AN INFORMATION SYSTEM FOR THE CONTROL OF HIGH SCHOOL TRUANCY

Anthony Horvath

A Thesis

in

The Department

of

Computer Science

A Thesis
Submitted to the Faculty of Graduate Studies
in Partial Fulfilment of the Requirements for the
Degree of Master of Computer Science

Concordia University

Montreal, Quebec, Canada

June 1982

ABSTRACT

An Information system
for the Control of High School Truancy

Anthony Horvath

This thesis describes the implementation of a cost-effective monitoring system designed to meet the multiple needs of the Quebec comprehensive high school. It describes the design details and results of the interactive pseudo-batch system designed to tackle this problem.

The effectiveness, scope and range of such a system is considered to be critical to the overall performance of the school. The system had to meet the diverse requirements of the students, parents, teachers, truant officers, vice-principals, principal and provincial department of education. Therefore it was necessary for the system to provide various levels of information.

In order to be able to find the ideal solution, a

novel structured analysing methodology was used to design the optimal information system. Instead of studying a proposed system only from the input to the requested output stage (which in a sense is too limited to the computer and too inadequate to evaluate the resulting informations' overall effect on the organization), a five step information system analysis was introduced that goes beyond that and considers a system from data generation to the data usage level. In this way the outcoming system will functionally blend in with the organization on the operational level. These five steps are: data generation, data collection, input to output processing, information distribution, information usage. The ideal system will provide the user computer assistance in all these steps. In this study ABL presents a great advantage for the modelling and implementation of the outcoming system. It is a most effective tool to achieve the functionnal breakdown of a system to its components and permit the logical simulation of it.

By studying existing systems with the help of this methodology, it was possible to design a unique information system for attendance monitoring. It combines a series of user controlled on-line and batch processes that covers all aspect of student attendance control. The software and the techniques developped make the system simple to apply by the personnel, flexible to operate, highly accurate and fast regardless to the volume of the data transactions. It offers detailed attendance related information accessible with no delay through a terminal.

The system was implemented over several years. First in a small school of 500, then in a medium size school of 1200 and finally in a big comprehensive high school of over 3000 students, where the participation of nearly 200 people had to be synchronized at every lecture period. The system was evaluated a success where four other previous attempts failed.

ACKNOWLEDGEMENT

Here I would like to express my thanks and greatest gratitude to Mr. André Rousseau, Deputy-Minister of Education who, in 1974, personnally authorized this research to be undertaken and financed by the Ministry of Education of Quebec, through the Chambly Regional School Board. In the same school board, I would like to express my thanks to the following people: to Mr. Jan Palkiewicz, Director of Research and Planning, who was more of a collegue than a superior and whose constant support made the impossible to become possible, to Mr. Rolland Dechenaux, Coordinator of the Computer Center, who gave me the initial data from which the school's data base have been set up, thus saving a lot of time, to the administrative and teaching staff of three High Schools: Mont-St-Gabriel, Collège de Longueuil and André Laurendeau, who throughout these years worked with me on the succesfull implementation of this system, to Mr. Hugh Maire, Vice-principal in charge, who coordinated the operation on the school level and evaluated its performance.

At Concordia University, I would like to express my gratitude to my Thesis Director, Dr. W.M. Jaworski, whose patient guidance and personnal example was a constant stimulus to reach for a better solution. I sould like to thank also Mr. Ivan Fuchs, Director of the University's computer center for his consideration when we were faced with implementing the system on a large scale and needed computer resources. I would like to thank also Mr. Kevin O'Mara and Ms. Diane Eddy for their help at the writing of this thesis.

At home, I would like to thank Ann, Paul and Judy for their understanding about my odd and long working hours.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

CHAPTER 1

INTRODUCTION

This thesis describes the implementation of a cost-effective attendance monitoring system designed to meet the multiple needs of the Quebec comprehensive high school.

It describes the design details and results of the interactive pseudo-batch system designed to tackle this problem. The effectiveness, scope and range of such a system is considered to be critical to the overall performance of the school (34). The system had to meet the diverse requirements of the students, parents, teachers, truant officers, vice-principals, principal and the provincial department of education. Therefore it was necessary for the system to provide various levels of information. Figure I illustrates the structural hierarchy observed within this system.

To be of value to the role of the school's truant officers the system must be able to easily provide quick, detailed and particular information on the attendance record of any student.

To be acceptable to the school's teachers it must minimize the effort involved in taking an accurate roll call of each class while allowing them to correlate a student's progress in a given course with attendance records and grades. With this system the teacher has at his disposal the accurate and powerful tool needed to counsel and advise a student and his/her parents. It is possible for any teacher to easily analyse the academic progress and attitude of any

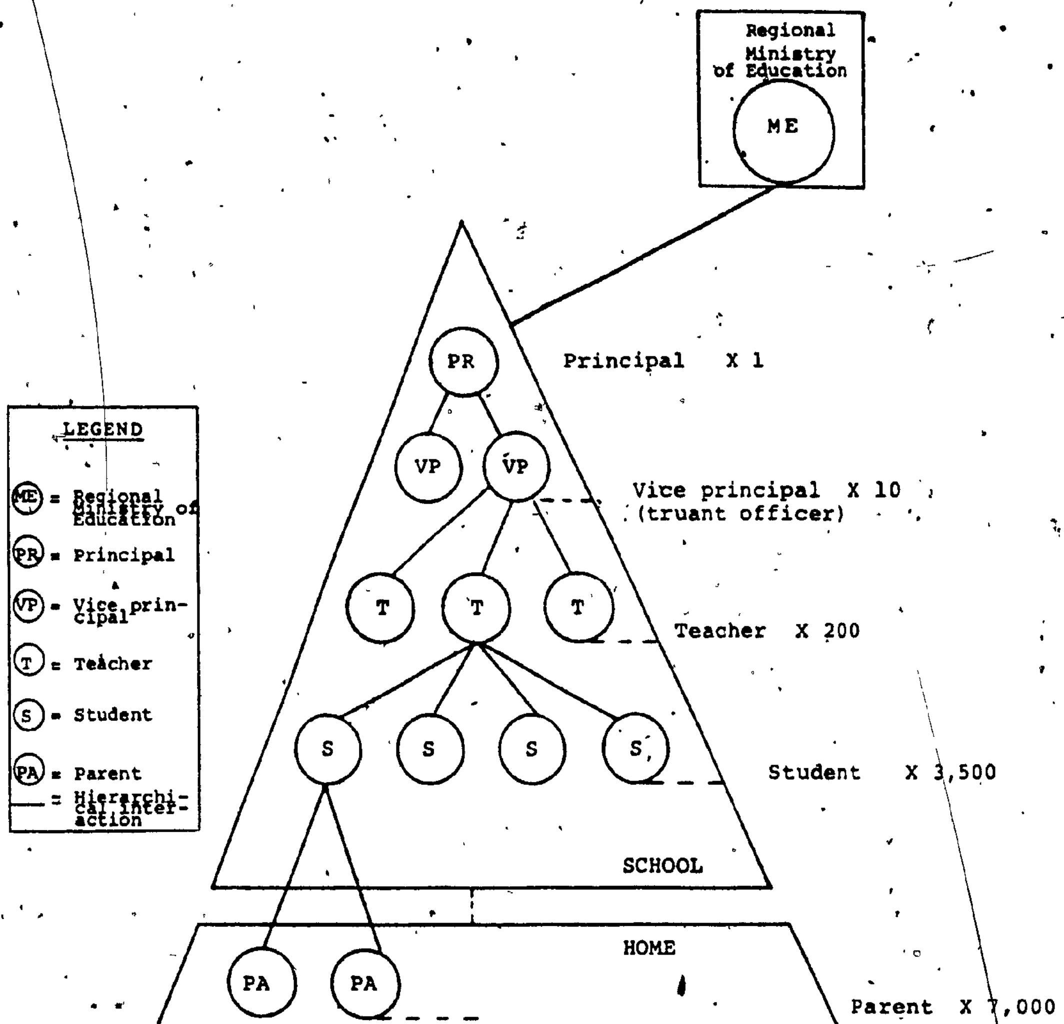


Figure 1 - User levels. The structural hierarchy observed within this system and also the symbols used to represent them throughout this thesis.

one of the couple of hundred students under his charge.

The system provides the pupil's parents with up-to-date, accurate information on their child's attendance record. It also provides this information on any pupil's attendance within a reasonable time lapse. The system's design took into account the relationship between the value of system's information in combatting truancy and the time taken for it to reach the attention of the pupil's parents and school authorities. For example, the data collected by the school's 200 teachers on the attendance of its 3500 students delivers comprehensive truancy information to the 10 assistant vice-principals within one hour of the period's roll call. Besides providing rapid data reduction facilities the system must also be very accurate. Dubious or incorrect truancy reports lead to most embarrassing confrontations between students, teachers, parents and school administrators. Such confrontations would quickly erode the credibility and usage of a system by all concerned.

For the student the system should be as inobtrusive as possible. It should also be accurate and flexible enough to allow a student to easily drop or transfer courses throughout the year without suffering from false accusations of truancy.

For the vice-principal and other school administrators the system produces a comprehensive and concise daily listings of absentees. In order to support the vice-principal's role within the school's functional hierarchy the system is flexible enough to allow the user to query the daily absentee listings. A query subsystem provides these users with precise information on the more serious truancy cases. For example, this option allows a truant officer to eliminate from the absentee listings students who have missed less than a certain number of lectures in a given course.

The system generates reports for the school's principal which provide school-wide attendance statistics. In this manner the system was able to demonstrate its utility in controlling the truancy problem.

To be acceptable to the office of the provincial minister of education the system had to work while being cost-effective. The latter objective was met by both a low operating cost and the requirement of only a modest capital investment in hardware. This system was designed for implementation around an inexpensive on-site hardware configuration with access to a remote computer. This configuration allows one to rent the use of the more expensive, infrequently used but nonetheless essential components of its design. Essentially the system's operation should not require employment of additional data processing personnel. This budgetary constraint was partially met by the development of two naive user interfaces; query and update sub-systems. These system interfaces allow various users to directly interact with the system through custom-taylored utility programs. Nevertheless some users such as various officials or the students themselves interact with the system through one of the school's clerks or secretaries. These clerks and secretaries were recruited from the ranks of the school's employees. As such the system's tasks were shared, where possible, among the school's employees. In this manner, the overall operating cost of the system was minimized. The reduction in the system's operating cost required that the system be designed around interactive utilities that were easy to use while being secure to the database's integrity. In the implementation of this cost-effective system particular attention was paid to the many problems of man-machine communication. Only by the careful design of interactive programs can one really optimize the system's utility to the user while minimizing the trouble and work required of the user in achieving his goal.

While the system reported in this thesis serves the needs of the six groups shown in figure I, only the school's teachers, truant officers and vice-principals have direct interactive access to the system.

Figure 2 demonstrates the functional interaction of these various groups using the system, while figure 4 schematically shows the spatial distribution of these groups within the school's environment.

Of the six other groups only three may interactively query the system's database for the immediate information they require for their daily duties.

The construction of the query sub-system designed for this purpose required a substantial outlay of time and expertise. The query sub-system is composed of utility routines which are the kernels of a naive user interface to one level

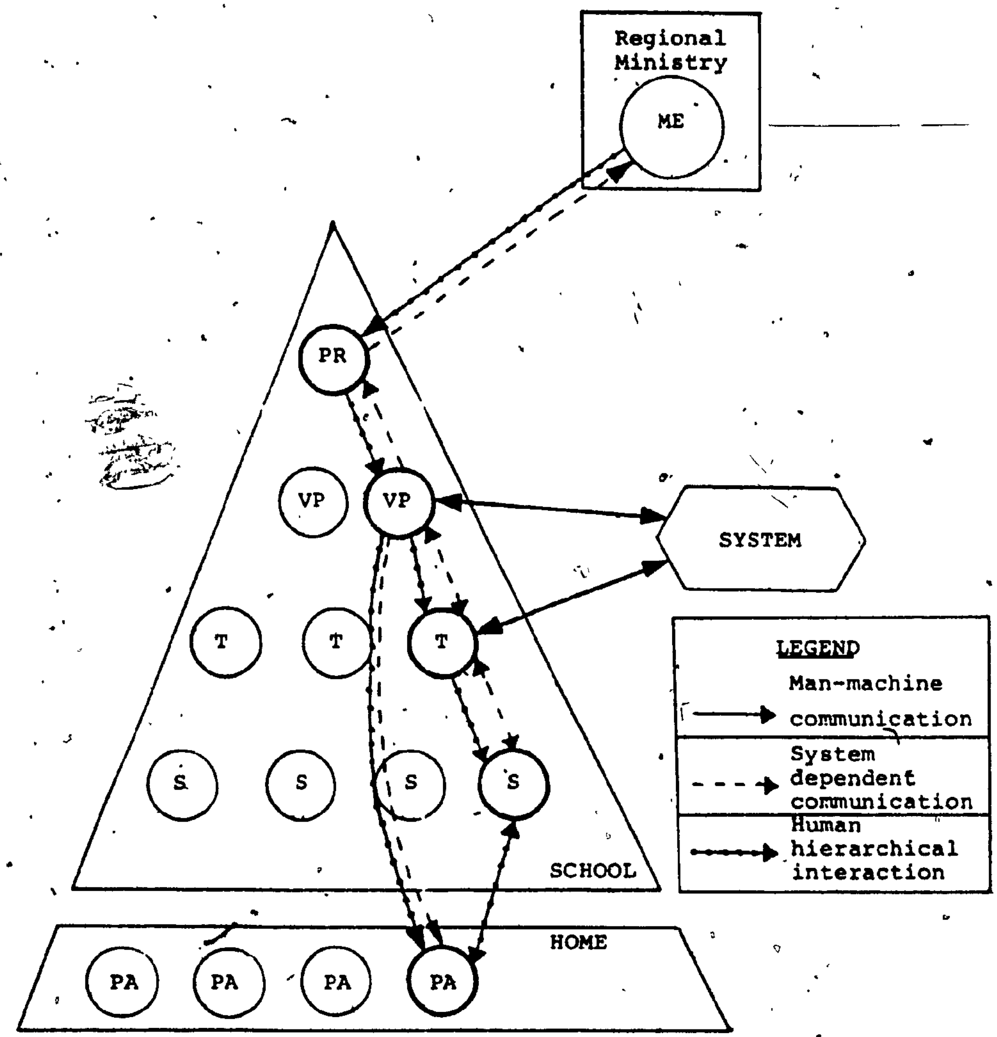


Figure 2 - The functional interaction between the various user groups using the system. The functional interaction is information dependent.

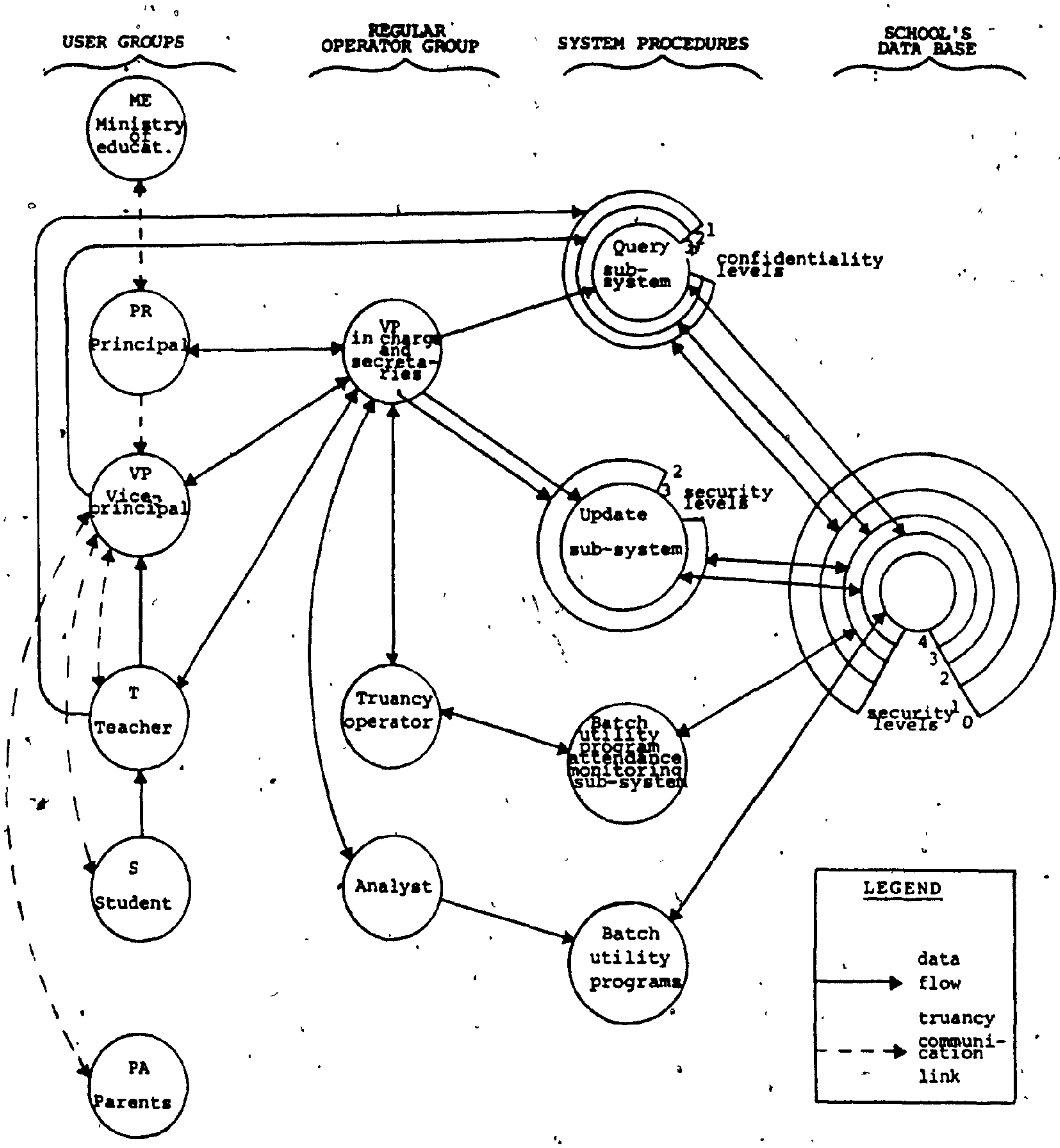


Figure 3 - The overall logical structure and interaction of this system with its users and the resulting meaningful communication channels established between users, thanks to system derived truancy information.

