Marvin Westrom

H is for "heuristic"
HARTS

Can it be that one of the great traditions of North American schooling is about to go under at the hands of - sorry, on the screens of, the microcomputer? The time-honoured opening ceremony of the mass timetable foul-up, followed by the annual gathering of school principals at their wits end, may indeed be on its way out if Marvin Westrom's account here is not a fantasy. He describes how a computer may be programmed to behave heuristically, responding immediately with its resources to the intuitive and creative judgments of a principal who has been compelled to fiddle with the parameters of his or her Master Timetable, under the stress of ceaseless changes in the variables it must cope with. And this program exists. If he sells a few, as well he may, only the students for whom the customary confusion of September affords a cushion, against the shock of back-to-school, can have cause for complaint.

The purpose of this paper is to explain a new and effective method of handling some major school administration tasks. It depends upon the microcomputer and is integral to the HARTS school administration system. HARTS is described in Appendices A and B.

"Heu-ris-tic, a.(Gr. Heuriskein, to invent, discover). helping to discover or learn; specifically, designating a method of education or of computer programming in which the pupil or machine proceeds along empirical lines, using rules of thumb, to find solutions or answers." - Webster's dictionary

In computing, the heuristic technique is used mainly for the most difficult tasks - problems that cannot be solved with brute- force computation. Such tasks usually arise from attempting to make the computer mimic human activity. For example, playing chess, understanding speech, and reading handwriting are relatively simple for humans; but computers can do these tasks only with extensive heuristic programming.

Making a master timetable and scheduling students is a difficult task, even for humans. Certainly it is more difficult than playing chess. There are traditionally three approaches to the solution of this problem for a school principal:

1. Hand Scheduling. The scheduling person or committee solves the problem mentally (usually using a large holding board and bits of paper to indicate classes, rooms, teachers, etc). If the school has no more than 300 students and the schedule is not overly complex, this is often the best means for solving the problem.

2. Arena Scheduling. The schedulers prepare a Master Timetable which contains sufficient class positions for all students. Students then come to the Arena (or gymnasium) and attempt to enroll in the classes they most want by trading a card with their name on it (student card) for a card with the name of the course on it (course card). At completion, each student has a set of course cards for the courses in which he is enrolled (a student timetable), and the school has a set of student cards for each class (class lists). This procedure will solve the problem, but in a way that is usually not completely satisfactory. Students design a personal timetable only to find that all of the cards for a class they want are gone; now they must re-design their timetable. The process is characterized by long lines of waiting students, bewilderment, disappointment, and frustration. Some students will be unable to take the courses they should, because of an unusual arrangement of other courses they must take. Seeing a particular problem, the principal might be able to change the Master Timetable to accommodate a student, but it is now too late; too many other students are already scheduled, and it is impossible to foresee all the effects that a timetable change would have.

3. Computer Scheduling. Students are asked to identify the specific courses they wish to take next year. This information is coded into computer form and counted to provide the principal with preliminary information (Tallies and Conflicts) to make the Master Timetable. When the timetable is made, it too is encoded and the computer attempts to 'load' all student course requests into the timetable (this process is sometimes called a 'simulation'). Again, it is probable that all students will not get what they require, but this time the principal can examine individual problems and make changes to the master timetable. These changes are coded, and the computer can 'load' the students again to see if the changes have caused improvements. Over several months, the principal

can make five or more different timetable trials and attempt to optimize the solution.

A fourth approach, the subject of this paper, is herein called Heuristic Scheduling.

Heuristic Scheduling is Computer Scheduling with a human (usually the principal) in control of the process. The programs do not operate heuristically, but the problem-solving process is heuristic, controlled by a human supervisor. The supervisor weights the uncertain factors and makes guesses about which changes are likely to have the most beneficial effects; the computer does the calculations, projects the effects of these decisions, and provides detail and summary information upon which further refinements can be made. Rather than attempting to arrange students in classes in a brute-force computational way, an interactive method is used; one that combines the best capabilities of the human, to make creative decisions and judgements, and of the computer, to do computations and to assemble summary results.

