CLASSROOM ALLOCATION:

A HEURISTIC APPLICATION OF APL

by

Miklos A. Vasarhelyi*

and

Jose Carlos Sardinha*

Submitted to

APL 76

Ottawa,

September, 1976

The authors are most appreciative of Ms. Rosemary Bitonti's help in the development and implementation of the system.

*Graduate School of Business Administration, University of Southern California

ABSTRACT

The classroom allocation problem is a classic application of operations-research based techniques. However, experience has indicated that in spite of optimization being desirable, the main problem that affects classroom management is the constant change in problem specifications.

This paper describes a heuristic, online system developed initially for the Catholic University of Rio de Janeiro and now being implemented at the School of Business Administration of the University of Southern California. The system is programmed in a very restrictive APL version of 24 K workspaces that creates interesting data coding, decoding and storage problems. The paper initially describes the classroom allocation problem and its possible solutions. The second part discusses the system analysis with data flows and possible reports. The third part presents the APL implementation, file organization, data features and workspace management. Finally, the conclusions discuss the possible expansions of the system under a more complete APL system, the usage of APL for such a system and other system interfaces to be employed along with the class management subsystem.

I. INTRODUCTION

The literature in management science describes a large number of resource allocation models. Schroeder (1973) surveyed a wide range of applications of management science to university management. One of the surveyed areas was space management which was defined in the following context: "resource allocation models relate the input of the educational process to the resources required."

CAMPUS (Judy and Levine, 1965) and RRPM (Gulko, 1971) are two examples of resource management models. CAMPUS (Computerized Analytical Methods in Planning University Systems) was designed at the University of Toronto in Canada to simulate university functioning at the individual course level. "Even though CAMPUS has been tested at several institutions, only limited usage to date has been achieved as part of the ongoing management program." (Schroeder, 1973)

On the other hand, he also discusses the RRPM (Resources Requiremenets Prediction Model) which is similar to CAMPUS, "the major difference ... is that RRPM is more aggregated ... The lowest level of aggregation in RRPM is the discipline level."

Another type of approach involves models that use mathematical formulations and optimization to deal with the classroom management problem. One of the most interesting of these formulations is based on the application of Markov chains to classroom allocation. "In these formulations a state is usually the student's grade level (freshman, sophomore, etc.) and perhaps his major. The number of students in each state then depends in a Markovian fashion on the numbers in previous states, transmission rates and the new admissions." (Schroeder, 1973). Markovian processes are especially used in the predictions of student enrollments (Gani, 1963).

The classroom allocation problem is analogous to the facility utilization planning problem in an industrial environment. It is of major importance for resource management and can have substantial impact upon global facility planning, efficient resource utilization and cost efficiencies.

Anthony et al. (1965) define resource management systems as "all procedures for collecting and processing recurring quantitative information that: (1) relates to resources and (2) is for the use of management... This definition excludes all nonresources ... and the systems are orginarily described in terms of the flow and processing for information...."

The classroom allocation system described in this paper is the actual conversion of a manual classroom control system to a terminal based allocation procedure that follows basically the same procedural steps. Therefore, the system herein described is heuristic in nature, implementing procedural rules. It attempts no optimization effect.

II. THE GENERAL PROBLEM

The classroom allocation problem is a subset of the more general procedure of matching three resources and a service user - respectively faculty, courses, classrooms and students. Curriculum and budget considerations are the key factors in the decision of what courses are to be offered in a certain semester. The classroom allocation process has no effect on this initial scheduling effort. This paper deals specifically with the problem of allocating a limited number of scheduled courses to the available classrooms within a series of key parameters.

A series of courses with differing levels of expected enrollment is scheduled for a semester. These courses have already been scheduled to certain days and hours and sometimes have listed faculty classroom preferences. The objective of this system is twofold:

1. Provide efficient allocation of classrooms to courses
2. Help in everyday classroom management

The general allocation problem has many dimensions including: classrooms have specific characteristics, student enrollment at scheduling time, faculty have classroom preferences, course scheduling is considerably heterogeneous with hours fully booked and hours not scheduled at all, distances between classes can make students late to class.

Our problem at the School of Business Administration of the University of Southern California is somewhat simpler. There are about 210 to 250 courses scheduled each semester and 25 different classrooms available. A series of parameters has to be established by systems analysis.