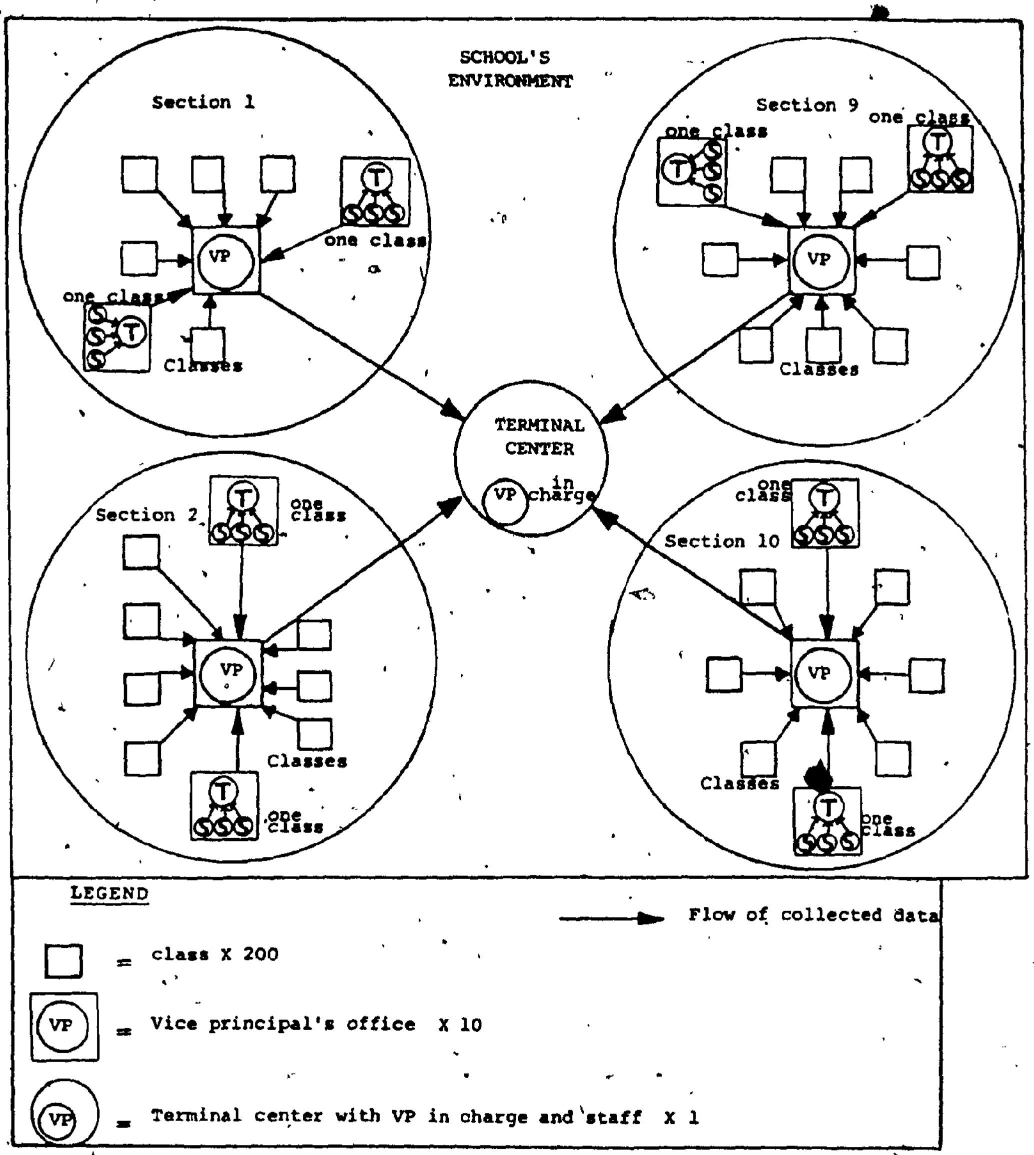


Figure 4-A - The spatial distribution of user groups within the school's environment and the flow of collected data.

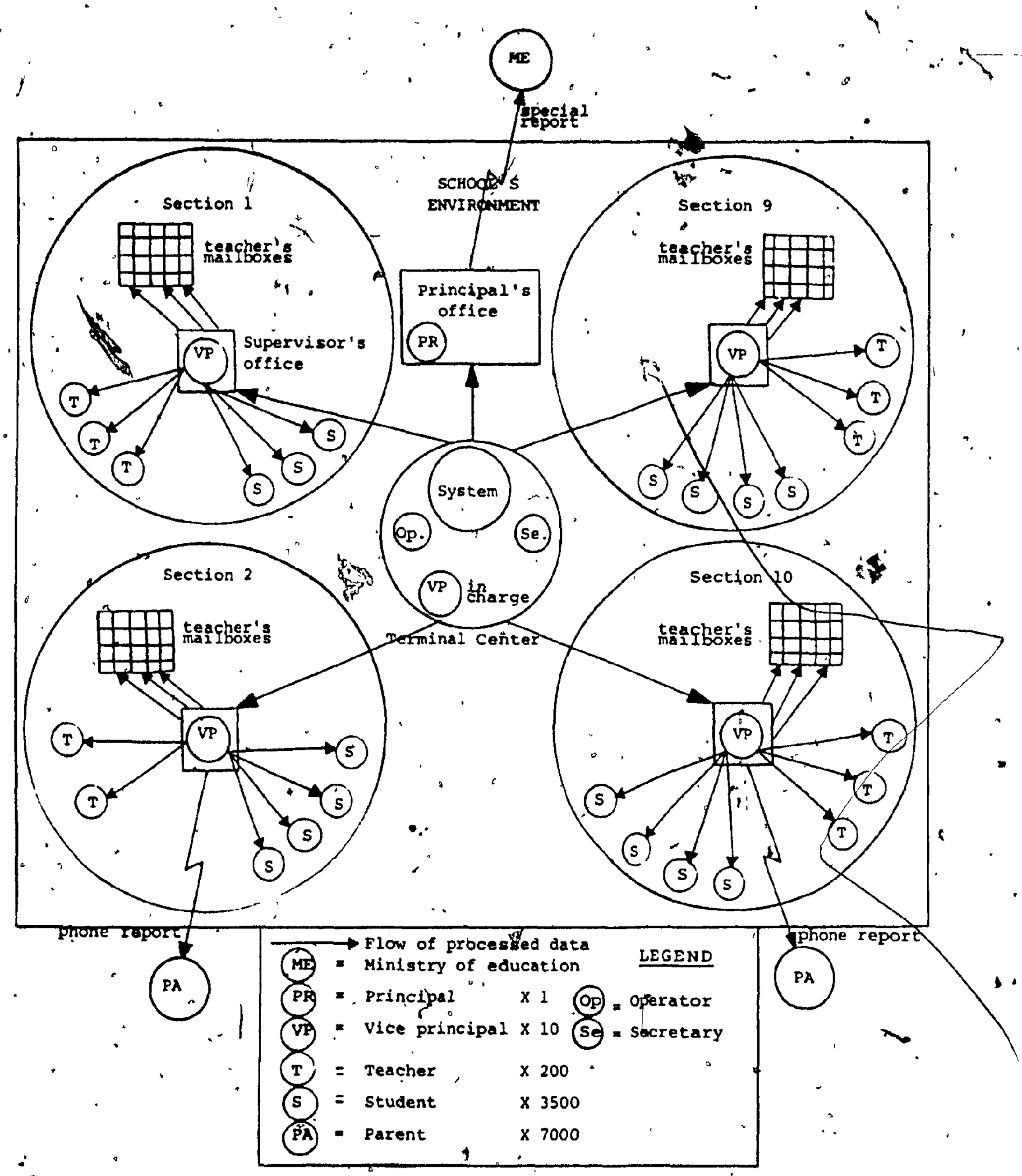


Figure 4-B - The spatial distribution of the different user groups within the schools environment and the flow of processed data.

of the system's operation. The design and operation of this interface is crucial to the system's overall acceptability and success. It is principally through this single interface that the school's immediate and daily needs are met.

The update sub-system contains a second naive user interface. This sub-system was designed for very limited interactive use by the school's clerks and secretaries. This interface to another level of the system was built to allow for the conversational interactive updating of the system's two master files. The utility programs constructed for this task were designed to permit easy editing of specific fields within these files while protecting the database's security and integrity. This second level of interactive communication with the system allows a secretary to easily change a student's course schedule. Such changes may be performed any time at his request without affecting the system's ability to accurately keep track of the student attendance throughout the term. The update sub-system also allowed teachers and administrators to easily reschedule courses as well as open or close course offerings throughout the year.

In addition to the two interactive naive user interfaces the system has batch utility routines which allow its operator to process batch job requests. All of the various reports and statistics as well as the daily updating of the system's master files are processed in batch mode.

Figure 3 schematically outlines the overall logical structure and interaction of this system with its users.

This logical structure is described in detail in chapter 3 of this thesis. The physical design of this system is discussed in chapter 4. Chapter 2 contains a brief account of the previous attempts to design and implement similar systems for use within the Quebec comprehensive high school system. The concept of a production cycle through which the attendance records are interactively updated is introduced in chapter 2. This concept facilitates the presentation and comparative evaluation of this system.

Chapter 5 summarises the system's performance and discusses its applications, shortcomings and possible improvements.

CHAPTER 2

PROBLEM ANALYSIS

In the mid sixties the concept of comprehensive high school was implemented in Quebec as a means of modernizing and improving the quality of secondary education. The comprehensive high school was also implemented to meet the pressing needs of a growing student population in a desecularized Quebec. The concept of the comprehensive secondary school (36) was modelled on similar high school systems that had been developed in the United States during the early sixties. Essentially these systems differ from the traditional secondary school in that they offer the student the ability to choose his own curriculum, through a flexible but full time schedule, from a wide variety of subjects and course offerings. In many ways their structure and operation resemble that of a university environment. To be cost effective these institutes had to have a large enough student population to ensure sufficient enrollment in its various course offerings. Unfortunately the very size of these schools made it difficult to ensure proper surveillance and academic guidance of its young student population. Furthermore the flexibility of the curriculum, with the student's ability to choose his own courses and students' mobility (56) in the school environment exacerbated the problem of monitoring the student's progress. Within the comprehensive high school system each student is promoted on the basis of his achievements in individual courses rather than on the traditional basis of his average academic performance for the year. This system posed many serious problems to both the teachers and administrators.

A Quebec government study was commissioned in 1972 to evaluate these problems (36). The study was prepared by a task force with the help of 200 specialists in 22 comprehensive high schools. Four of the participating schools were in the United States, fourteen were operational in Quebec itself and the remaining four schools were in Ontario.

Three principal results of this study may be concisely summarized for our purposes.

The task force noted that the comprehensive high school enrollment in Quebec varied from 1500 to 4000 students while that incurring in both Ontario and the United States ranged to a maximum of 2000 - 2500 students. The high enrollment within each school in the Quebec system exacerbated those problems that existed within the secondary school system that operated in the U.S. and Ontario. The very size of the school in the Quebec system produced new problems of its own. Unfortunately the size of the Quebec comprehensive high schools is principally established on the grounds of cost effectiveness. Consequently the administrative and academic problems that resulted from the size of the schools had to be solved without reducing enrollment.

The second point noted by the task force was that in all cases the management of the comprehensive school rested on the performance of the vice-principals. The number of vice-principals needed to manage a school varied from 3 to 12. This wide difference in the number of vice-principals needed to perform this task was attributed to both, the management style adopted by a school and its relative enrollment size.

The third point emphasized by the task force was that in the Quebec system, schools did not resort to use a home-room or a base for the management of its student population. In both the Ontario and American schools the home-room concept was introduced to effectively transfer the responsibility of monitoring each student's attendance and progress to his home-room teacher. The home-room teacher served also as a link between the administration and the student population. While the home-room concept had many shortcomings, it did offer the student, his parent and the schools administration the option to relate to a fixed frame of reference. The Quebec system which operated without the resort to this concept presented increased administrative difficulties because the link between the student and the school's administration was considerably weakened.

Considering these points the truancy problem within the Quebec comprehensive school system was inevitable. Truancy problems were also indigenous to the Ontario and American systems. The task force noted that the Quebec system reported a 4.4 - 15% truancy rate, while that observed in Ontario was 7 - 10%. Those from the U.S. comprehensive school system claimed to incurr a 10 - 20% absentee rate.

The task force attempted to assess the effects of the school's size on its truancy rate.

As we may see it in figures 5-A and 5-B, there is a significant correlation between the size of the school and its truancy rate. However the task force was unable to ascertain a reasonable cause for this observation. Among other results, the task force noted that truancy controls were very time consuming for school administrations, but they nevertheless form an essential element of school management to prevent the worsening of truancy related problems. Of the participating high schools, 67% felt that if truancy controls were abandoned class attendance would drop sharply; 28% of the high schools surveyed felt that failure rate and incidence of student drop-out would increase. Furthermore, 33% felt that this would effect the ability of the school's administration to adequately control the school environment. This report clearly demonstrates that attendance monitoring systems had to be an essential part of the Quebec Comprehensive High School System.

There are three essential phases, common to all of the truancy control systems considered in this thesis.

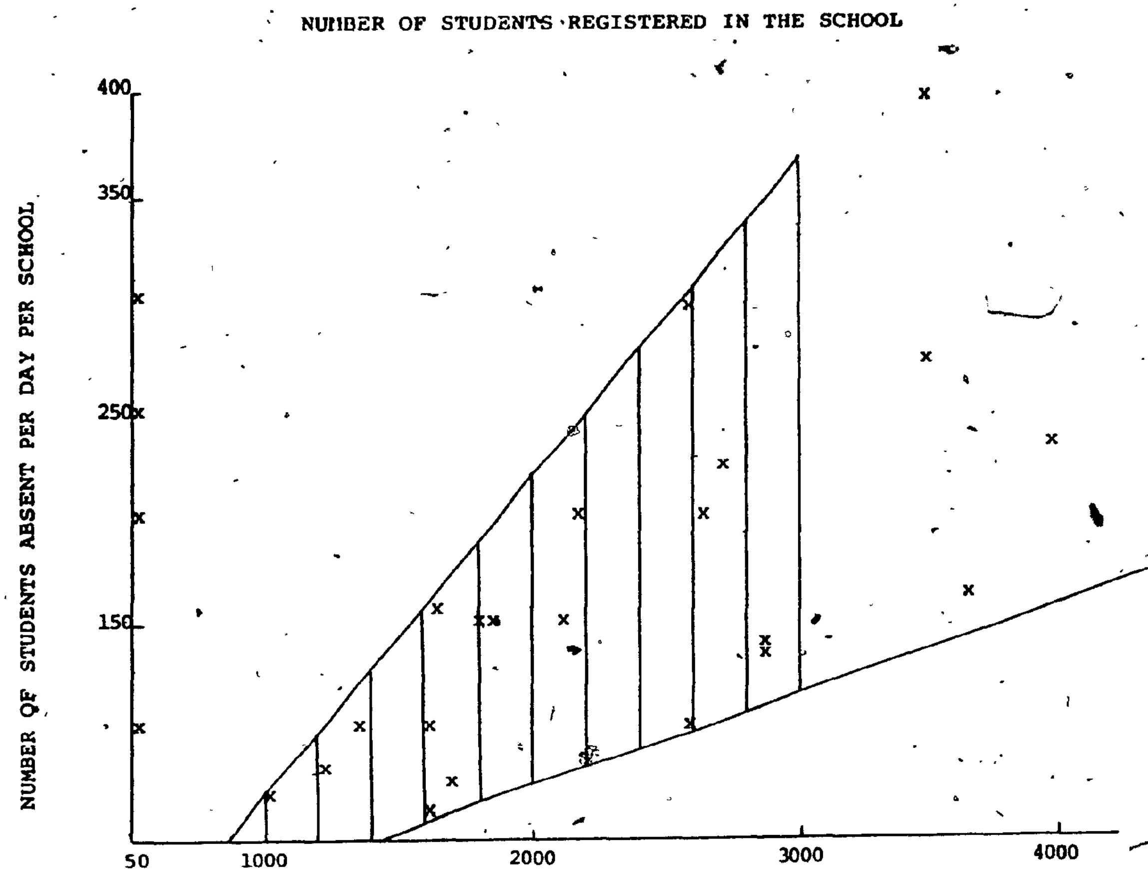


Figure 5-A - The number of students absent per day from school for the number of students registered.