Heuristic scheduling

HARTS consists of four component systems. The scheduling system is described in Appendix A. The other three systems (Registration, Attendance, Reporting) are described in Appendix B. In the hands of a skilled administrator, the HARTS Scheduling system is a powerful and efficient tool for optimizing the match of school capability with student requirements. The procedure for developing a master timetable and scheduling all students is as follows:

1. The school, consulting its teachers and considering its facilities, decides upon the courses it wants to offer. These are numbered and entered into HARTS, but can be changed at any time.

2. Students select the courses they would like to take (course requests) which are key-entered to provide basic information for timetabling and scheduling. Often these requests are collected in January and February, six months before the following school year. The later the collection can be done, the more accurate the requests will be, and the fewer revisions will be caused by students changing their minds. HARTS enables collection to be postponed until June.

3. HARTS tallies the requests and calculates conflict matrices so that the school can determine the number of sections needed for each course, place the sections in non-conflicting slots, and make a preliminary version of

the Master Timetable. Tallies and conflict counts are always exact and up to the minute. The effects of enrolments, withdrawals, even changed course requests are always included in the latest tallies, so that the Master Timetable can be adjusted according to accurate data.

4. The Master Timetable is entered or adjusted. This could involve adding or dropping courses, but normally it consists of altering sections of existing courses. The courses, with detail concerning each section, appear on the screen. The user can edit the Master Timetable with full information about how this (and accumulated previous changes) affects his school offerings and the teacher concerned. Many schools start scheduling in March or earlier. With HARTS, this can be postponed to as late as August, with final scheduling completed the first week of September when teacher availability, student enrolment, and final course requests are known.

5. Students are loaded or given timetables according to their requests by the HARTS scheduler. A 'simulation' for a medium-size school (1200 students) normally takes about five minutes. All scheduling summary reports are available at the touch of a few keys, so that 'rough' development and pruning of the Master Timetable can be accomplished very quickly. If finer adjustments requiring detailed analysis are indicated, numerous summary and detail reports can be printed out for more careful examination.

6. As the operator examines the results generated by the scheduler, and the students who didn't get the courses they wanted, he often adjusts the Master Timetable, and sometimes alters student course requests (If changes are made, go back to Step 4). Because of the short time required to do a simulation, some changes might be made and tried to test hypotheses - even unlikely hypotheses. It is this ability to 'fiddle' with parameters of this complex problem, at almost no cost, that is one key to heuristic scheduling.

   One principal had the school's course requests entered, took his Apple home for the weekend, and returned Monday morning with the whole school scheduled. He had run more than twenty simulations on Sunday afternoon. On a hunch, another principal tried swapping two seemingly unrelated courses; this resulted in eliminating nearly all the conflicts from his tenth grade. A third principal didn't know whether a teacher who retired at Christmas was going to be replaced until after school had resumed. He re-scheduled the school for both eventualities, and so was prepared for whatever decision the board made.

7. When the school feels that the Master Timetable is optimal, scheduling can continue at an individual level. Requests can be entered and the scheduler used to schedule individual students. Global systematic changes can easily be made if courses have to be added, removed, or compressed. Students who still didn't get exactly what they wanted and students who change their minds can ask counsellors to change their timetables. Counsellors have full-screen access to all students' timetables; they can make required changes and override class limits if necessary. Class lists are automatically adjusted.

8. Teacher assignments, timetables, and class lists are computer-printed. These are always accurate when they are printed. Changes may be made to the timetables that make the lists inaccurate, but then new lists can be readily printed at any time. The school knows which lists are inaccurate and how important this is. It has complete control over when new lists are produced, and new lists can be produced at a moment's notice.

The advantages of using an in-house microcomputer system such as HARTS is immediacy and close control of data and transactions; human skills are put to best use. Present computer systems are not capable of "understanding" a school's scheduling problems in the same sense as the human scheduler understands them. The computer does not accumulate experience; it has no insight, no intuition, no wisdom; it does not have to explain to teachers or students why they cannot have what they want (even though it may get the blame). Humans have these advantages, but they cannot compete with the computer in adding and comparing values, storing and retrieving data while never losing the smallest detail, and following procedures without making mistakes.

The problem is best solved by using the strengths of each and avoiding the weaknesses of both. For the human to participate fully, the scheduling procedure must be redesigned. The time interval between running a simulation and seeing the results must be very short; the human must be able to select the results he wishes to see and view these in summary as well as in detailed form; the computer data must be up to the instant (not days or weeks behind the school situation); and the customary remoteness and aloofness of the computer system itself must be removed. The system must be extremely reliable and available "on demand" to the scheduling function. It must be extremely easy to use, and not require the "looking up" of procedures to make it operate.