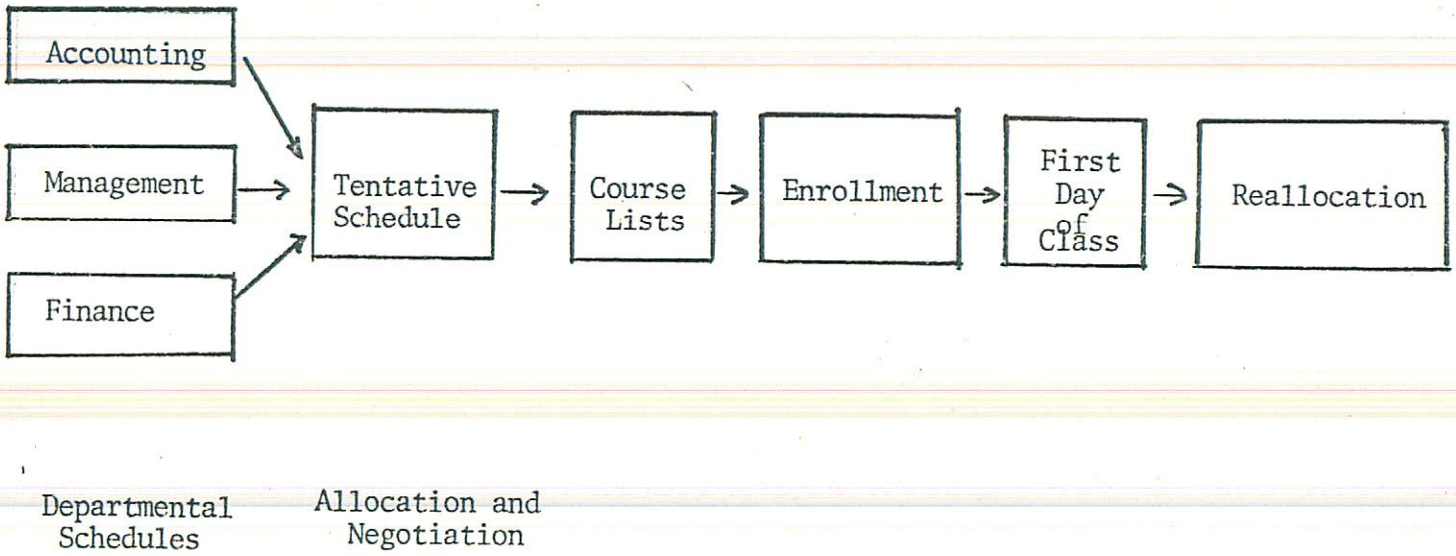
Systems Analysis

Our first endeavor is to establish how the system works in its manual form and the parameters which determine the decision process.

It was established that the present manual system considers the following parameters: days of the week, class hours, type of classroom desired, expected course enrollment and classroom preference. Classrooms are then allocated based on these parameters and on traditional classroom allocation. The original system alloted classrooms to departments and these did their own classroom scheduling. The present system is somewhat more centralized but still gives classroom preferences to departments. Figure I summarizes the manual classroom allocation process. Other factors are also of importance: it was noticed that keeping track of temporary classroom allocations for special seminars, lectures requiring special classroom characteristics etc. was of major

FIGURE 1

CLASSROOM ALLOCATION PROCESS



importance for the system. Faculty and staff would call up the resource management staff and request classrooms. These would be allocated for a specific day or days with notes made in the classroom book. A second problem of importance involved differences between the real class enrollment and the expected one. Classroom reallocations were necessary after the first day of classes.

These observations and others of lesser importance lead to the reformulation of objectives in greater detail:

1. Provide efficient classroom allocation
   Provide means of efficient class reallocation
   Decrease allocation effort involved

2. Help everyday classroom management
   Provide weekly status reports (temporary and permanent)
   Provide inquiry facilities
   Provide utilization summaries

Specific Parameters

The problem analysis also led to the development of specific parameters concerning the general problem as applied to the School of Business.

Class hours can be batched into six categories each of one hour and forty minutes. Three different types of classes and six classroom sizes were identified after careful analysis of specific requests. All these parameters are listed in Appendix 1.

Manual Allocation Procedure

The following steps detail the manual classroom allocation process:

1. Determination and examination of class schedules, i.e., which course, which day, which time.

2. Preliminary allocation of classes to classroom based on faculty preference and expected enrollment

3. Communication with department chairmen concerning non-fulfilled preferences, time schedule conflicts, etc.

4. Preparation of final version of class schedules

5. Adjustment of semester schedules due to increase or decrease in the number of sections, addition and deletion of courses, etc.