COMPREHENSIVE HIGH SCHOOL	STUDENT POPULATION	TRUANCY RATE
Pierre-Dupuy	853	5.9%
Andre-Laurendeau	2 995	9.2%
Mgr. A.-M.-Parent	3 024	10.1%
Gerard-Filion	3 153	12.5%

Figure 5-B - A correlation between the size of the school and an increasing truancy problem for schools in the Chambly Regional School Board. (5)

The first phase consists of collecting relevant information about absent students from the teachers at useful and regular intervals. The second phase involves collating and updating the attendance records of each absent student. This phase involves both the file manipulation during the maintenance of the school's attendance database as well as a certain degree of data reduction.

The final phase involves the review and analysis of these files by authorized personnel. This phase requires that the files be quickly accessible and structured to easily provide the information required by the various users of the system. These three phases form a production cycle through which every attendance monitoring system iterates as shown in figure 6.

The implementation of this production cycle within any given attendance monitoring system depends on five major design factors (figure 7). The availability of human and technical resources is the first and most important of these factors. The next consideration in the design of such systems is the expected volume and rate of student absentism. The third factor is the depth of the data reduction and information processing required to meet the need of the system users. The fourth factor in the implementation of such systems involves evaluating the acceptable response times for the system to meet the various queries of its users. The time delays required to process truancy information diminishes its value to educators and administrators alike. Each

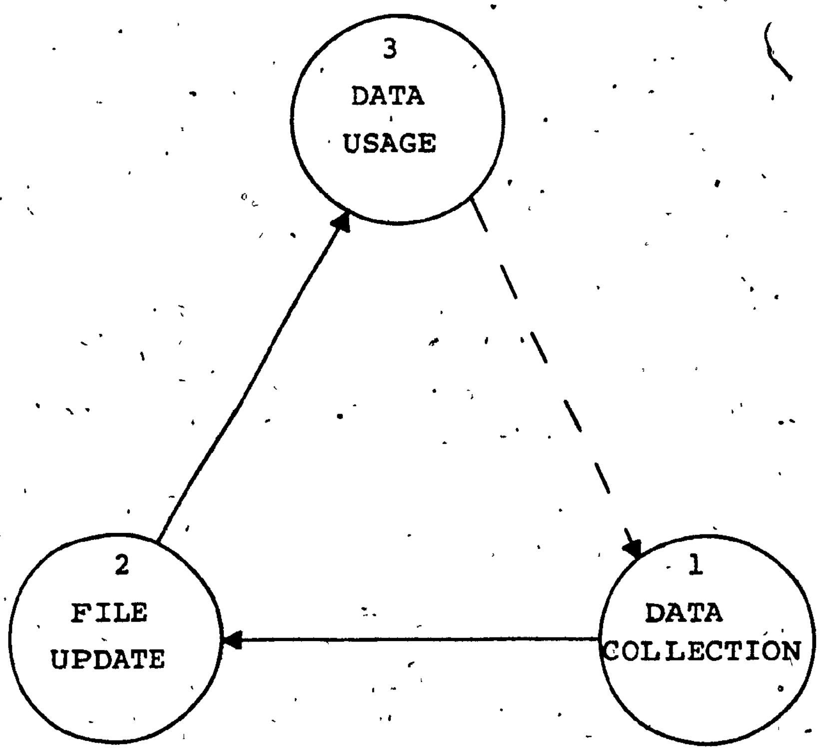


Figure 6 - Each production cycle has three phases

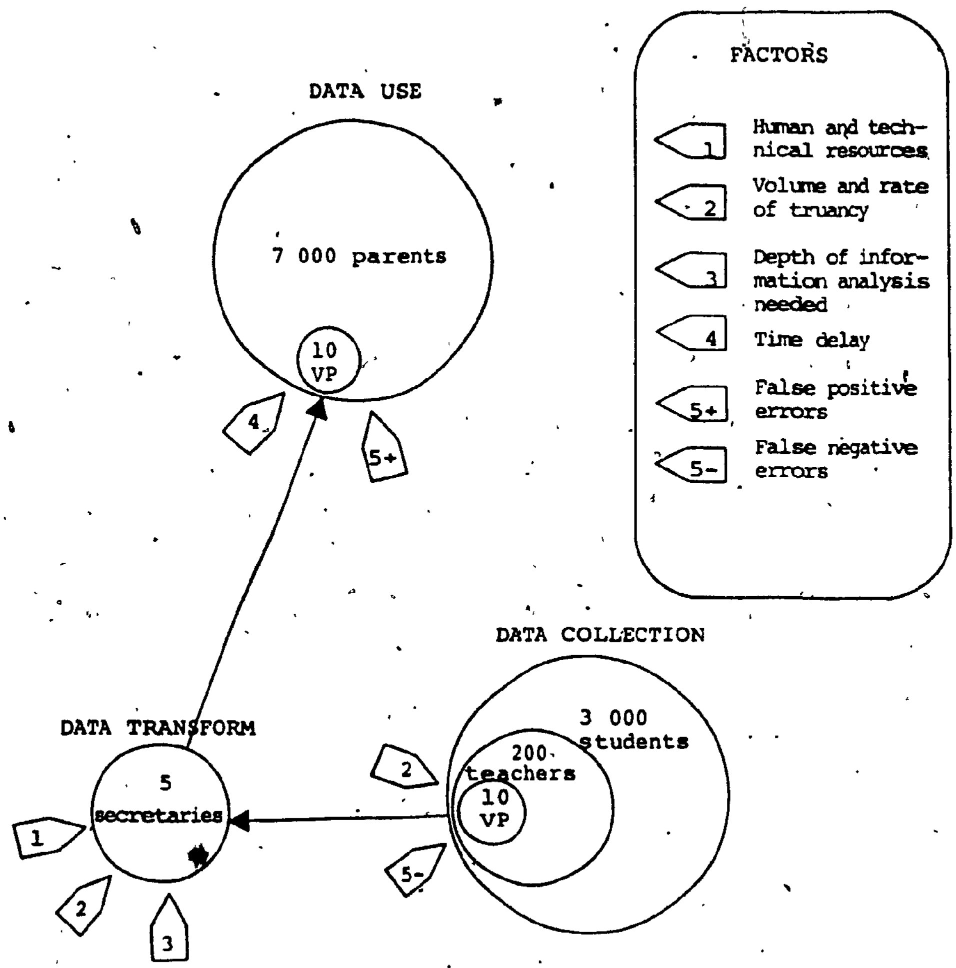


Figure 7 - The factors that affect the production cycle efficiency.

group of users within the school may need the information they require from the system over a range of acceptable time delays. While absolute response times required by the users may vary they share common characteristics. The system's truancy information is of greatest value for the user within a short time of its acquisition. Subsequently the value of the system's information decays in the sigmoidal manner schematically outlined in figure 8.

The fifth and last factor involves establishing the degree and type of errors that are acceptable to the systems users. Essentially there are two types of error that can occur within such systems. The first of these is a false-positive error. This error occurs when a student is incorrectly reported and accused of being truant. This type of error has very serious and embarrassing consequences for the users of the system. False positive errors totally erode the system's credibility and use. The second type of error is a false-negative one. Such errors occur when a student's truancy escapes notice. Within bounds the consequences of this type of error are far less serious.

In order to avoid false-positive errors a lot of redundant information processing is required to verify and pinpoint the occurence of actual errors or malicious intents. In order to avoid false-negative errors one must politely inform a responsable pedagogical supervisor of each teacher's reliability in taking and reporting attendance. This last factor has many far reaching ramifications for the system's intended use as well as the quality of the teaching environment. This fifth factor requires very careful consideration in that it encompasses the most delicate task underlying the acceptability of any attendance monitoring system.

These five factors are essential in that they determine the relative partitioning of work involved in daily operation of the system. They also effectively determine the degree of automation that can or should be undertaken into its implementation. Without adequate consideration of these factors a given system often requires too much manual labor and burocratic cooperation to provide a degree of data processing which is sufficient to meet the user's needs.

In the worst case it can infringe on privacy and precipitate open confrontation between the school's teachers and its administration, as well as create tensions between parents, teachers, students and the school administration.

In order to analyse and compare different systems, it is necessary to further elaborate on the production cycle introduced in figure 6. These comparaisons may be simplified by considering a functional decomposition of this production cycle into the five basic processes outlined in figure 9. These basic processes are data generation, data collection, data transform, data distribution and data usage. A careful analysis of these phases provides valuable insight into the cost-effectiveness of any truancy monitoring system.

Data generation is the process by which teachers identify absentee students from the class list. This process

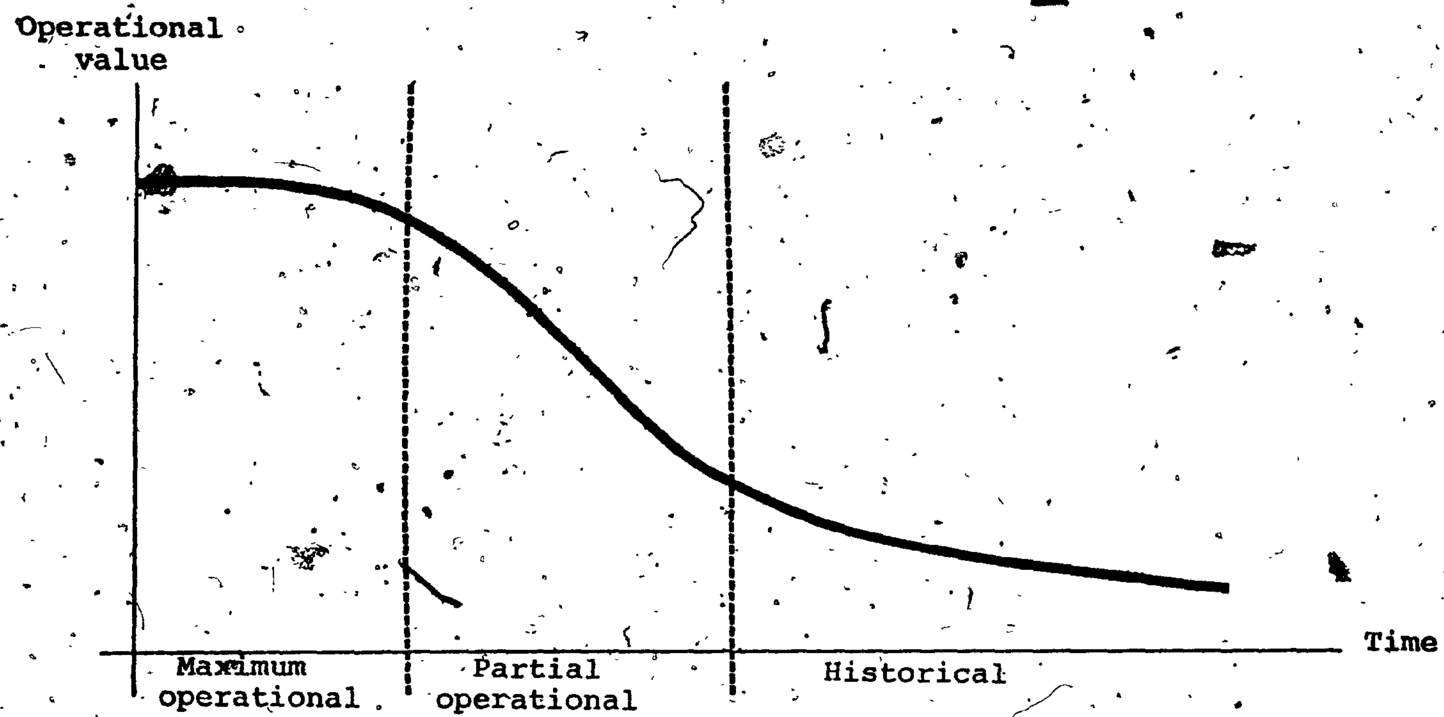


Figure 8 - Operational value of information.

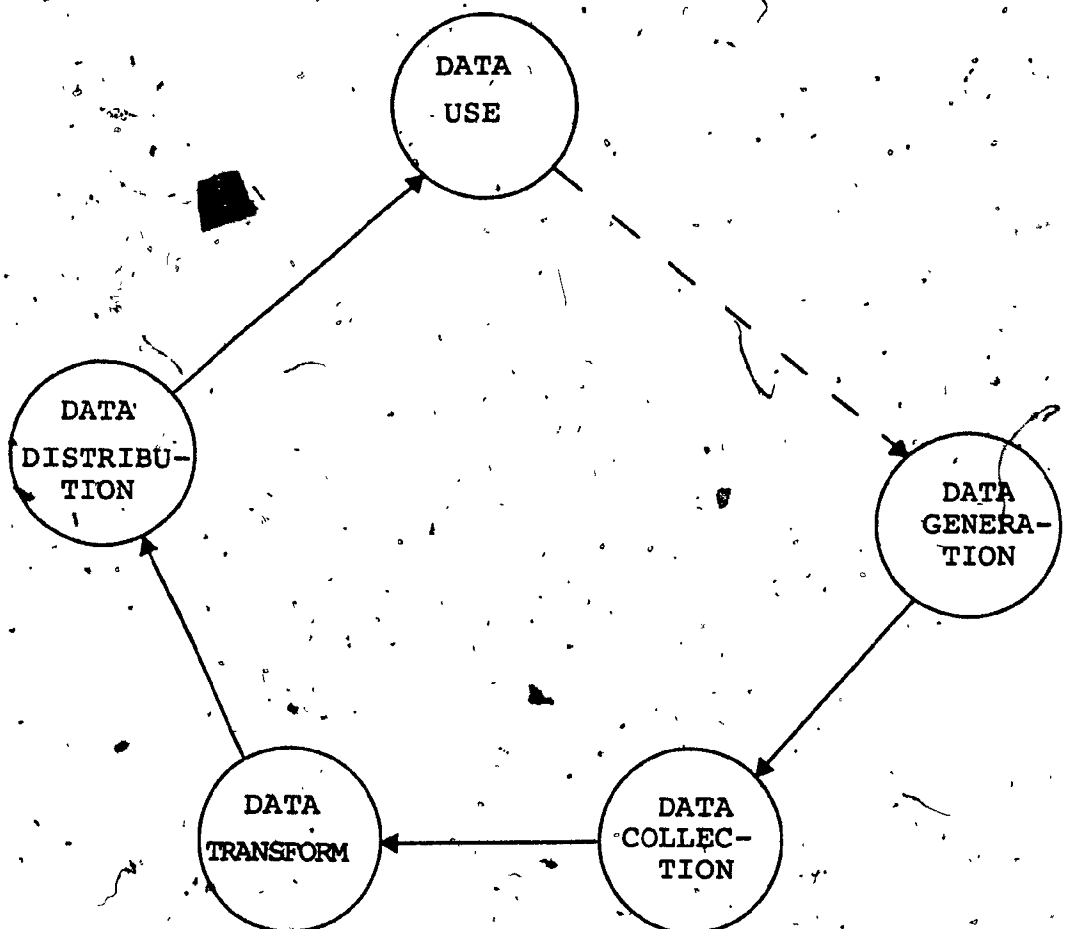


Figure 9 - Production cycle broken into five functionally distinct phases.

has been implemented in many ways throughout the history of the Quebec systems. Evaluation of the various methods for identifying absentee students involves consideration of time and effort required for a teacher to complete this task. The overall accuracy and complexity of this chore often established a given implementation's acceptability.

Data collection is the process by which this truancy data is collected and transmitted to a central site for processing. The critical evaluation of any implementation of this task must consider the time and accurary of this process in addition to the actual resources required to effectively manage the value of the collected data.

Data transform is the process by which the truancy data is acquired, collated and stored by the system. This process also involves the synthesis of comprehensive reports from truancy data and files. An evaluation of any given implementation of this process must consider both the data reduction and computation ~~required~~ for this task. The time and human resources required to meet the need of the various users of the system must also be considered.

Data distribution is the process by which the information provided by the system is channelled towards the decision-making users of the system. Evaluation is based on time and human resources needed for this task.

Data utilisation is the process where the degree of integration of the provided information to the actual decision making policies is established. Evaluation of data utilisation for a given impl-mentation is a difficult task. One way of approaching this problem is to monitor the rate of use of the system's report facilities. It is also possible to compare the value of the information provided by the system with that anticipated by the user.

By using the paradim of a production cycle it is possible to evaluate a given system's ability to provide comprehensive information in an environment in which there is a high volume of data transfer and where resources are highly constrained.

Between 1969 and 1974, a series of automated attendance monitoring systems were implemented by the Quebec Ministry of Education in several comprehensive high schools. All four of these implementations utilized a batch processing mode. (28) These systems differed from each other principally in the data generation techniques employed in their implementation.

A concise description of the production cycle of these systems is as follows:

1. Collect from teachers class truancy data for every lecture throughout the day.
2. Collate this data and once a week send it to a remote site.
3. Process this data to synthesize a weekly truancy report on each student.
4. Mail a version of this weekly report to the student's parents.
5. Synthesize a monthly report or attendance directory for the schools administration.

Each of the four batch systems developed during this period failed to provide an effective means for combatting the school's truancy problems. These batch systems were also labor-intense, costly and prone to high false positive error rates (13 %). In term of the five phase production cycle, outlined in figure 9, it is possible to briefly describe the first batch system as follows:

1. Data generation

Each teacher produced a hand written list of absentee students. Figure 10 shows a copy of the form developed for the teachers to record truancy transaction data for each of his/her class.