The school knows which changes are important and which are not. The computer system must be of a uniform smartness - not too smart; it must do the tasks which it does best and do them well, and leave the rest to the humans. The system must "trust" the user. It cannot protect the human from his own mistakes without making some necessary procedures overly complex. Any piece of data that can be entered must be capable of being edited, re-entered, or un-entered. Any result that can be calculated must be able to be re-calculated with revised data. Any particular piece of data should be stored only once; and no datum should be stored unless they it be needed many times.

The HARTS programs were designed with these principles in mind. They are fully documented and easy to use. The screen displays are self-explanatory and the functions performed are both necessary and obvious. Most users don't bother to read the documentation, yet they are still able to use HARTS with fluency. Where needed, the programs provide full screen editing; one can exit at any desired point. The system produces highly specific reports as opposed to voluminous everything-you-might-want-to-know reports necessarily generated by batch processing systems. The programs perform repetitive complex calculations and analytical tasks without making the user wait. All complex routines are written in machine language, and disk access is five to twenty times faster than through Apple DOS. If the computer goes "down", the whole machine can be replaced (often with another machine right in the school).

The microcomputer boom, now almost five years old, has sent ripples into nearly every aspect of human endeavour. Microcomputers have obvious limitations when compared to their big brother main-frame computers. But like a knife compared to a sword, they have advantages as well. Microcomputers today are as capable as, and more sophisticated than, the main-frame computers of 20 years ago. Certainly many sophisticated businesses relied only ten years ago upon computers less powerful than the APPLE. The micro has put a power that once was available to only a few into the hands of everyman. Heuristic scheduling is one of the many new techniques made possible by the microcomputer revolution.

APPENDIX A

HARTS Scheduling

The HARTS Scheduling system contains eight main functions - listed below in approximately the order that they would be used.

Course information. The user controls the name of each course (a long name for external reporting and a short name for internal use), the term (first, second, or both), the number of periods, and the class limit. Lists of courses in a variety of formats can be printed.

Course requests. Student course requests (12 courses, or ten courses and two alternates) are entered and edited. General course requests can be entered automatically (for example, every student in grade eight could be given Math 8 as a request). The requests can be printed in a long form (suitable for filing, giving to students, or sending home to parents) or in a short form (for in-school examination and timetabling).

Tallies. Tallies (counts) of the number of students requesting each course, and Conflict Matrices (counts of the number of students wanting two, three, or four courses simultaneously) are produced. Conflict matrices for every possible pair of courses can be generated, but this is not often done because many pairs are known to be dependent or independent. HARTS can calculate just those counts that are needed.

Master Timetable. Using the tallies and conflict results, the schedulers create a Master Timetable by specifying the sections each course is to have. The teacher, the term(s), and the room are entered (or the teacher and room can be postponed for later). The Master Timetable can be printed in course order or in teacher order (teacher timetables).

Schedule. The scheduler looks at a student's course requests and determines the sections of each course in which he or she is to be enrolled. It normally does this at about four students per second. While doing this it keeps class sizes the same size (as far as possible), and schedules every student that can possibly be scheduled. At the same time it prints a running count on the screen of students who were completely scheduled, those for whom it had to use alternates, those who didn't have a full set of requests, and those whom it was able to only partly schedule. Within this program, the user can examine the balance among class sections, look at the 'Unsatisfied Demand' (a count of requests that the scheduler was unable to fill), and examine individual student timetables.

Global Timetable. Specific course sections can be selected for groups of students. For example, the system can be instructed to find every student in Grade 8 taking section 3 of Physical Education, and enroll them in section 2 of Guidance, provided they are not already enrolled in Guidance and that the period (block) is not already filled with another subject.

Student Timetable. Any individual student's timetable can be examined, changed, and/or printed at any time. A full-screen editor enables the user to alter any item readily, display the results immediately, and update the school records and class lists. The function is particularly useful for students who enroll late. The counsellor can build the student's timetable right on the screen; discuss possibilities with him or her, and print a copy for the student as soon as it is complete.