6. Examination of actual classroom enrollment after the first and third weeks of classes for reallocation purposes

7. Reallocation in "problem cases"

8. Establishment of a semester "basic schedule" to be used for temporary allocations

9. Temporary allocations for each week.

Courses at USC have a three digit departmental number and an unique "class number" which identifies different sections of the same course or courses of the same number in different departments. For instance, the course called Accounting Concepts and Financial Reporting taught by Dr. Miklos Vasarhelyi and scheduled for Mondays from 7:30 to 9:10 P.M. is designated ACCT 510 and has a class number of 20001.

Further Considerations

The above analysis of parameters identifies four critical areas of the problem: 1) scheduling and information flow; 2) allocation and renegotiation; 3) reallocation after the first classes; and 4) temporary allocations and classroom usage management.

A review of the literature indicates that some types of optimization are possible for this problem, however, the deeper analysis of the magnitude of the factors involved in the decision process described in this section led to an heuristic approach in the classroom allocation problem.

III. THE SOLUTION

The system was developed in the APL/360 standard language. The USC implementation of the APL language is a very restrictive one with 24 K workspaces and no file capabilities. This leads to interesting data packing and unpacking procedures within the system. Also, it places serious limitations on the utilization of this system in larger problems.

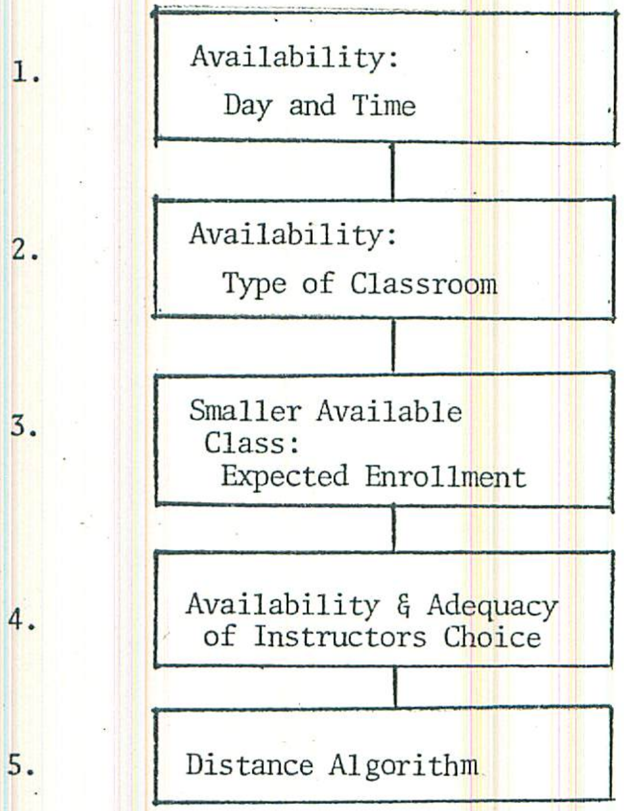
Program Logic

The system design was based on the heuristic implementation axiom. This axiom assumes the problem to have idyosincratic characteristics which cannot be programmed into a optimization routine. These characteristics include features such as classroom preferences by older faculty, departmental priorities and other specific considerations only relevant in the semester being considered.

The detailed study of the manual system and led to the computer based implementation of those steps described in the earlier section. The logic of requests along which the program was developed can be seen in Figure 2. This program logic adopts the manual system steps and enhances with two features: careful data entry checks and a distance alogrithm to resolve final selection once all the other requisites are fulfilled.

FIGURE. 2

ALLOCATION HEURISTIC SEQUENTIAL CHOICES*



*Availability checks for all possible classrooms that fulfill all the requirements up to that step. Step four checks for availability and adequacy of instructor's choice. If no unique solution is available, the classroom that presents the least distance to the department will be chosen.