2. Data collection

The completed forms were collected and sorted daily at a central location in the school complex. At the end of the week, the sorted form deck was sent to a remote computer center for keypunching and processing. A typical deck contained over 5000 forms specifying over 15000 transaction records for the week.

3. Data transform

The truancy data was keypunched and stored on a magnetic tape until a batch run was scheduled. Sample outputs from this data transform process implemented in this phase of the system are given in figures 11 and 12. For a school with an enrollement of 3000 students, the weekly batch run produced 1500 pages of listings (figure 11). The monthly run generated a 1500 page attendance directory (figure 12).

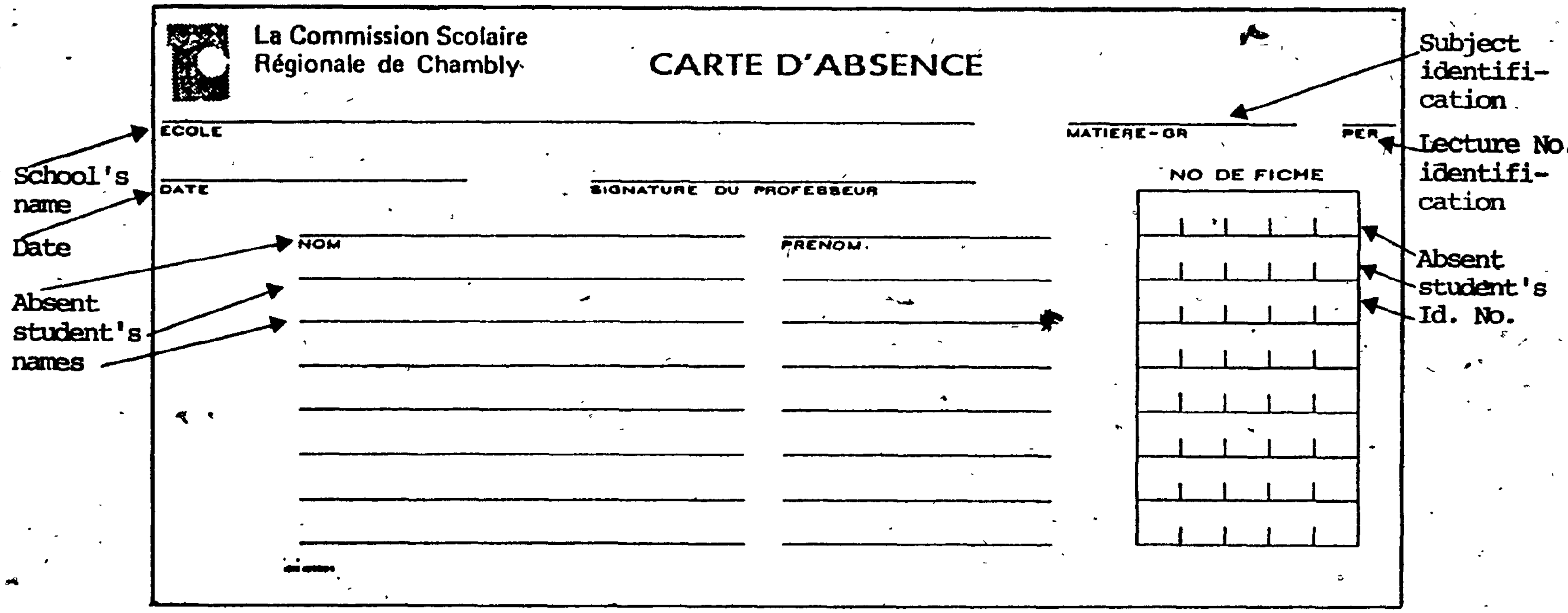


Figure 10 - Class absentee registration form.

Class attendance was recorded in handwritten cursive script at the data generation phase of the first batch truancy system implemented in Quebec. Class attendance was recorded in this fashion for every lecture of the day. This technique required 3-8 minutes of the teacher's time per lecture and demonstrated a high error rate.

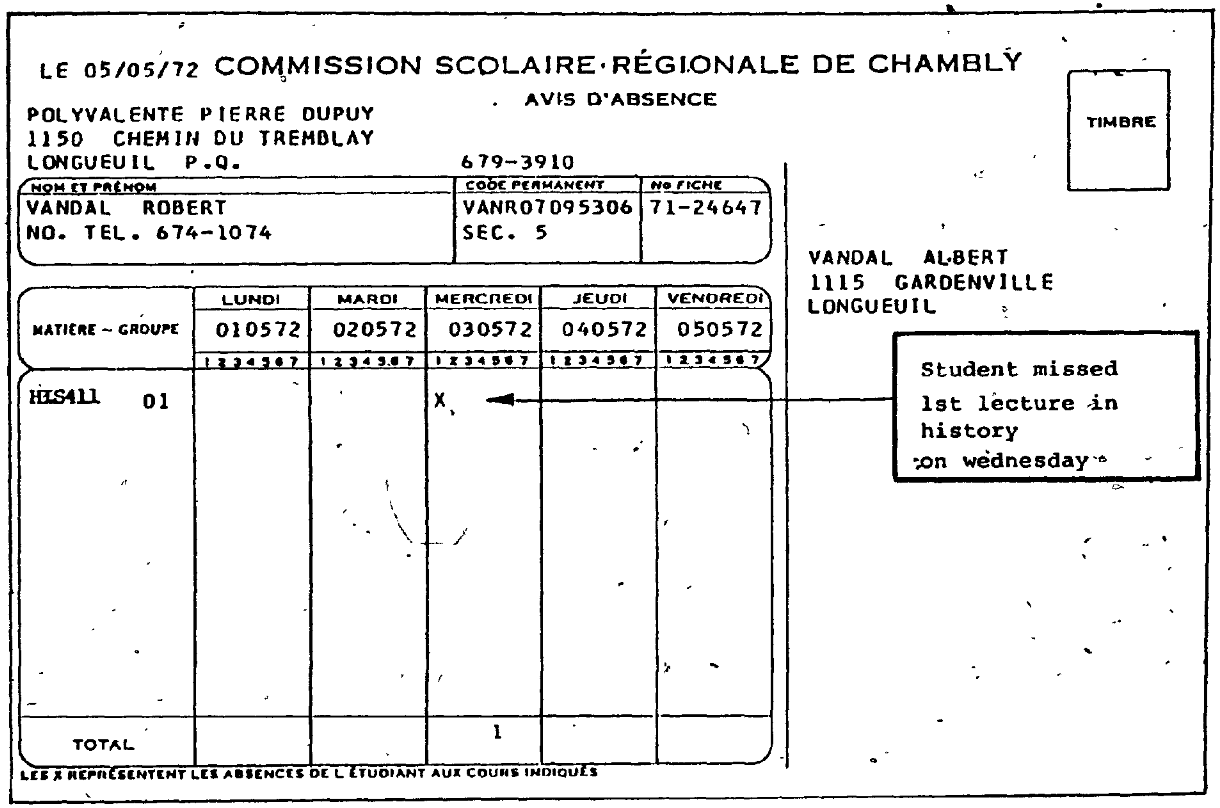


Figure 11 - This weekly letter to parents contain short, overall truancy information about the student. In system #1, #2, #3, all parents received it. Such letters in system #4 were mailed only to parents of truant students.

NOM	PRÉNOM	CODE PERMANENT	NO DE FICHE	ÉCOLE
VALADE	SYLVIE	VALS24605700	71-17989	POLYVALENTE PIERRE DUPUY

CUMUL TOTAL	MATIÈRE – GROUPE	CUMUL SEM 1	S E P. 1234567	O C T. 1234567	N O V. 1234567	D E C. 1234567	J A N. 1234567	MATIÈRE – GROUPE	CUMUL SEM 2	F E V. 1234567	M A R. 1234567
3	ANG321 01		0	0	1	0	2	ANG321 01		0	0
1	ANG411 01		0	0	0	0	0	ANG411 01		0	0
4	BIO411 02		0	0	2	0	2	BIO411 02		0	0
3	CHI425 03		0	0	1	0	2	CHI425 03		0	0
3	FRA321 03		0	0	2	0	1	FRA321 03		0	0
3	MAT321 02		0	0	1	0	2	MAT321 02		0	0
4	SCR311 02		0	2	0	0	1	SCR311 02		0	0
21	TOTAL		0	2	7	0	10	TOTAL		0	0

Figure 12 - A sample listing from a page in the school's attendance directory. A single copy of this directory was produced on a monthly basis, for use by the administrators and teachers of each comprehensive high school.
The directory contained information on the overall monthly attendance records of every student in each of his courses.

4. Data distribution

The weekly computer generated attendance listing for every student was mailed to his/her parents. The attendance directory which contained the monthly statistics for each student in alphabetic order was kept at a central office for consultation by the system's users.

5. Data usage

The system placed the responsability of combatting a (22) student's truancy directly on his/her parents. It however (13) failed to provide parents with either current or accurate attendance information. The weekly reports which were mailed to parents provided crude and sometimes erroneous statistics that were often three to four weeks old. Such statistics proved to be of very little value in combatting truancy. This system was also labour-intense and costly to implement.

The second version of this attendance monitoring system arose out of an attempt to cut data processing costs and delays. The keypunching phase of the first system was replaced by using optical mark sense equipment to directly read attendance data from special forms shown in figure 13. The data generation phase was essentially the same except that the teachers recorded class attendance directly on mark sense forms. This system did demonstrate a reduction in both data processing costs and delays. However the system's error rate skyrocketed as the result of faulty pencil marks on the mark sense forms. The teachers also found the work involved in taking attendance with these forms to be totally unacceptable

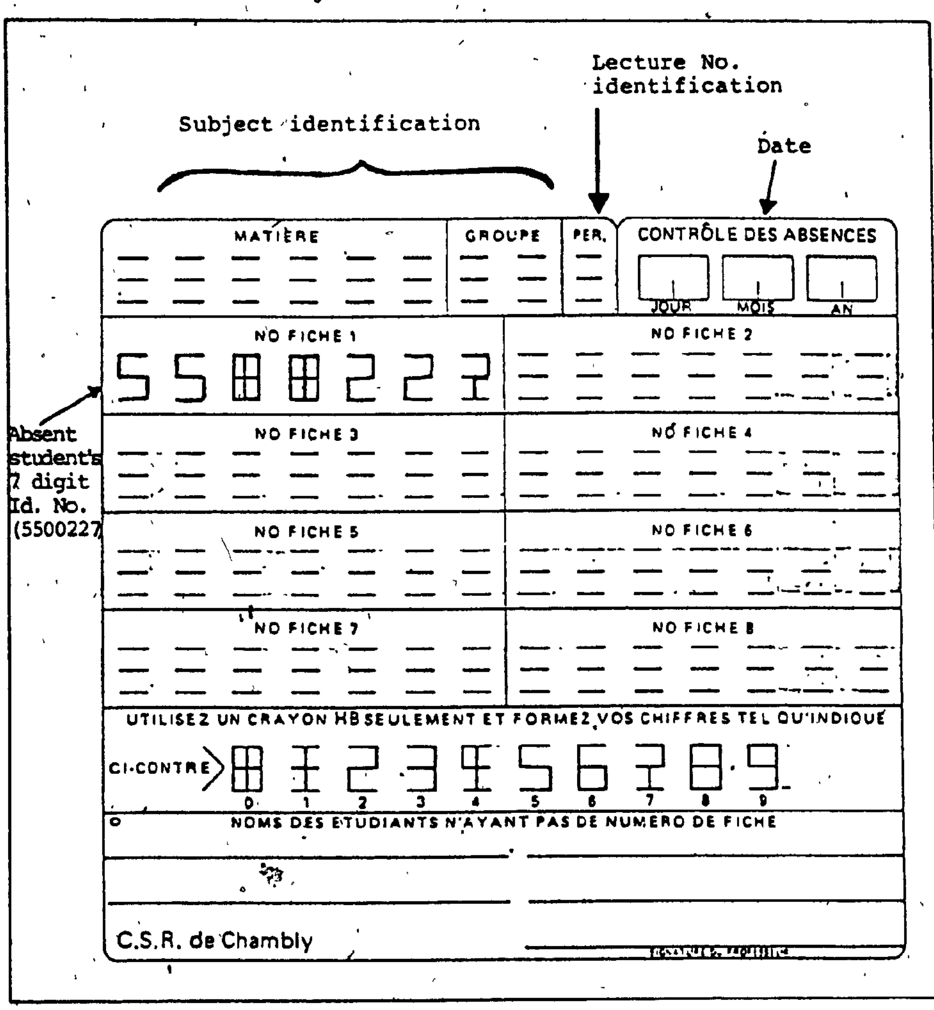


Figure 13 - This data generation technique for the second batch system was a plain mark-sense form that had to be filled out carefully. The teacher needed 5-15 minutes per lecture to take the class attendance. This method was prone to high error rate.

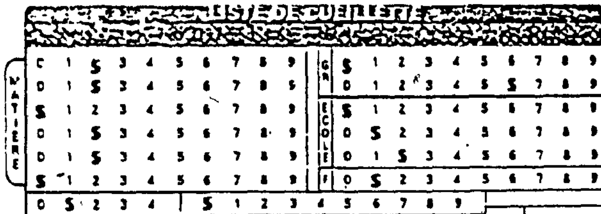
on a routine basis, and a serious labor problem arose.

The third version of these batch systems attempted to diminish the teacher's workload in recording attendance at the data generation phase. Pre-formatted mark sense forms for each class like that shown in figure 14 were developed for this purpose. These preidentified forms were used to record a student's daily attendance in class over a five day period. This technique reduced both the labor and error rate involved in the data generation phase of the second batch system. However its error rate was still 10 - 20% because it was difficult to maintain these preidentified classlist forms up to date among a high volume of never-ending course changes.

The fourth system attempted to cut down the data processing costs incurred by earlier systems. As such, attendance reports were printed and mailed only to those parents whose children had truancy problems. This apparently clever approach quickly backfired. The truant student was often able to intercept the household's mail before his parents got home. This fourth and most sophisticated of these systems still exhibited a 10 - 20 % error rate.

It had a two-three week production cycle, demonstrated poor information utilisation and a low credibility.

Thus working experience with these four systems led to the formal recommendation in 1974 (5) that the use of automated attendance monitoring systems be discontinued in the environment of the Quebec comprehensive high school.



RÉGIONALE DE CHAMBLY

ECOLE		DATE	PAGE
POLY.GERARD-	012	03/12/74	242
MATIERE	GROUPE	DESCRIPTION	
220220	06	PHYSIQUE	
PROFESSEUR			NO SEMAINE
HORVATH A		1	10

NOM & PRÉNOM	CUM.	NO. TEL.	09/12 Lundi	10/12 Mardi	11/12 Mercredi	12/12 Jeudi	13/12 Vendredi
BEAUDOIN CAROLE	0	670-1031	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
BELANGER SYLVAIN	0	670-2741	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
BLANCHETTE RICHARD	0	674-7900	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
BOUDREAU GHISLAIN	0	679-0141	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
BROSSEAU DANIEL	0	651-2981	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
CHARRON DENIS	002	674-8979	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
CHAPUT MONIQUE	0	670-9972	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
CHRETIEN MANON	0	670-9258	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
COMEAU CAROLE	002	679-4675	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
COMTE NORMAND	0	677-1025	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
COMETE SERGE	0	679-7696	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
COSQUER MONIQUE	002	670-0054	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
DUFRESNE MANON	0	674-7430	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
DUMONT CAROLE	0	677-3525	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
FRENETTE MICHEL	002	670-6075	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
GAUDREAULT ANDRE	0	670-3779	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
GAUTHIER DANIELLE	002	679-2192	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
GENEREUX GINETTE	0	677-3377	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
GRAY ALLEN	0	674-9675	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
LABBE SYLVIE	0	670-8981	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
LAPRISE CHANTAL	0	674-0177	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
LAVALLEE LYNDA	002	670-1742	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
LEFEBVRE GUYLAINE	0	674-9535	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
LEMIRE ALAIN	002	872-5720	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
LEPINE ROXANE	0	679-3961	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
MALLETTE BENOIT	002	677-6288	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
MICKEL MIREILLE	0	651-3721	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
O MEARA MAUREEN	0	670-2887	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
PROVOST CLAUDETTE	0	679-8911	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
PROVOST FRANCIS	002	677-4802	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
RIVEST MICHEL	002	677-3484	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
SCHILLER MARIO	0	679-1145	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
TREMBLAY SYLVIE	002	651-0071	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
VANIER LINDA	0	670-3199	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4
			1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4	1 2 3 4

Figure 14 - The data generation technique employed in the 3th batch system was an optical mark sense form that was preprinted for the teacher's use. This form contained a class list matrix suitable for recording the attendance of each student in the class throughout the week's lectures. To specify the absence of a given student (x), during tuesday's lecture, the teacher made a mark in tuesdays' column (x). While this technique provided for very quick and accurate data generation, its error rate was still 10-20% due to the difficulty of maintaining an up-to-date class list.

It was in an aftermath of disillusionment with automated attendance monitoring systems that the system described in the next chapter was developed.

CHAPTER 3

LOGICAL DESCRIPTION OF THE SYSTEM

Chapter 2 presented a brief account of the various unsuccessful attempts at implementing a satisfactory computorized attendance monitoring system within the Quebec Comprehensive School System. A careful analysis of these system's failures is essential to understanding of the logical design of this system. The problems which lead to the disillusionment and eventual rejection of early attempts at automated systems had to be resolved. (31)

The single most important requirement was that (26) the decision makers within the school system wanted to (42) establish quick and accurate cooperation with the parents of truant students. In order to establish such a rapport the decision makers needed truancy reports within an hour of the roll call.* These considerations determined the period of the production cycle. This factor in turn determined that the system's processing capability was to be implemented via a dedicated computer or an on line terminal located somewhere in the school.

