in, and entry is made with a verification code O (out) in the Log. (Manual follow-up may be desired for patients with a verification I or O.) It is at check-out that calculations are made to bring the patient's statement up to date.

STMT

This procedure produces either a CRT display or a hard copy print-out of the patient's statement as it currently appears.

SUMARY

This procedure provides for the printout and CRT display of the Daily Summary Report including:

(1) Number of patients seen this day.

(2) Number of visits for each doctor for this day.

(3) Average number of visits per doctor for this day.

(4) Total charges for this day.

(5) Daily Visit Log (MRNO, name, doctor seen, verification code, diagnosis, charge) (taken from Daily Patient Log).

This procedure also contains facilities for adding to and deleting from the summary information as well as for changing information in the Daily Patient Log. Any further information desired for Daily Summary Reports may be requested and, if possible, will be provided.

SUBSET

This procedure provides for the printout and CRT display of subsets of the Master Patient File by any of the following attributes:

(1) Sex	(5) Last visit date
(2) Age	(6) Diagnosis
(3) Sex and age	(7) Sex and diagnosis
(4) Zip code	(8) Sex, age and diagnosis.

CONVRT

This is a utility procedure which permits converting strings of digital characters to integer or real numbers.

ROTATE

This procedure is used to clear today's schedule after it is printed out and no longer needed. It also prepares the space used for today's schedule for reuse as a schedule for the furthest future date maintained.

USE OF THE SYSTEM

In normal circumstances, daily operation of the system will proceed as follows:

a.m.

(1) Sign on and set current day's date

(2) "Rotate" schedules

(3) Print hard copy of current version of today's schedule

(4) Check that most recent Master Excess File data has been entered into Patient Master File and that the former file has been cleared. If not done, do so now.

(5) Check that most recent Summary Report has been produced and that summary information has been cleared. If not done, do so now.

During the day

Users sign on and access desired subsystems. SCHED and CHEK will be subsystems most often used. Users conduct daily business as necessary. Manager utilizes MAINT as necessary to maintain files. Manager enters data into Patient Master File for those patients seen between last shutdown and current day's startup (overnight or since last regular workday). Hand enter related data on previous day's schedule, Daily Summary Report, and Daily Patient Log.

p.m.

Prior to shutting down the system for the evening, do the following:

(1) Print hard copy of final version of current day's schedule.

(2) Print hard copy of Daily Summary Report and Daily Visit Log, and clear files used for these reports for next day's use. (Both reports are based on the Daily Patient Log.)

(3) Update Patient Master File from Master Excess File and clear latter file.

(4) Sign off.

DAILY REPORTS

For each regular workday, the following reports are produced:

(1) Final version of current day's schedule (may be compared to initial version taken in a.m.).

(2) Daily Visit Log, taken from Daily Patient Log, contains MRNO, name, doctor seen, diagnosis, verification code, and charge (fee) for each patient seen.

(3) Daily Summary Report (see description for procedure SUMMARY).

INFORMATION SYSTEM AND THE REAL WORLD EXPERIENCE

Information systems do not exist in a perfect world made up of mathematical formulas and flowcharts; computers must co-exist with and serve the humans who created them. The information system described in this paper has taken a compromise approach by combining several subsystems into one system. This approach can benefit from the advantages of both centralization and decentralization.

The preliminary results obtained by implementations of subsystems are very encouraging. The implementation of organized scheduling process would enable fewer doctors to attend a greater number of patients. This would certainly help in rural areas where there is a scarcity of doctors. The process of investigating the cost factors involved in operation and maintenance is currently in progress. The programming language used is PL/C and implemented on IBM 370/145 system.

CONCLUSIONS

This paper has considered the general issues in designing an information system for a pediatric clinic. The prototype design has been presented to show how a real life problem situation could be organized better with an information system.

At this point, the preliminary results show that fewer doctors and more patients can be served with an organized scheduling process and a good data base built into the information system.

Thus the importance of the present study lies in showing the large pay off which was found when an information system design concept was used in a medical clinic.

REFERENCES

1. R. L. Ackoff, Management misinformation system. *Mgmt Sci.* **14**, 4 (1967).
2. A. M. Lee, *System Analysis Frame Work.* MacMillian, London (1970).
3. S. S. Iyengar & V. Alia, A model of an information system for a city planning agency. Sent for publication to *Op. Res.* (1979).
4. R. N. Mason & I. Mitroff, Programs of research on management information system. *Mgmt Sci.* **19**, 475 (1973).
5. S. M. Padwal & R. Bandhopadhyay, Information system design in a commercial bank. *Ops Res.* **20**, 99 (1977).
6. S. H. Rouse & W. B. Rouse, Design of a model-based on-line management information system for interlibrary loan networks. *Inform. Proc. Management* **15**, 109 (1979).