Data Organization

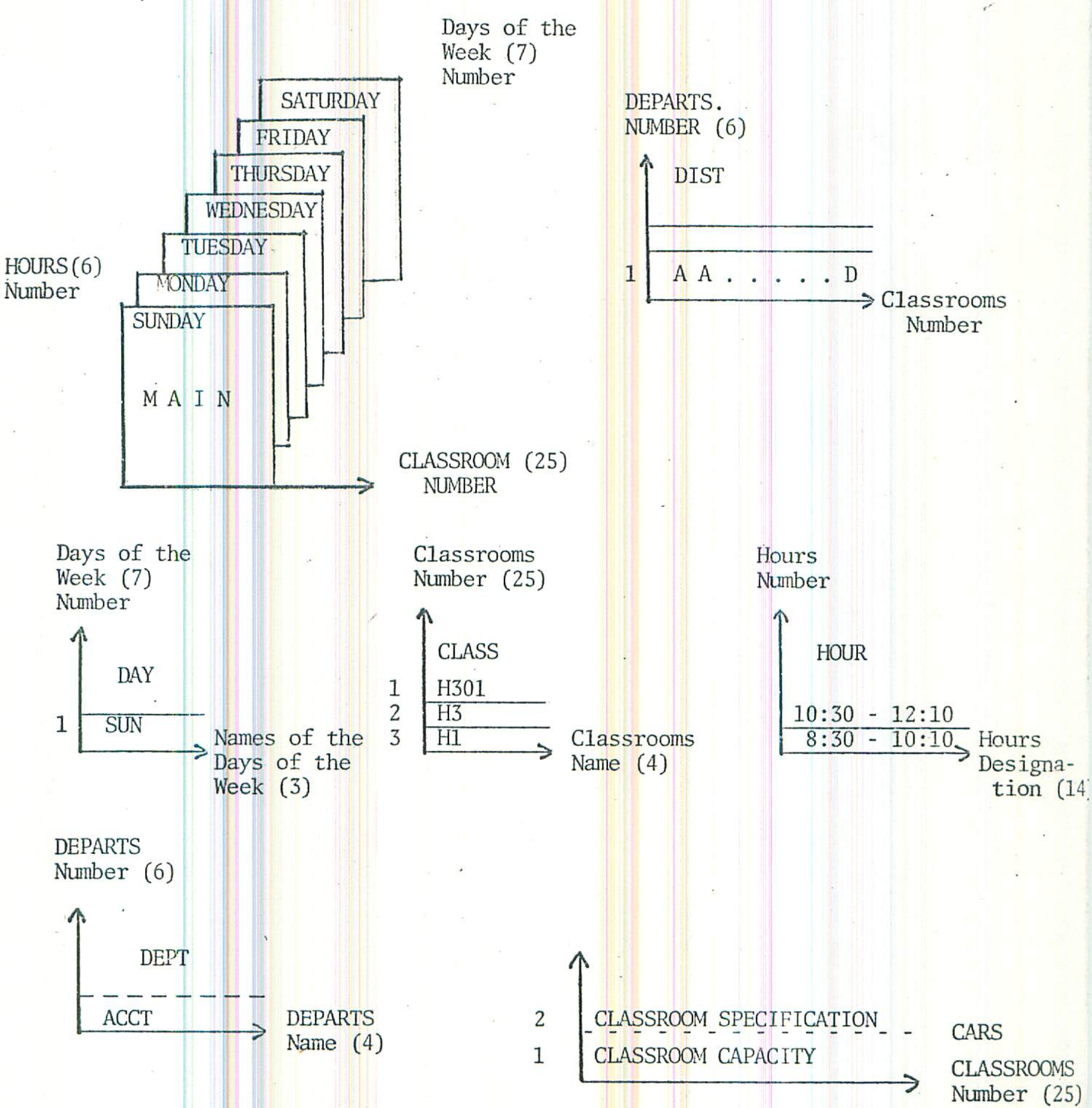
The data is structured into seven matrices, all of which contain well packed information. The most important matrix (MAIN) keeps track of space allocation in a three-dimensional array. The first dimension is the day of the week, the second the time-slot and the third a list of all classrooms in the system. Data packing stores specific course information into the MAIN matrix by means of a 9 digit number. The first digit specifies the department, the next three the course number and the last five the class number. The course ACCT 518 20001 is represented as 151820001. Figure 3 displays the structure of the data arrays. The other six matrices store information concerning the school (DIST, CARS, DEPT, CLASS) or general print parameters (DAYS, HOUR). All information (except the Matrix MAIN) is compacted into character representations (1 character per byte) to avoid integer representation (1 number per 4 bytes).

The matrix of distances (DIST) has 6 rows and 25 columns. The rows represent the departments and the columns represent each of the 25 classrooms available. Distances are coded into alphanumerics and represent approximations to the distances between the departments and the classrooms with penalties for stairs and a bridge.

The characteristics matrix (CARS) has 2 rows and 25 columns. The first row has classroom characteristics (M for moveable chairs, F for fixed chairs, D for desks, etc.). The second row contains an

FIGURE 3

DATA ORGANIZATION



alphanumeric code for the maximum number of students that can be placed in the classroom. For instance, classrooms capable of seating 40 to 45 students are classified by the letter C.