The second requirement was the rate and volume of the transaction records. Routine range was 100 to 250 transactions per lecture period. However in exceptional cases

---

* In routine operations the administration was content to handle truancy reports which were one day old.

truancy levels rose to a peak of 1000 transaction file records per lecture period. The volume and frequency of this processing and other constraints required a very efficient data generation and collection technique. These will be described in detail in chapter 4.

The third requirement was that the system had to be implemented without additional staff. These budgetary constraints demanded that the duties of operating the system would have to be distributed over existing personnel. The system was constructed to provide a number of functionally and hierarchially interrelated components.(12) These would have to coincide with the responsibilities of the school system's administration. This factor determined the degree of system sophistication and the interaction required in the design of its subsystems. Each subsystem had to operate in an apparently self-contained, easy to use and transparent manner. The subsystems were designed to be directly accessed in an interactive mode by the user. Most of the future system's users had been frustrated by the inadequacies of the many previous attempts at implementing such a system. Furthermore most of these users had never directly interacted with any of those systems. Consequently it was imperative to offer a suitable control and command language for each subsystem. These languages would permit interactive, meaningful and direct communication between naive user and the computer.

The fourth requirement was the depth of the needed (17) data analysis. The system had to provide statistics and extensive data on each pupil over the entire school year.

The degree and type of admissible error in the system constituted a fifth requirement. A false negative error rate of 5 to 10% would not seriously affect the management and the decision making process involved in truancy management. However the upper limit of the false positive error was 0.1%. The system's credibility would be lost should it ever surpass this false positive error threshold. The acceptable bounds of error influence the choice of data transfer techniques, database design and security measures. The query update subsystem was organized to provide secure access to the database.

An on-line system (11) was used to implement the rapid production cycle required by the school. (48) Random access files were used to permit interactive updating of the system's database. A pseudo-batch phase extended the on-line system's capability to generate and process large transaction files efficiently. The query subsystem provided the user with an interactive facility for the detailed analysis and retrieval of any student's record. In accordance with the very diverse requirements of the users, the system's

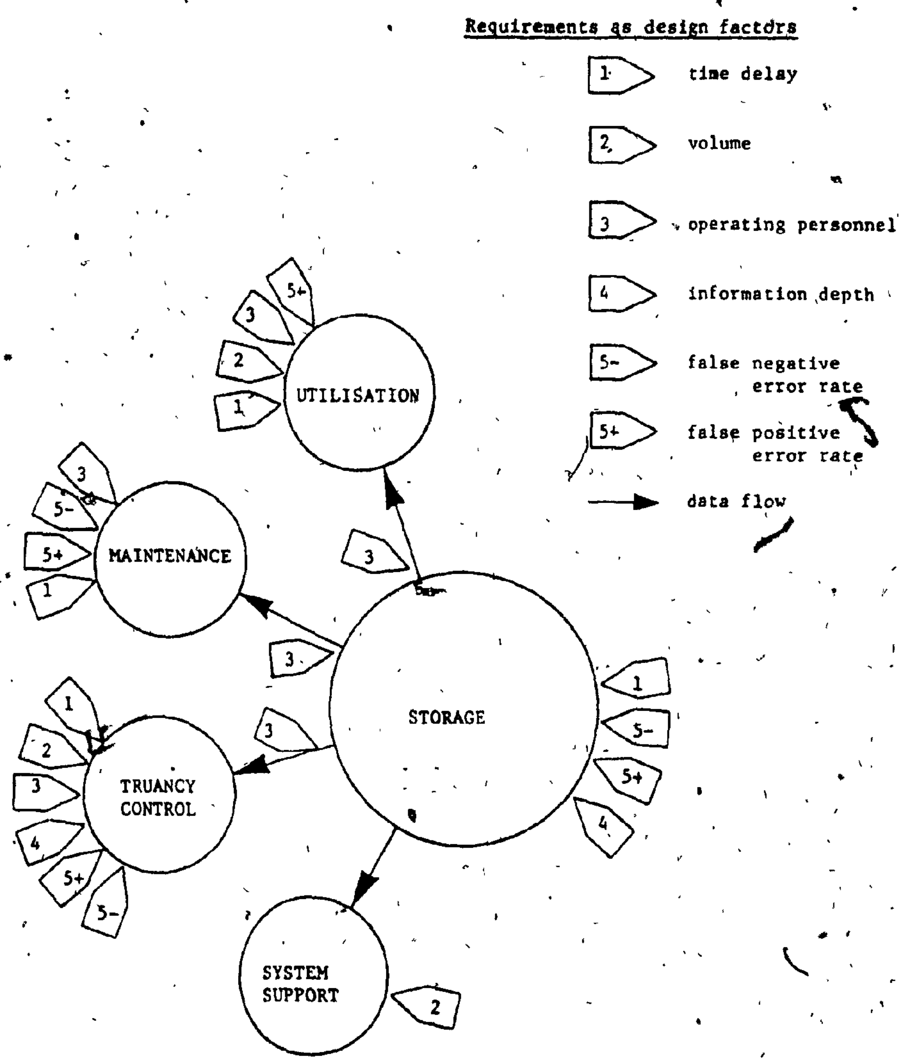


Figure 15-A - The impact of design factors on the entire attendance monitoring information system.

The system is functionally split up into sub-systems in relation to the indicated design factors

Storage = on line database sub-system
Maintenance = update sub-system
Utilisation = query sub-system
Truancy control = attendance monitoring sub-system (production cycle)
System support = batch utility sub-system

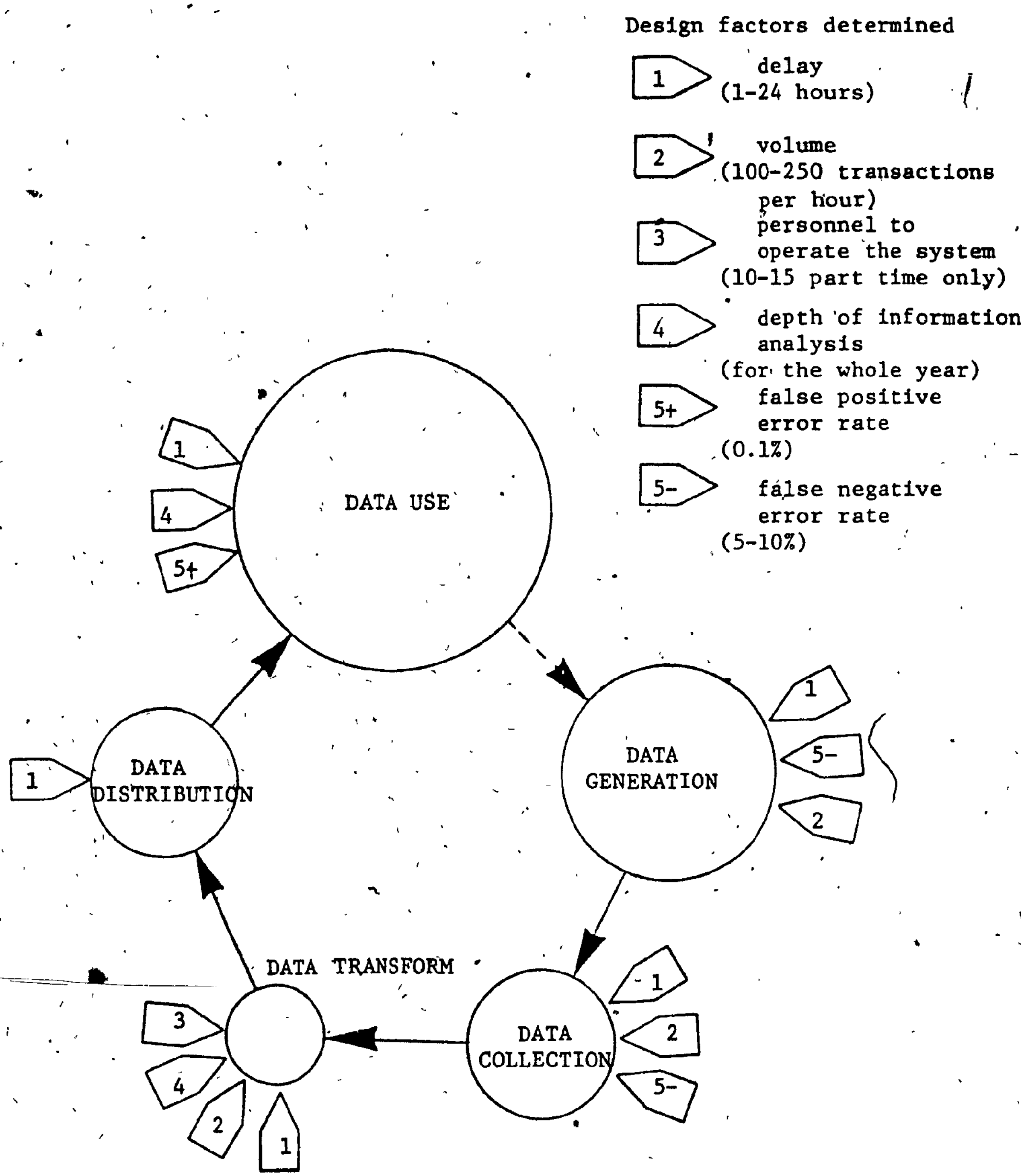


Figure 15-B - The impact of design factors on the pseudo-batch attendance monitoring sub-system.

This diagramm shows how the design factors shape and affect this sub-system throughout its production cycle. Each step of the production cycle is an independent component of the sub-system.

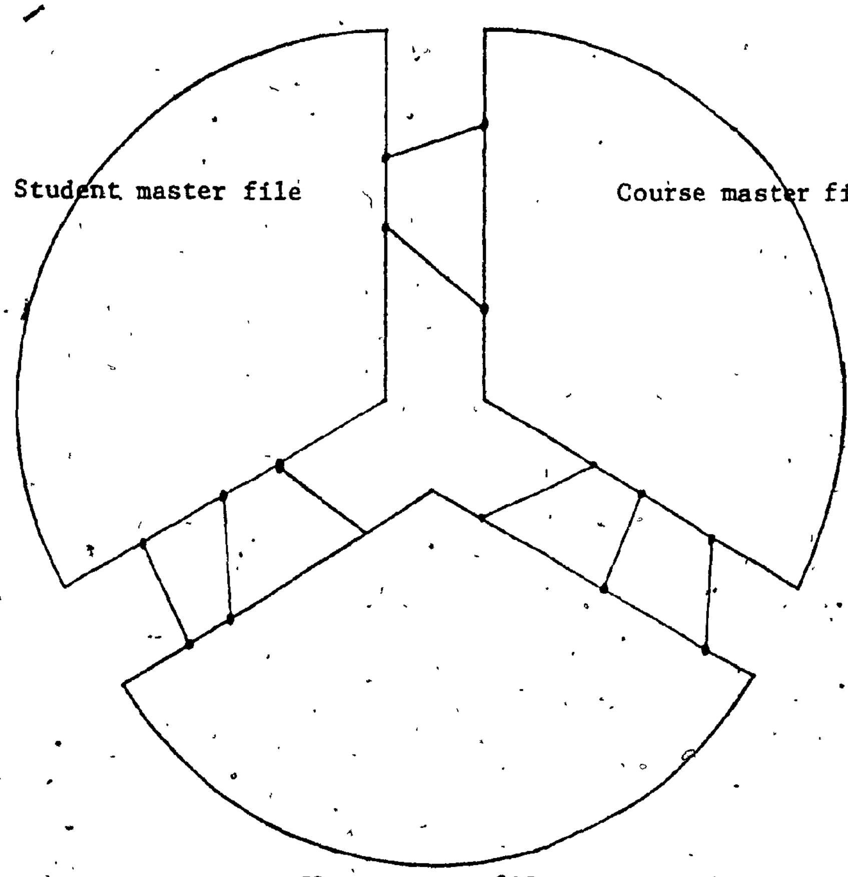


Figure 15-C - The data-base of the attendance monitoring information system.

The data base consists of three separate random access files linked to each other.
This data organization is dependent on access organization. Access organization is dependent mainly on the number of accesses, security, error protection (malicious or accidental) and lastly on CPU costs.

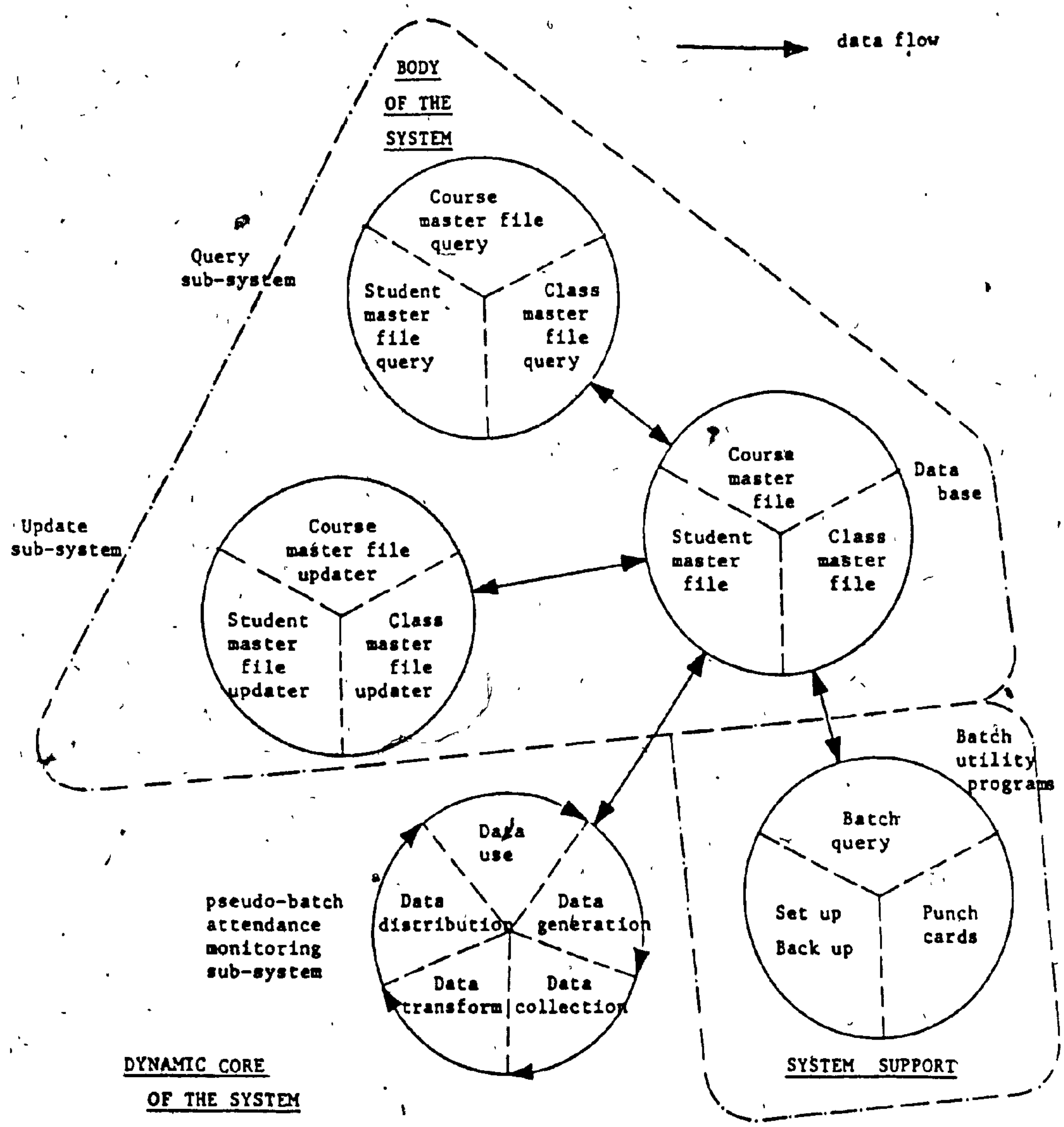
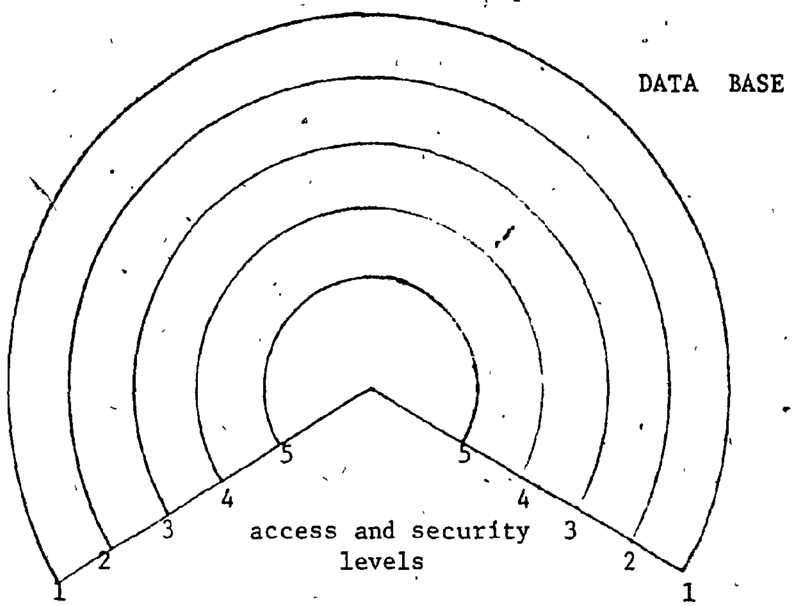


Figure 16 - Data flow within the attendance monitoring information system.