Class Lists. Class lists can be printed in register order (student number) or alphabetical order. They can be printed in four different formats, with combinations of student number, name, squares, address, and phone number. They are always accurate at the time of printing.

APPENDIX B

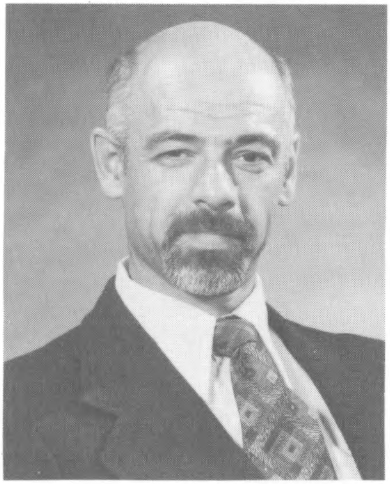
HARTS III

HARTS III was piloted in the spring of 1981 at Maple Ridge Secondary School in British Columbia. Since then, the system has gone through many stages of revision and development to become a complete and powerful school administration software package. It is in use across North America. HARTS consists of four basic component systems: Registration, Scheduling, Attendance, and Reporting (Report Cards). It contains about thirty programs, which manage three megabytes of school data. The current system can handle 1558 students in six groups (grades), up to 255 courses, 99 teachers, and 12 courses per student. It is the Scheduling system which is of most interest here, but a brief description of the other three systems is presented to provide a context.

The Registration System is conceptually very simple. Its purpose is to enable management of basic information about the students and teachers in the school. Student names and demographic information (such as their parents' or guardians' names, address, division, and phone number) are collected along with teacher names and teacher information (such as division and homeroom). The system produces a variety of student lists (such as division lists, grade lists, an alphabetical list of all students in the school) and miscellaneous reports (such as counsellor cards, mailing labels, age/sex tally). It contains facilities to reassign student numbers (into alphabetic order) and maintain the student log. The log keeps track of dates of all enrolments and withdrawals so that any student's history can be examined; a report showing students "in" and students "out" by month is available.

The Attendance system provides the school with a complete facility to collect, edit, and report all attendance data. Students are marked Present, Late, Absent, Holiday, Withdrawn, Excused, or one of two other user-defined categories for AM or PM homeroom and/or any period. Up-to-the-minute individual reports, daily, monthly, and yearly summary reports, and attendance statistics reports (frequency distributions with percentages by attendance state, by month, and by group) are available; and searches can be performed to determine, for example, which students have missed three or more classes in the past week. The school can define attendance summary periods for the generation of special summary reports. All necessary information from the Registration and Scheduling systems is available; in particular, enrolments and withdrawals are automatically posted and student records adjusted. Class attendance and home room attendance data are forwarded to the Reporting system for printing on report cards.

The Reporting system keeps accurate student achievement data and produces report cards for up to five reporting periods. For each student for each of four reporting periods, the system maintains homeroom attendance and a course mark, a work-habit code, and class attendance for each class; plus two comment codes, and a final mark for each class. Percentage marks (0 to 100%) or user-specified two-character letter grades may be used. Data may be key-entered by class (system prompts with student names) or by student (system prompts with student courses), or entered by OMR cards. Two hundred and fifty-five teacher-controlled comments and two "broadcast" comments are available to be edited by the school and printed on report cards. A variety of reports is available: mark lists and confirmation lists for teachers, short-form report cards for counsellors and principals. Marks are tallied (symbols and percentages) by teacher or by course, and detail and summary reports produced. Student average marks are calculated and reported as both GPAs (Grade Point Averages) and percentages, in ascending or descending mark order, or in student order; upper and/or lower bounds on GPAs or percentages can be set (to produce Honours List or Failure List among others); optionally, each student's actual marks can be included. Mark searches can be made; for example, all students with missing marks or those with three or more 'A's could be listed. Of course the system also produces report cards with computer speed, and with more flexibility and accuracy than can be provided in any other way.



Marvin Westrom, formerly a teacher of mathematics in rural Alberta, has worked on educational software for IBM Canada and for the National Research Council. He designed the CAI language Natal, and produced HARTS, a system that runs of the Apple. He is at the Faculty of Education at the University of British Columbia, and is an active member of several editorial committees and advisory boards on computing in education.