System Organization

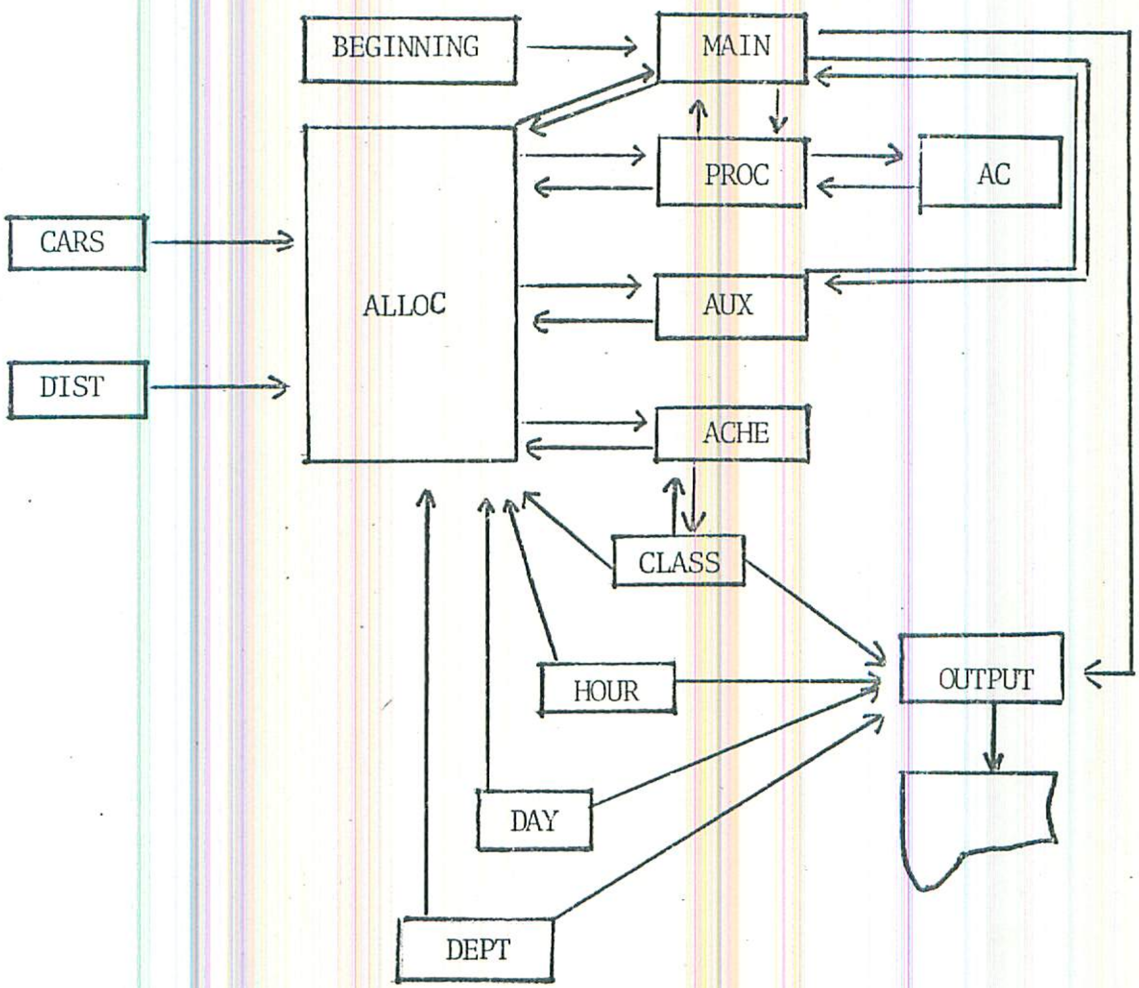
The system contains six programs that perform modularly all the initial functions in the system. One program (INI) sets up the system to be used at the beginning of each semester. The second and most important program (ASSOC) is used to allocate classroom requests to available classrooms. Requests should be placed within priority order as classrooms already alloted can not be freed upon later requests. The three additional subprograms perform allocation functions for the ALLOC program. Figure 4 displays the programs and their interactions with the database.

The initialization (INI) program zeros the MAIN matrix at the beginning of each semester. At that point all classes are available for the first course to be allocated. Then INI calls the allocation program (ALLOC) which will allocate the courses to the classrooms.

This program (ALLOC) is the key program for the system and its function is to check course data, find available classrooms and to reserve the chosen classroom for the course.

FIGURE 4

PROGRAM AND DATA INTERACTION



The output program displays the allocation status for each day of the week.

The three other subprograms have the following functions: (1) determine whether the course to be allocated is already in the system; (2) find the classrooms available; and (3) find (and check availability) of the preferred classroom.