User access organization required the further splitting up of each sub-systems into operationally independent components.



Logical access and security level definition to the data base:

1 = Access to one record of the student master file limited only to validated fields.
Authorized user: student and / or his parents
(Not implemented though)

2 = Access to a set of records of the student master file and / or the class master file limited only to validated fields.
Authorized user: teachers, vice-principals.

3 = Access to all records of the student master file and the class master file limited only to validated fields.
Authorized user: record's office.

4 = Access to all records of the student master file, the class master file and the course master file limited only to validated fields.
Authorized user: vice-principal in charge and his staff.

5 = Access to all records of all files to all validated or not validated fields.
Authorized user: the analyst.

Figure 17 - Data base access organization definition.

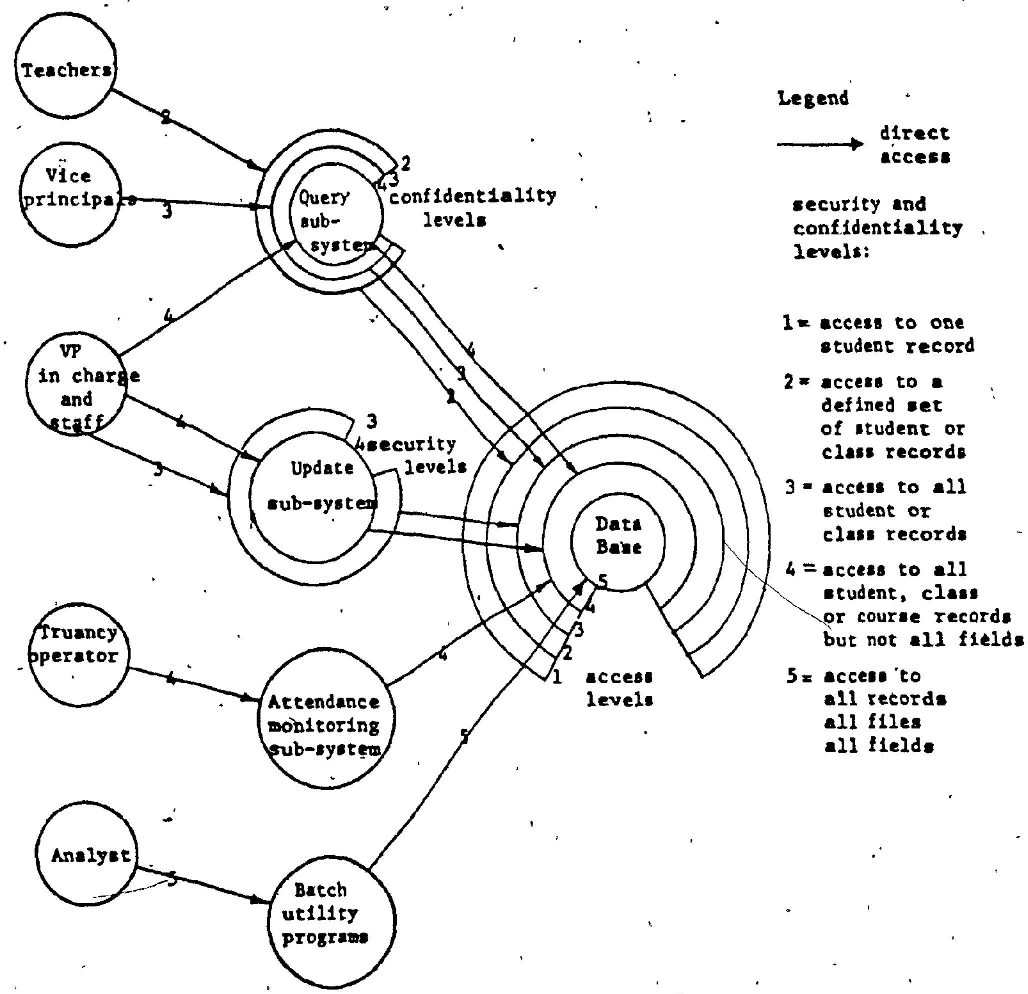
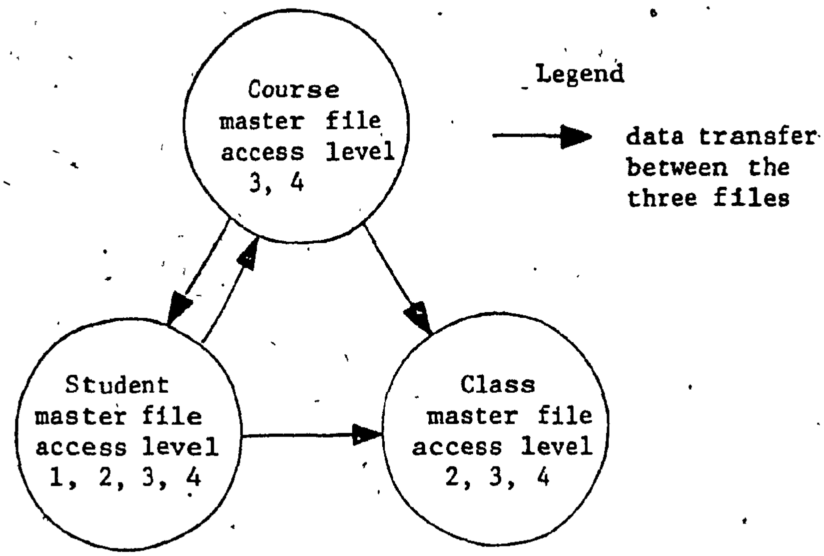


Figure 18 - Hierarchial structure and functionnal description of the system with its access organization based on security.

Access levels are the same for the query and update sub-systems. However a person may not have the same access permit for querry and update.



The Course master file contains vital, course related data, about the school's course organization. Mainly it contains precise information on time and space coordinates of each course as well as its allocated student and teaching resources.
Automatic accesses are done 500-2000 times a day, during the regular run of the attendance monitoring sub-system. Errors would create a chain type reaction of data rejection process.
For this security reason all interactive accesses (query & update) are limited to only the vice-principal in charge of the school.

The student master file contains personal and academic type data for each individual student.

Automatic accesses are done 500-2000 times a day for updating purposes during the regular run of the attendance monitoring sub-system. Interactive updating is permitted by 2 - 3 persons in the records office. Query access is permitted to all record related persons (including students and parents).

The class master file contains vital information on the student distribution in the classes and related academic information.
Interactive updating is done by 2 - 3 persons in the records office.
Query access is permitted to all 200 teachers.

Figure 19 - Data base description and its access dependent organization.

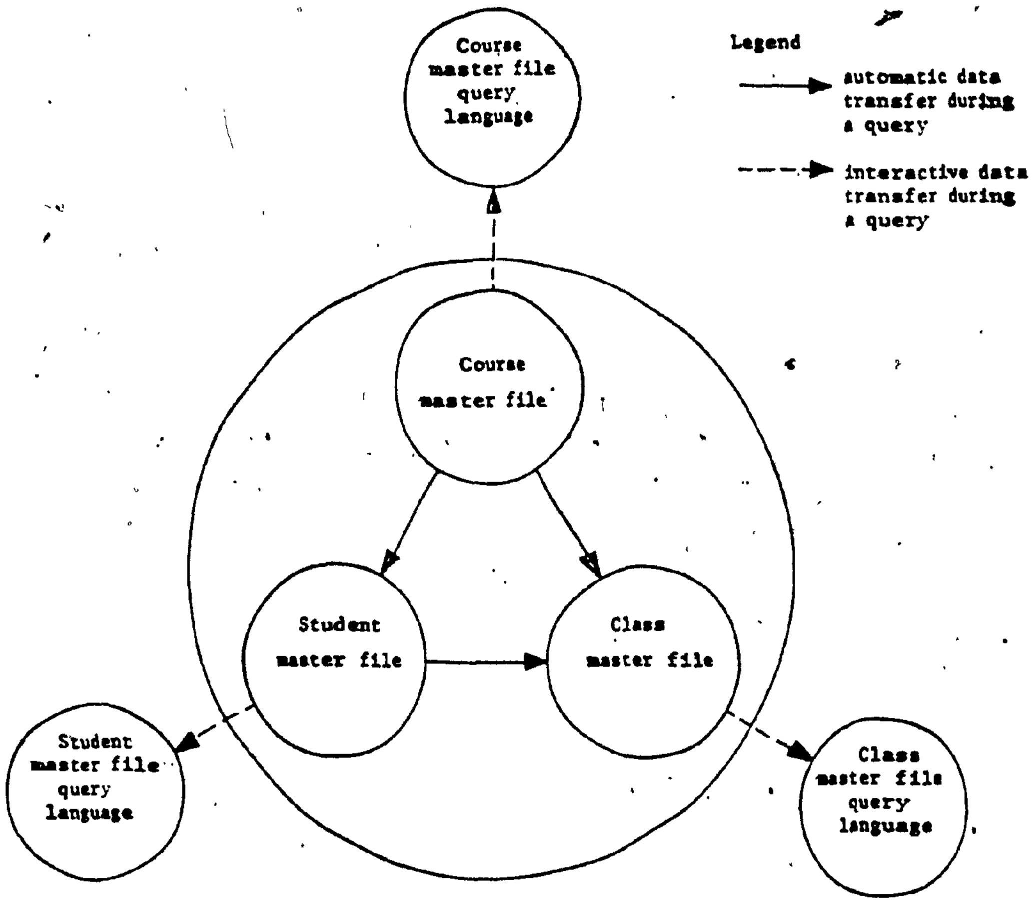


Figure 20 - Data base query sub-system.

Each master file has its own query language accessor sub-system.

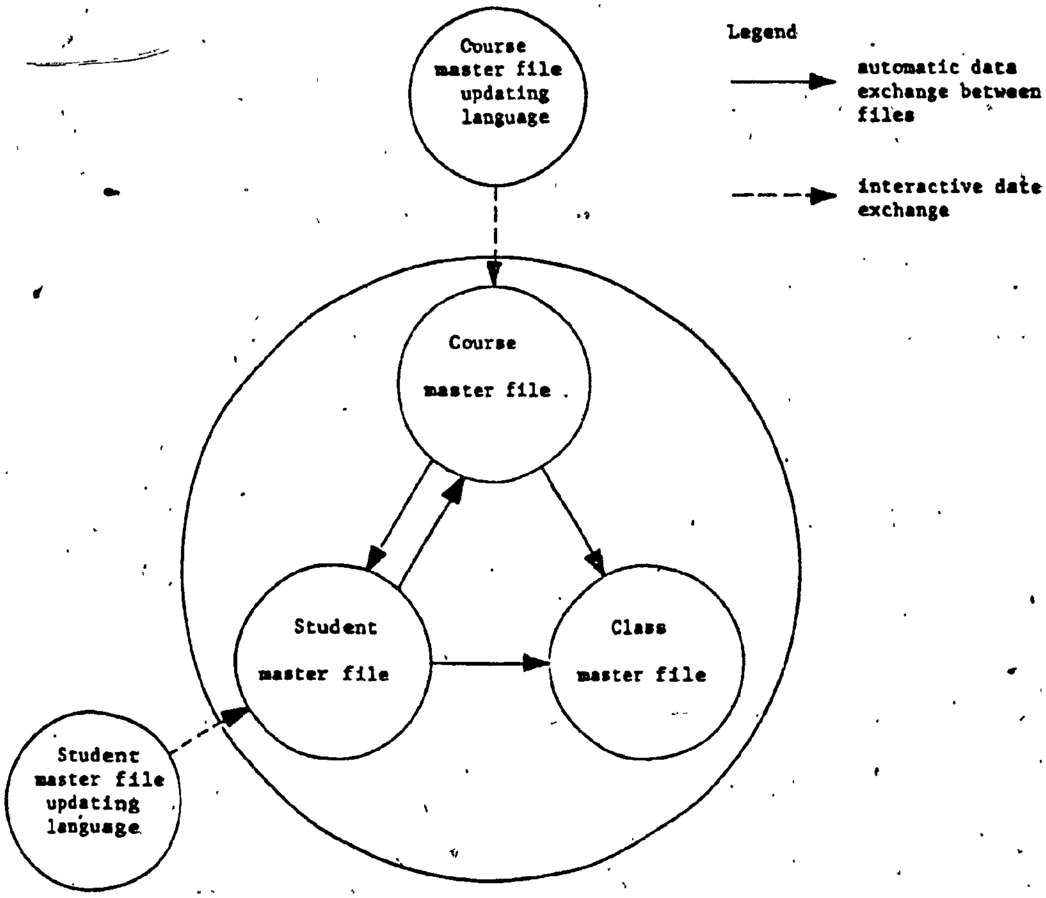


Figure 21 - Data base update sub-system.

Only the course master file and the student master file can be updated interactively. The class master file records are automatically updated when the others are.

report facilities were either on-line or pseudo-batch.

The effects of these five requirements on the system is shown in figures 15a and 15b. Figures 15c and 16 show the overall logical design of the system. The body of this system is its on-line data-base. Its skeleton and vasculature are the query and update subsystems which support and permit the rapid and accurate use of the data-base. Conceptually the system's core would be its iterative production cycle in which the truancy information is processed. Figures 17 and 18 provide an overview of the security system (9) designed to meet the joint objectives of acceptable error rate, system integrity and confidentiality. Figure 19 gives (41) additional details about the data-base. Figures 20 and 21 give an overall view of the update and the query sub-systems. (8)

To gain understanding of the overall system we are going to use ABL notation.* (19, 21, 30) This notation is ideally suited to describe structured logical interrelations between a system's components. In ABL notation the overall system is first broken down into its main entities called clusters. Then each cluster is broken down further into sub-entities called alternatives (see figure 22). In turn each alternative is represented by a set of independent action components which are to be

---

* Alternative Based Language

1	Cluster - 1		
	1.1	Alternative 1	Then go to cluster 2
	1.2	Alternative 2	Then go to cluster 3
2	Cluster - 2		
	2.1	Alternative 3	Then go to cluster 2
	2.2	Alternative 4	Then go to cluster 2
	2.3	Alternative 5	Then go to cluster 3
3	Cluster - 3		
	3.1	Alternative 6	Then go to cluster 0 i.e. finish
	3.2	Alternative 7	Then go to cluster 2
	3.3	Alternative 8	Then go to cluster 4
	⋮		
N	Cluster - N		
	N.1	Alternative n-1	Then go to cluster 0 i.e. finish
	N.2	Alternative n	Then go to cluster 2

Figure 22 - ABL model of the functional breakdown and control flow of a system.

executed in a predetermined sequence. However this sequence of actions is to be executed only if a specified group of attached conditions (called predicates) prevail. While each alternative is unique any of the actions or predicates that this alternative accesses may be shared by any number of alternatives (see figure 23). The sequential string of actions and predicates formed from concatenating ABL alternatives is specified completely by the ABL program's strategy matrix in conjunction with its next state vector. The strategy matrix and next state vector specify the ABL program's control flow (see figure 24).

This logic flow starts at the strategy matrix's first row or cluster. During execution of the ABL program the conditions that are specified by the sub-set of predicates selected by this strategy are evaluated. Should the conditions specified by the first alternative in this cluster be met, then the actions specified by this alternative will be executed. After that the logic flow will branch to the indicated cluster. Should the conditions specified by the first alternative of this cluster fail to be met, then the next alternative for that cluster or row of the strategy matrix will be evaluated. In this way the ABL notation specifies both the control flow of

1. Cluster - 1

1.1 Alternative - 1 Then go to cluster 2

If Condition 1
Condition 6

Action 1
Action 3
Action 5

1.2 Alternative - 2 Then go to cluster 3

If Condition 2
Condition 3

Action 3
Action 6
Action 4

2. Cluster - 2

2.1 Alternative - 3 Then go to cluster 2

If Condition 2
Condition 7
Condition 5

Action 9
Action 10
Action 11

N Cluster N

Figure 23 - ABL model of the functional breakdown of a system.

The execution of any alternative is dependent on the attached conditions. Each execution is composed out of a selected group of actions. The program's logic will then branch to the next cluster specified in the alternative.

			M	Number of subfunctions (alternatives)									
			1	2	3	4	5	6	7	8	-	-	M
	Clusters												
STRATEGY MATRIX	Cluster	1	x	x									
	Cluster	2			x	x	x						
	"	3						x	x	x			
	"												
	"												
	"	N											
	Predicates												
	Condition	1	Y							Y			
	"	2		Y	Y		Y						
	"	3		Y		Y							
	"	4				N							
	"	5			Y								
	"	6	Y						N	N			
	"	7			Y			Y		Y			
	"	8						Y					
	Actions												
	Action	1	1										
	"	2											
	"	3	2	1			1	1	1	1			
	"	4		3				2	2				
	"	5	3										
	"	6		2									
	"	9			1	1							
	"	10			2			3					
	"	11			3								
NEXT STATE VECTOR	**Then go to cluster no.**		2	3	2	2	3	3	2	4			

Figure 24 - An example of an ABL table with the program's logical path through the clusters contained in its strategy matrix determined by its next state vector. The ABL program is the concatenated sequence of alternatives determined by this path. This sequence in turn specifies which predicates and actions are accessed at a given point in the program.

the program and the sequence of actions that are executed. The path of the logic flow through the ABL table is determined by the circumstantial conditions in existence at the time and by the predicates specified for each alternative. This logical path is flexible and adaptable to a variety of possible situations.

'Walking' through the ABL table one gains understanding of the logical description of a system in a shorter and more concise manner than with many other forms of documentation. This method of partitioning the strategy of a program into clusters of alternatives allows one to easily locate logical contradictions and oversights. Furthermore ABL's structure allows one to modify a program with ease and clarity. ABL is thus very well suited to the design and simulation of a complex system.

ABL notation permits also the logical description of all processes with the same efficiency whether it is a manual procedure, a complex system, a program or a simple subroutine. These allow a system's logical structure to be represented in a uniform way from its highest level into its lower level components. To perform such a hierarchial analysis of its logical structure we simply consider each of the actions of the highest level ABL table of the system's organization and if a more detailed description is needed for any action then turn this action into another ABL table of a lower level. Again if a lower level action needs more details we may repeat the same process until the logical description of any task is specified to a desired detail. In this way we can assemble all system related tasks (whether manual or computerized, complex or simple) under the same logical superstructure. For a complex system this multi-level (40) logical description greatly facilitates communication between the designer, the user and the maintenance team. While logical description at level 1 - 2 would be of interest mainly to the system manager, the description at level 2 of selected modules would be of interest to the operator or to corresponding user groups and the description at level 3 - 4 of specific modules would be of interest to the analyst or programmer (see figure 25). Thus operation and maintenance of the system would be much easier.

Our intended attendance monitoring information system was first divided up into the following five logical modules: the data-base subsystem, the update subsystem and the query subsystem, the attendance monitoring subsystem and the utility support subsystem (see figure 15-A). These modules were further subdivided (see figure

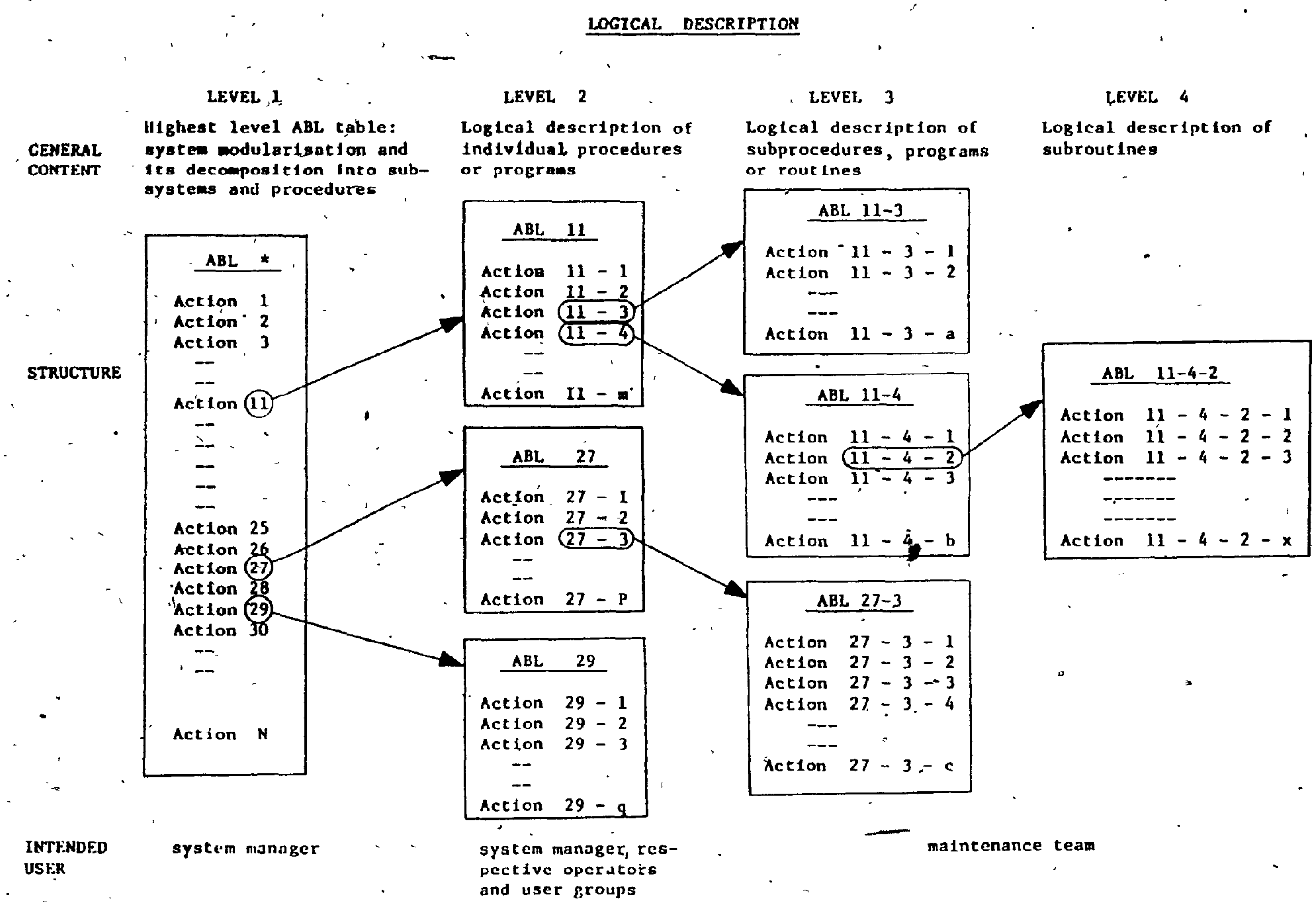


Figure 25 - A multi-level logical description using ABL notation allows to break down a system into its most elementary components.

16) and then finally we determined 44 actions necessary for the implementation of the system (see Table 1). Together with the predicates they will form the highest level ABL table. Each of these actions is a complete operation in itself. Put together in a logical sequence these actions will permit to set up, run and use the whole information system. However due to the complex nature of the intended truancy information system and due to the many constraints mentionned earlier in this chapter, and also considering the history of past failures, we had first to create a logical plan for the modeling of this system. Each of the above mentioned actions must respect some criterions and in trying to solve how to tailor each action to suit its defined criterions, we must follow some priorities.

To see clearly the modeling task of the designer we built an ABL table called MODELING. This ABL table (see appended printout in Annex I) interrelates all actions to all defined criterions.

In this table one can see what set of constraints a given group of actions would have to obey in order to be considered adequately designed. Should any action not respect its specifications, there would be a deadlock in the logic flow of the design table. By following

A1 : Install the terminal in school
A2 : Set up the course master file
A3 : Set up the student master file
A4 : Set up the class master file
A5 : Set up course master file updater program
A6 : Set up student master file updater program
A7 : Set up course master file query program
A8 : Set up student master file query program
A9 : Set up class master file query program
A10 : Update course master file interactively
A11 : Update student master file interactively
A12 : Update class master file interactively
A13 : Verify course master file data
A14 : Verify student master file data
A15 : Verify class master file data
A16 : Produce teachers data generation kits
A17 : Set up data collection routes in school
A18 : Install optical mark reader on terminal
A19 : Verify data distribution routes
A20 : Determine data usage parameters
A21 : Set up daily initialization interactive program
A22 : Set up data collection pseudobatch program
A23 : Set up #ats. data processing pseudobatch program
A24 : Set up procedure to print spooled output
A25 : Perform daily system initialization
A26 : Perform transaction data generation
A27 : Perform transaction data collection
A28 : Perform data processing (reporting and storage)
A29 : Perform data distribution
A30 : Verify output usage in decision making
A31 : Punch new # ABS-cards
A32 : Update teachers data generation kits
A33 : Query course master file interactively
A34 : Query student master file interactively
A35 : Query a class master file interactively
A36 : Write a file in course master file query language
A37 : Query course master file pseudobatch mode
A38 : Print spooled output
A39 : Write a file in student master file query language
A40 : Query student master file pseudobatch mode
A41 : Write a file in class master file query language
A42 : Query class master file pseudobatch mode
A43 : Produce # ABS-cards for the class
A44 : No action - next cluster

Table 1 - List of the 44 actions of the system's highest level ABL table.

Each of these actions is a complete task.

through the functional breakdown of this modelling table we can establish the sequence of problem solving that took place to come out with the desired system. This ABL table MODELING is equivalent to a logical simulation of the intended system. Assuming that each alternative is solved before branching into the next, only at the end of the logic flow in this table, we can assume that all actions are adequately designed and ready for implementation. Then we take the same actions and interrelate them but now according to an operational sequence to set up, run and use the system. This implementation logic is described in another ABL table called SYSTALL (see appended printout in Annex I). ABL tables SYSTALL and MODELING share the same actions but while ABL table MODELING deals with design predicates, ABL table SYSTALL deals with operational predicates. As we mentioned earlier, each action in Table 1 is a complete operation in itself but the more detailed logical description of any individual action requires its decomposition into an ABL table of a lower level. The logical description of a complex system is often described in a tree-like manner. The complexity of such a tree may be given by its depth. In our case the logical description of the ABL table SYSTALL reaches a depth of four. In this thesis we will detail only the system's essential modules to this depth. These modules are: the six composite actions (A25, A26, A27, A28, A29, A30) given in TABLE 1 that correspond to the system's production cycle, and All which describes the updating of the student master file. Figure 26 shows the hierarchy of the tree of ABL tables that define these seven actions in ABL table SYSTALL to depth four.

In the computer printout in Annex I we give the logical descriptions for all ABL tables shown in figure 26. To facilitate the understanding of each ABL table, four logical descriptions of increasing detail levels are provided for each of them. These levels are intended to provide the reader with specific as well as conceptual details. The first level or description no. 1 gives the functional breakdown of a task into its main procedures (clusters) and sub-procedures (alternatives). Description no. 2 provides the set of conditions (predicates) which control the execution of any given alternatives. This represents the effective control flow. Description no. 3 adds the decomposition of each alternative into a string of corresponding actions. Description no. 4 gives for the ABL table the overall list of actions and predicates with the logical path through them, that is

LOGICAL DESCRIPTION

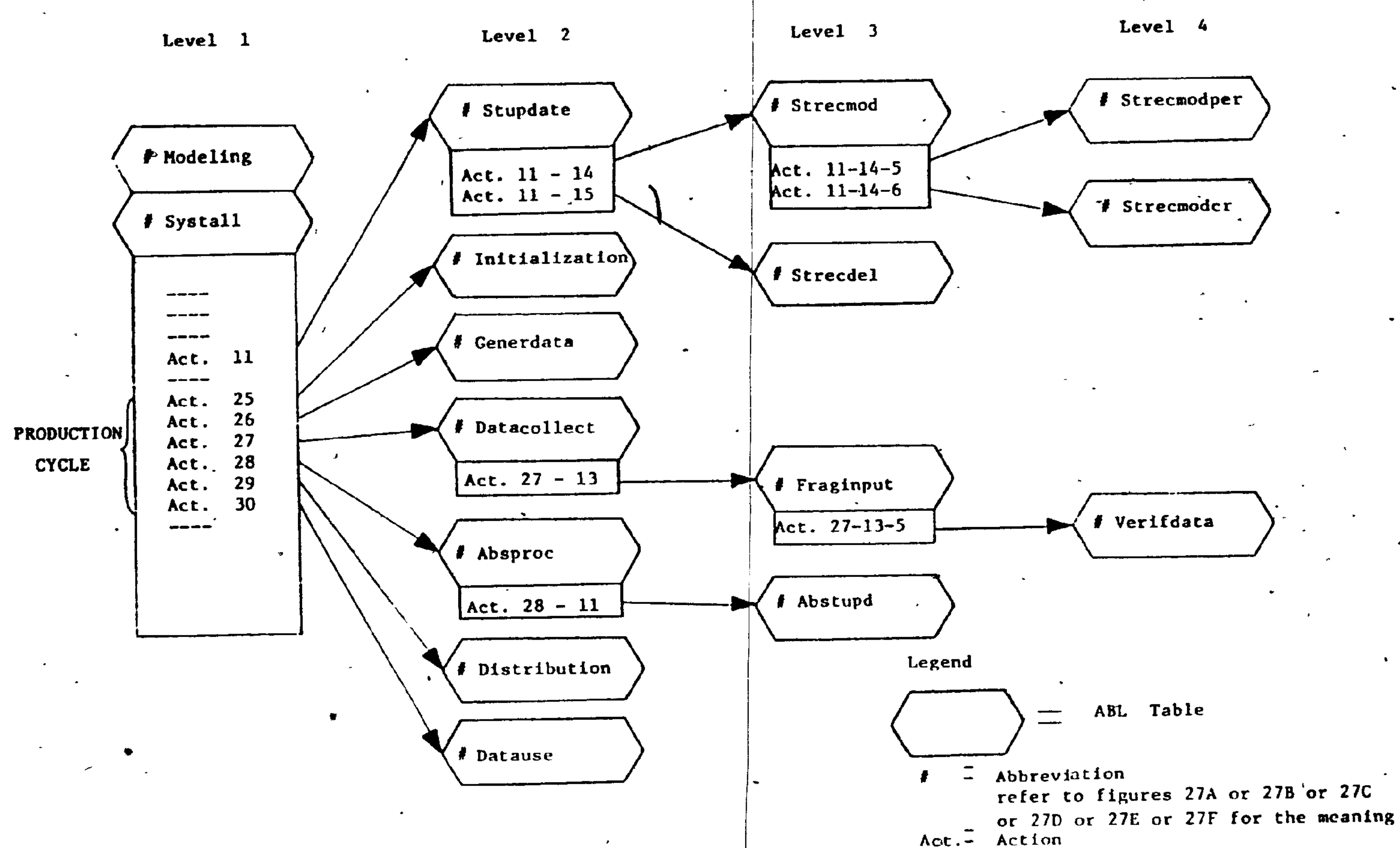


Figure 26 – This tree of ABL tables shows the plan of the multilevel logical description we will find in the following computer printout.
The underlying box of any particular table indicates which of its action components has been developped into another ABL table of a lower level.

specified by the table's strategy matrix and next state vector.

To generate the ABL tables, however we need a special program. This program (called "CLOCKIN") was supplied by Dr W.M. Jaworski. It needs two specially constructed tables as an input for each ABL table. Using this program interactively with the help of different commands we can display any ABL table at various detail levels. This is how the ABL tables in Annex I were produced.

The input tables to produce the ABL tables MODELING and SYSTALL are included in Annex II as an example.

1. ABKPLST

   Procedure file name: It transfers transaction data from a dummy input file to a verified transaction file. It produces a report and an error listing.

2. ABKPNLT

   Procedure file name: It transfers transaction data from a dummy input file to a verified transaction file. It produces a short error summary.

3. ABS-CARD

   Cards: Specially printed keypunch cards for the data generation process. One card for each student in the course is prepunched in the data generation kit.

4. ABS-REC

   Reserved record in the student master file: This record contains attendance related pointers for the student record.

5. ABS-SUBTOTALS-IN-CR

   Variable: Total number of lectures missed by the student in a particular course. It is compiled for the whole year and for each semester.

6. ABSCOM

   Procedure file name: In the morning the operator calls for execution an interactive program to initialize a new production cycle.

7. ABSOIR

   Procedure file name: This is a pseudobatch procedure called to update the data-base with the truancy transaction data at the end of the day.

8. ABSPROC

   ABL table name: It performs data processing (reporting and storage) on truancy transaction data.

Figure 27A - This table contains abbreviations contained in the computer printout in Annexe I.

9. ABSTUPD

ABL table name: This routine is part of ABSPROC and it updates the student record with attendance related data.

10. ATS

Attendance Monitoring Subsystem.

11. ATS INTERACT PROGS

Interactive programs in the attendance monitoring subsystem.

12. CLASS-REC

Class record in the class master file.

13. CRE, STUDENT NO

Command in the interactive updating process of the student master file. It means: Create a new student record with the retrieval key being the student's identification number.

14. COURSENO

Variable: Course identification number. This is a unique 8 digit retrieval key for each course or class record.

15. CRNO

Course identification number: This is a unique 8 digit retrieval key for each course or class record.

16. DABCR

File name: Daily absentee cards or verified transaction data file for the day.

17. DAT-GEN-KIT

Cards and pouch: Data generation kits supplied to the teachers. These kits allow the teacher to easily record class attendance.

18. DATACOLLECT

ABL table name: Transaction data collection procedure.

Figure 27B - Follow up from Figure 27A

19. DB

Data-base.

20. DEL, STUDENT NO

Command in the interactive updating process of the student master file. It means: delete a student (with the retrieval key being the student's identification number) from the data base.

21. DISTRIBUTION

ABL table name: This is the redistribution process of all data and reports related to the attendance monitoring subsystem.

22. FIN

Command in the interactive updating process of the student master file. It means: finish the updating session and wait for the related listing.

23. FIN, NOLIST

Command in the interactive updating process of the student master file. It means: finish the updating session and suppress all related listing.

24. FRAGINPUT

ABL table name. Fragmented transaction data collection process where data is partially collected into a single transaction file.

25. GENERDATA

ABL table name: Transaction data generation procedure.

26. ITNO

Fixed field item number: A record is subdivided into numbered items.

27. LAST-DATE-ABSENT

Variable: Date of the last absence of a student from at least one lecture period during that day.