Data Validity Checks

In order to avoid excessive data entry, the system requests only five inputs. The department name is entered by a number from 1 to 6 which indicates the six departments in the School of Business. The same form is used to type the days of the week and hours; the days of the week are entered by a number from 1 to 7 and the hours from 1 to 6, since there are seven days in a week and six periods of time per day in this system.

For each of these entries the system has data validity checking features. The system knows that a department number is between 1 and 6; the courses number can be between 100 and 700 and the class number between 10000 to 99999.

Another data screening feature checks to see if the course being entered has been previously allocated.

The data validity check for the stream of days and times is composed of three steps: (1) checks on whether the vector has an even size (day, time; day, time; ...); (2) if the day of the week is between 1 and 7; and (3) if the time is between 1 and 6.

The classroom type is examined to see whether it falls among the four allowable categories: T F M A. (See Appendix 1)

Man-Machine Interfaces

The system is designed for ease of man-machine interaction. Questions are objective and short with a help mode available for data entry errors. For the specific case of the School of Business of the University of Southern California an average of about 220 courses will be entered for each semester. Therefore, if each entry takes 1 minute the total data entry time will encompass about four hours. This will create considerable boredom to system manipulation. Extensive verbosity in terminal dialogue would create even more problems. Therefore, dialogues are designed to be concise and objective with error messages and exception reporting carrying the weight of improving the quality of man-machine communication.

Later versions of the this system will encompass variable machine user dialogues as proposed by Vasarhelyi (1973) which will contain initially long and carefully explained sentences and progressively shorter and shorter prompters.

APL as THE Programming Language

Three interactive systems are available at the University of Southern California: BASIC (on an HP 2000F), TSO and APL (in a 24 K miniversion).

The BASIC system was not used due to the extended programming difficulty that use of such a system would present. Our experience in the development of an interactive stock market simulator (Vasarhelyi, Mock and Allen, 1976) showed that the coding effort and system capabilities are interrelated in an exponential manner in that the addition of a few more sophisticated features required extensive reprogramming in order to be inserted in the BASIC version of the simulator. Meanwhile changes in the same simulator under its APL version were easy and simple to perform.

TSO on the other hand presents very slow response time, high system overhead and considerable complexity in the preparation of man-machine interfacing programs. The proposed TSO system would entail programming in the PL/1 language and extensive JCL coding for the accomplishment of the desired tasks.

Among the three alternatives above APL was the obvious choice. Unfortunately our very restricted APL does not have File capabilities or the increased power of an SV or Plus-File version. These factors were somewhat restrictive and in the overall systems design created some problems. Fortunately the small dimensions of the problem under consideration permitted its compression into 24 K workspaces.

The specific storage characteristics of the APL language allowed the compression of all of the data of one class into one integer occupying four bytes. Therefore, the MAIN database, a three-dimensional matrix, occupied 4200 bytes (7 by 6 by 25 multiplied by 4 bytes per integer). Unfortunately workspace-size limitations did not allow the final preparation of the course schedule using the APL system as it would have required the storage of course name information for all courses (220 courses by 30 characters per course).

The APL matrix and vector manipulation features are extremely well suited for the solution of this type of problem. Data can be effectively stored in matrix format and searches, changes and initialization can be performed very compactly.

Present System Status

The system is in actual data usage testing at the present moment with data of the spring semester being entered in a parallel testing mode. Several changes in data entry format and system capabilities have been proposed and some of these implemented. A multiple workspace configuration will serve for temporary classroom allocations. Initally a basic workspace will be created allocating the classes as requested by the departments with the conflicts resolved by negotiation or priority assignment. The second step in implementation of the system involves receipt of actual course enrollment data. Preparation of summaries of class utilization and manual reallocation of those courses with considerable difference between scheduled and actual enrollment. This step will later be performed by a semi-optimization algorithm not yet implemented in this system. The third step involves preparation of copies of the basic workspace (with reallocations reflected) for four weeks in advance. Temporary classroom allocations will be made on these workspaces with an additional program to keep track of the names of those requesting these temporary allocations, and the nature of their needs.

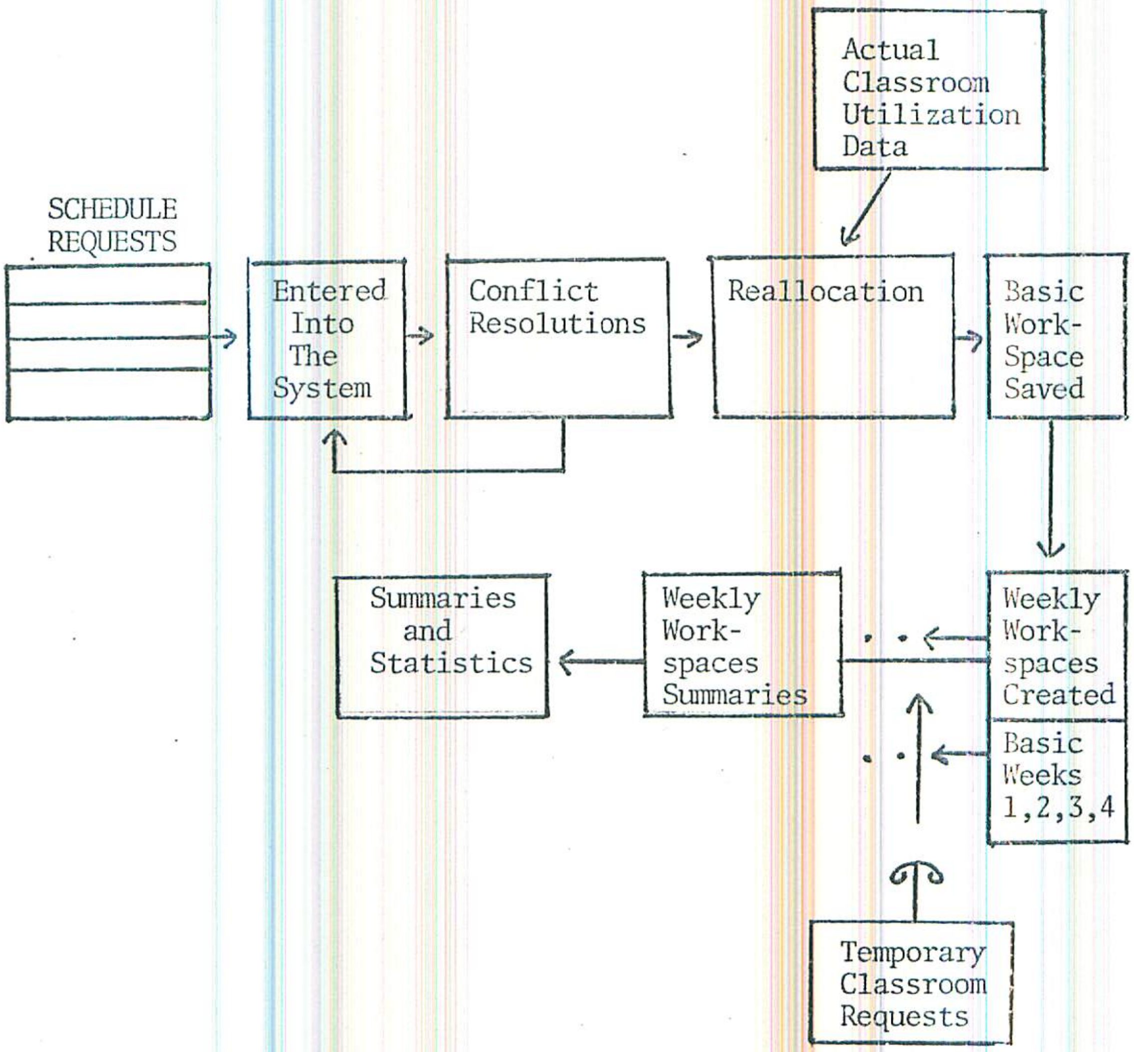
Figure 5 represents the present system operations status which is expected to go into full operation for the fall semester of 1976.

Additional Considerations

Summary, query and report capabilities are available. A summary indicates classroom utilization by type and day of the week for any workspace status. Summaries are useful for facilities planning and temporary classroom allocations. The query facility finds the specific allocation of a class from the MAIN matrix.

FIGURE 5

COMPUTERIZED SYSTEM OPERATION



Report displays a complete course allocation schedule at the moment of its request.

Additional auxiliary programs are progressively being made available upon the request of the user or perceived need by the designer.

IV. CONCLUSIONS

The system has its pros and cons. Ultimate evaluation of its desirability will be made considering its labor requirements of data entry and manipulation vis-a-vis the relative simplicity of a manual system. This examination will have to be made under the light of its added classroom management facilities and the possibilities of its enhancement and interface with other types of models.

The interpretative nature of APL might present some problems in many commercial applications of considerable number crunching. This is not seen as a limitation of the present system which utilizes APL's positive features to the full extent in an environment of constant program change.

Some system analysis simplifications might present some problems in the future. The divisions of class hours into six possible slots allowed storage simplification but might cause classroom allocation inefficiencies in the event of one hour type lectures. Additional classrooms or courses might require deletion of Saturdays and Sundays from the system or their treatment in special ways.

The next step in system development will be the reallocation algorithm and the study of the possibility of the interface of this system with other systems in the school's Student Information System.

In conclusion, our experience says that APL is a viable language for commercial or university applications of specific characteristics. It is our feeling that this sytem will easily justify its costs. The usage of the APL language allowed considerable time savings in coding which will be reflected in the further sophistication of the system and the improvement of its man-machine interfaces.

V. BIBLIOGRAPHY

1. Anthony, R. N., Dearden, J. and Vancil, R. F. Management Control Systems. (Homewood, Illinois: Richard D. Irwin, Inc.) 1965.

2. Gone, J. "Formulae for Projecting Enrollments and Degrees Awarded in Universities," Journal of the Royal Statistical Society, A126, 1963, pp. 400-409.

3. Gulko, Warren W. "The Resource Requirements Prediction Model 1 (RRPM-1): An Overview," TR-16, NCHEMS, January 1971.

4. Judy, R. W. and J. Levine. "A New Tool for Educational Administrators," Report to the Comission on the Financial of Higher Education University of Toronto, Toronto, 1965.

5. Mock, T. J. and Vasarhelyi, M. A. APL for Management. (Los Angeles, Wiley, Becker and Hayes), 1972.

6. Nelson Associates, Inc. "Increasing College Capacity by Calendar Revision." A report to the State University of New York, (Albany, New York), 1961.

7. Schroeder, R. G. "A Survey of Management Science in University Operations." Management Science, Vol 19, No. 8, April, 1973, pp. 895-906.

8. Vasarhelyi, M. A. "Simulation: A Tool for Design and Pre-Implementation Testing of Large Scale Software Systems." Proceedings of the 1971 Winter Simulation Conference, New York, December 1971, pp. 261-267.

9. Vasarhelyi, M. A. Man-Machine Planning Systems: A Cognitive Style Examination of Interactive Decision Making. Unpublished Ph.D. Dissertation, Graduate School of Management, University of California, Los Angeles, 1973.

APPENDIX 1

SYSTEM VARIABLES BEING USED

CARS
TTMMFFMMFFTTFFMTTTMTMTTTT
EEBBCCBCCCAACCCDDBCBBCDAC

CHAR
ABCDEFG

CLASS
H1
H2
H301
H302
H303
H304
H305
H306
H401
H402
H403
H404
H405
H406
B4
B5
B103
B104
B204
B205
B206
B207
B302
B303
B304

DAY
SUN
MON
TUE
WED
THU
FRI
SAT

DEPT
ACCT
MKT
MANG
FIN
QBA
FOOD

DIST
FFBBBBBBCCCCCCEEDDBBBBAAA
FFBBBBBBCCCCCCEEDDBBBBAAA
DDBBBBBBAAAAAAFFEEDDDDCCC
DDBBBBBBAAAAAAFFEEDDDDCCC
FFBBBBBBCCCCCCEEDDBBBBAAA
DDBBBBBBAAAAAAFFEEDDDDCCC

HOUR
8:30 TO 10:10
10:30 TO 12:10
1:30 TO 3:10
3:30 TO 5:10
5:30 TO 7:10
7:30 TO 9:10

APPENDIX 2

EXAMPLE OF ALLOCATION INTERACTION

```
      )LOAD SPRING
SAVED  15.35.14 02/17/76

      ALLOC
ENTER DEPT, COURSE AND CLASS NO. (EXIT= 999)
⎕:
      1 518 20002
ENTER SUCCESSIVELY: DAY, HOUR, DAY....
⎕:
      22 44
IMPROPER WEEKDAY
ENTER DEPT, COURSE AND CLASS NO. (EXIT= 999)
⎕:
      1 518 20002
ENTER SUCCESSIVELY: DAY, HOUR, DAY....
⎕:
      2 4 4 4
SPECIFY MAXIMUM NUMBER OF STUDENT
⎕:
      25
SPECIFY TYPE OF CLASSROOM
T
ENTER PREFFERED CLASSROOM OR (A=ANY)
A
COURSE NUMBER: ACCT 518 20002
WAS ALLOCATED TO :
B303  MON   3:30 TO  5:10
B303  WED   3:30 TO  5:10

ACCEPTABLE? (YES OR NO)
YES
ENTER DEPT, COURSE AND CLASS NO. (EXIT= 999)
⎕:
      999
TYPE )SAVE FALL (OR SPRING)
```