Figure 27C - Follow up from Figure 27B

28. LAST-DATE-ABSENT-FROM-LECTURE-COURSENO

Variable: Date of the last lecture missed by a given student from a particular course.

29. LAST-DAY-ABS-IN-CR

Variable: Date of the last absence of a student from a particular course. (Same as 28 except in a shorter form)

30. MF

Master file.

31. MOD, STUDENT NO

Command in the interactive updating process of the student master file. It means: modify an existing student record, with the retrieval key being the student's identification number.

32. MODETUD

Procedure file name: It calls an interactive program for execution in order to update the student master file.

33. NA

Variable: Number of students on report A.

34. NB

Variable: Number of students on report B.

35. NOMT

Variable: Number of truant students met by the vice principal about truancy information.

36. NOPH

Variable: Number of phone calls made to parents about truancy information.

Figure 27D - Follow up from Figure 27C

37. QUS

Data-base query subsystem.

38. PSEUDO-BATCH

It is a mode of processing where the operator on an interactive terminal first defines a data file, then calls interactively for its execution in batch mode, the output arrives later.

39. STRECDEL

ABL table name: Student record deletion routine.

40. STRECMOD

ABL table name: Student record modification routine.

41. STRECMODCR

ABL table name: Student record modification subroutine for changes on the record's course registration area.

42. STRECMODPER

ABL table name: Student record modification subroutine for changes on the record's personnal information area.

43. STUD-MAST-FILE

Student master file.

44. STUD-REC

Student record of the student master file.

45. STUPDATE

ABL table name: Update the student master file.

46. SUBTOTAL-OF-LECTURES-MISSED-IN-COURSENO

Variable: Total number of lectures missed by the student in a particular course. It is compiled for the whole year and for each semester.

Figure 27E - Follow up from Figure 27D

47. SYSTALL

ABL table name: Overall sequence of procedure to set up, run and use the entire truancy information system.

48. TOTAL-DAYS-ABSENT

Variable: Total number of days the student missed from school. By definition a day absent is when at last one lecture is missed.

49. TOTAL LECTURES MISSED

Variable: Total number of lectures missed by a student in all his courses combined together. It is compiled both by semester and for the whole year.

50. TRANS-REC

Transaction record.

51. UPS

Data-base update subsystem.

52. UTS

Utility subsystem to support the user.

53. VERIFDATA

ABL table name: Routine to verify the transaction data for possible errors.

54. VP

Person: Vice principal

Figure 27F - Follow up from Figure 27E

CHAPTER 4

PHYSICAL IMPLEMENTATION OF THE SYSTEM

In the precedent chapter we discussed the logical description of the attendance system. In this chapter we will discuss the physical environment used in its successful implementation. (6, 29)

We will discuss the equipment used; the system's data base and the update subsystem.

We will also describe the different phases of the daily production cycle, the maintenance tasks and the query subsystem.

The fundamental objective of this implementation was to assure that the student truancy information actually helped combat truancy related problems. All programs, procedures, form designs, hardware, etc, that did not meet this goal were considered superfluous. (38)

This chapter is subdivided to follow the outlined implementation plan in ABL table SYSTALL (see table 2). The numbering of these subdivisions corresponds to the subdivisions in table 2.

```
1. SET UP THE BASIC SYSTEM ( # UTS. AND # DB. )                ==$                      < 1, 1, 1, 2 >
   1.1= INSTALL TERMINAL IN SCHOOL                             ==   --THEN-GO-TO-CLUSTER-> 1
   1.2= SET UP EMPTY DATA-BASE                                 ==   --THEN-GO-TO-CLUSTER-> 1
   1.3= SET UP ALL #QUS. AND #UPS. INTERACTIVE PROGRAMS.       ==   --THEN-GO-TO-CLUSTER-> 1
   1.4= UPDATE ALLOCATED FIELDS IN DATA-BASE                   ==   --THEN-GO-TO-CLUSTER-> 2
2. INITIAL DATA-BASE VERIFICATION. ( # QUS. AND # UPS. )       ==$                      < 2, 3, 3 >
   2.1= VERIFY INFORMATION IN DATA-BASE                        ==   --THEN-GO-TO-CLUSTER-> 2
   2.2= UPDATE INCORRECT INFORMATION IN DATA-BASE              ==   --THEN-GO-TO-CLUSTER-> 3
   2.3= NO UPDATE IS NECESSARY                                 ==   --THEN-GO-TO-CLUSTER-> 3
3. SET UP THE ATTENDANCE MONITORING SUBSYSTEM ( # UTS. )       ==$                      < 3, 4 >
   3.1= GIVE OUT ACCESSORIES FOR ATTEND.MONIT.SUBSYSTEM.       ==   --THEN-GO-TO-CLUSTER-> 3
   3.2= SET UP #ATS.INTERACT.PROGS.FOR DAILY PRODUCTION CYCLE  ==   --THEN-GO-TO-CLUSTER-> 4
4. DO REGULAR RUN OF THE ATTENDANCE MONITORING SUBSYSTEM (# ATS.)==$              < 4, 4, 4, 4, 4, 4, 5 >
   4.1= #ATS. DAILY DATA USAGE PARAMETER DEFINITION            ==   --THEN-GO-TO-CLUSTER-> 4
   4.2= #ATS. DAILY TRANSACTION DATA GENERATION                ==   --THEN-GO-TO-CLUSTER-> 4
   4.3= #ATS. DAILY TRANSACTION DATA COLLECTION                ==   --THEN-GO-TO-CLUSTER-> 4
   4.4= #ATS. DAILY TRANSACTION DATA PROCESSING                ==   --THEN-GO-TO-CLUSTER-> 4
   4.5= #ATS. DAILY DATA DISTRIBUTION                          ==   --THEN-GO-TO-CLUSTER-> 4
   4.6= #ATS. DAILY OUTPUT USAGE                               ==   --THEN-GO-TO-CLUSTER-> 4
   4.7= NO ACTION,NEXT CLUSTER                                 ==   --THEN-GO-TO-CLUSTER-> 5
5. MAINTAIN THE EXACTNESS OF THE SYSTEMS DATA  ( # UPS. )      ==$                      < 4, 4, 6 >
   5.1= UPDATE NON ATTENDANCE RELATED DATA                     ==   --THEN-GO-TO-CLUSTER-> 4
   5.2= UPDATE ATTENDANCE RELATED DATA                         ==   --THEN-GO-TO-CLUSTER-> 4
   5.3= NO ACTION, NEXT CLUSTER                                ==   --THEN-GO-TO-CLUSTER-> 6
6. INFORMATION EXTRACTING AND REPORTING ( # QUS. )             ==$                < 4, 4, 4, 4, 4, 4 >
   6.1= QUERY  COURSE MASTER FILE INTERACTIVELY                ==   --THEN-GO-TO-CLUSTER-> 4
   6.2= QUERY  STUDENT MASTER FILE INTERACTIVELY               ==   --THEN-GO-TO-CLUSTER-> 4
   6.3= QUERY  CLASS MASTER FILE INTERACTIVELY                 ==   --THEN-GO-TO-CLUSTER-> 4
   6.4= QUERY  COURSE MASTER FILE PSEUDOBATCH                  ==   --THEN-GO-TO-CLUSTER-> 4
   6.5= QUERY  STUDENT MASTER FILE PSEUDOBATCH                 ==   --THEN-GO-TO-CLUSTER-> 4
   6.6= QUERY  CLASS MASTER FILE PSEUDOBATCH                   ==   --THEN-GO-TO-CLUSTER-> 4
```

Table 2 - Plan to implement, run and use of the attendance monitoring information system. (ABL table SYSTALL).

```
1.  SET UP THE BASIC SYSTEM ( # UTS. AND # DB. )              ===1          < 1, 1, 1, 2 >
    1.1= INSTALL TERMINAL IN SCHOOL                           ==     --THEN-GO-TO-CLUSTER-> 1
         1.1 IF  NOT  TERMINAL IS SET UP IN SCHOOL
                  A1>  INSTALL THE TERMINAL IN SCHOOL

    1.2= SET UP EMPTY DATA-BASE                               ==     --THEN-GO-TO-CLUSTER-> 1
         1.2 IF  TERMINAL IS SET UP IN SCHOOL  AND IF
                NOT  ALL DATA-BASE FILES ARE SET UP ON-LINE
                  A2>  SET UP THE COURSE MASTER FILE
                  A3>  SET UP THE STUDENT MASTER FILE
                  A4>  SET UP THE CLASS MASTER FILE

    1.3= SET UP ALL #QUS. AND #UPS. INTERACTIVE PROGRAMS.     ==     --THEN-GO-TO-CLUSTER-> 1
         1.3 IF  ALL DATA-BASE FILES ARE SET UP ON-LINE  AND IF
                NOT  QUERRY AND UPDATE SUBSYSTEMS ARE SET UP
                  A5>  SET UP COURSE MASTER FILE UPDATER PROGRAM
                  A6>  SET UP STUDENT MASTER FILE UPDATER PROGRAM
                  A7>  SET UP COURSE MASTER FILE QUERY  PROGRAM
                  A8>  SET UP STUDENT MASTER FILE QUERY  PROGRAM
                  A9>  SET UP CLASS MASTER FILE QUERY  PROGRAM

    1.4= UPDATE ALLOCATED FIELDS IN DATA-BASE                 ==     --THEN-GO-TO-CLUSTER-> 2
         1.4 IF  ALL DATA-BASE FILES ARE SET UP ON-LINE  AND IF
                QUERRY AND UPDATE SUBSYSTEMS ARE SET UP  AND IF
                NOT  ALLOCATED FIELDS IN THE DATA-BASE ARE FILLED
                  A10>  UPDATE COURSE MASTER FILE INTERACTIVELY
                  A11>  UPDATE STUDENT MASTER FILE INTERACTIVELY
                  A12>  UPDATE CLASS MASTER FILE INTERACTIVELY
```

Table 3 - Implementation plan to set up the basic system (From ABL table SYSTALL).

1. <u>Set up the basic system (see table 3)</u>

1.1 Install the terminal in school.

The equipment configuration used in the attendance system is illustrated in Figures 28 and 29. In the school we maintain a Decwriter II terminal connected in series with a Hewlett Packard 7260-A model mark-sense and punched card reader. Both of these I/0 units communicate with Concordia University's CYBER 172 computer through a modem at 300 baud and we used 20 MB of disk storage. The attendance system also requires the monthly use in batch mode of the Concordia University computer center's high-speed lineprinter and the weekly use of the University's card punch and interpreter facilites.

1.2 Set up the empty data-base.

Given this hardware configuration the data-base was designed (4, 16, 25) in a manner to permit secure interactive and modularized access. As we mentionned earlier in chapter 3 (see Figures 15 C, 16, 17, 18, 19) the data-base is (41) divided into a course master file, a student master file, and a class master file. These random access files are used for interactive query and update tasks.

The course master file contains information on the time and space coordinates of each course as well as its allocated student population and teaching resources. The general structure of the course master file can be seen in Figure 30.

The student master file contains personal and academic data for each student enrolled in the school. For each student record, the academic record area contains information on the student's course registration, performance and attendance. The general structure of the student master file can be seen on Figure 31. (54)

The class master file contains information on the student distribution in the classes and related academic information. This file mostly contains redundant data which could be obtained from the other two master files but only by exhaustively searching the data-base for specific class related information about a group of students. The obvious disadvantages of such a process made it necessary to redundantly group student information according to classes. The class master file's organization substantially cuts down on transfer time from disk to central memory during class related accesses. This file's structure is outlined on Figure 32 (54). The master files are cross referenced through a series of pointers. Such a cross reference system enables to (52)

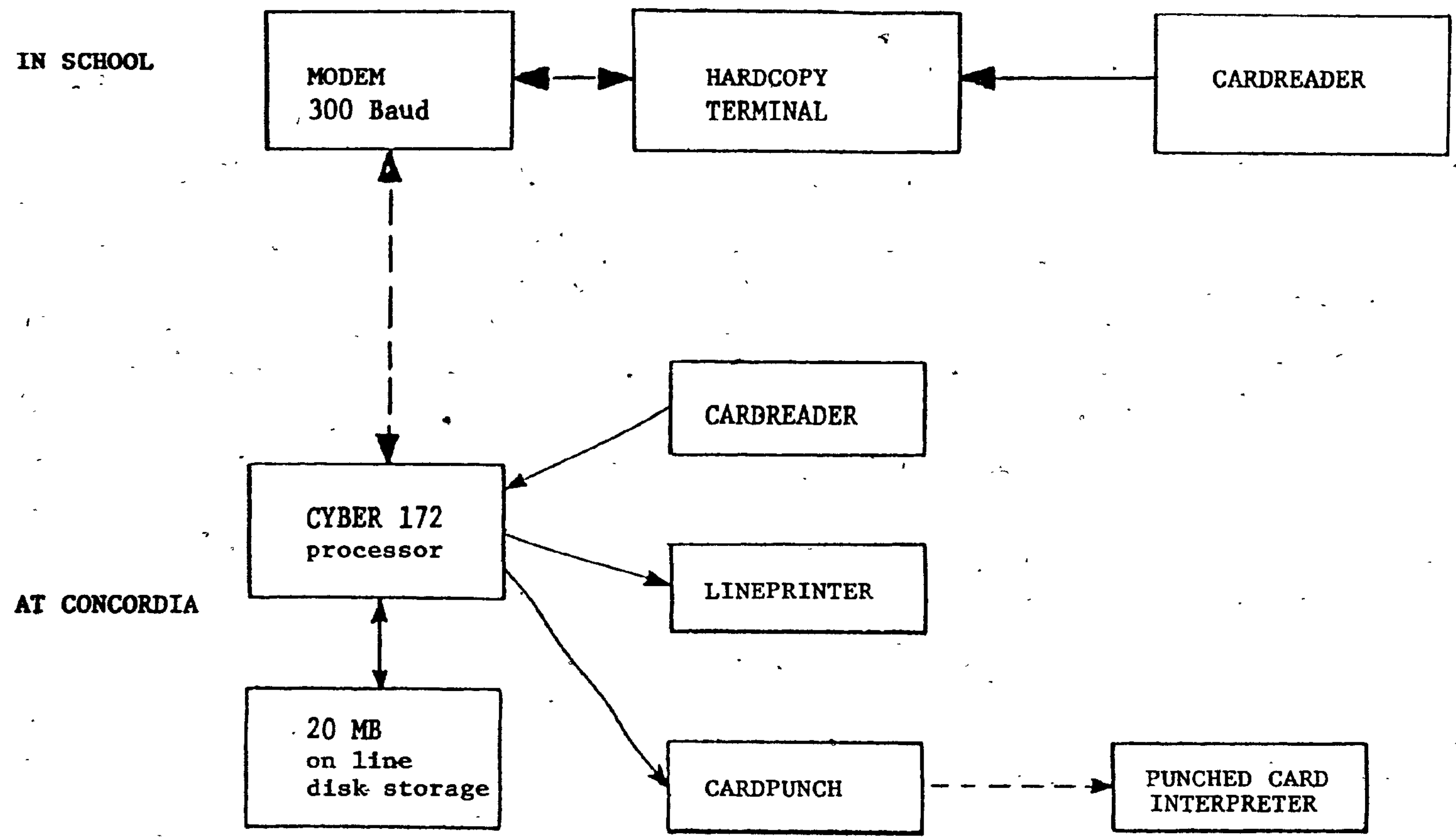


Figure 28 Equipment used for the attendance monitoring system.

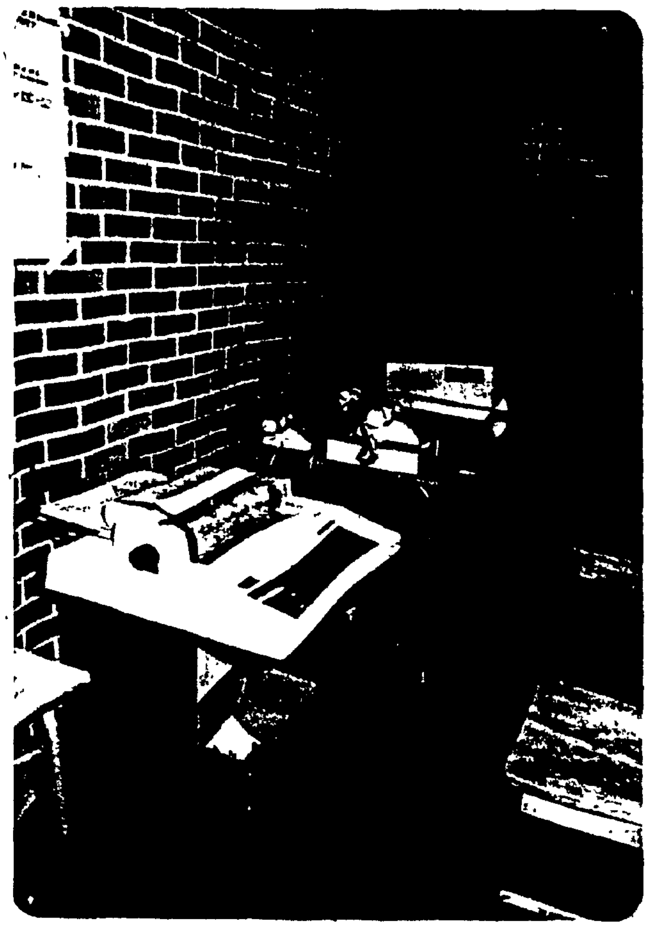


Figure 29 The actual equipment used in the computer center of Andre Laurendeau comprehensive high school.

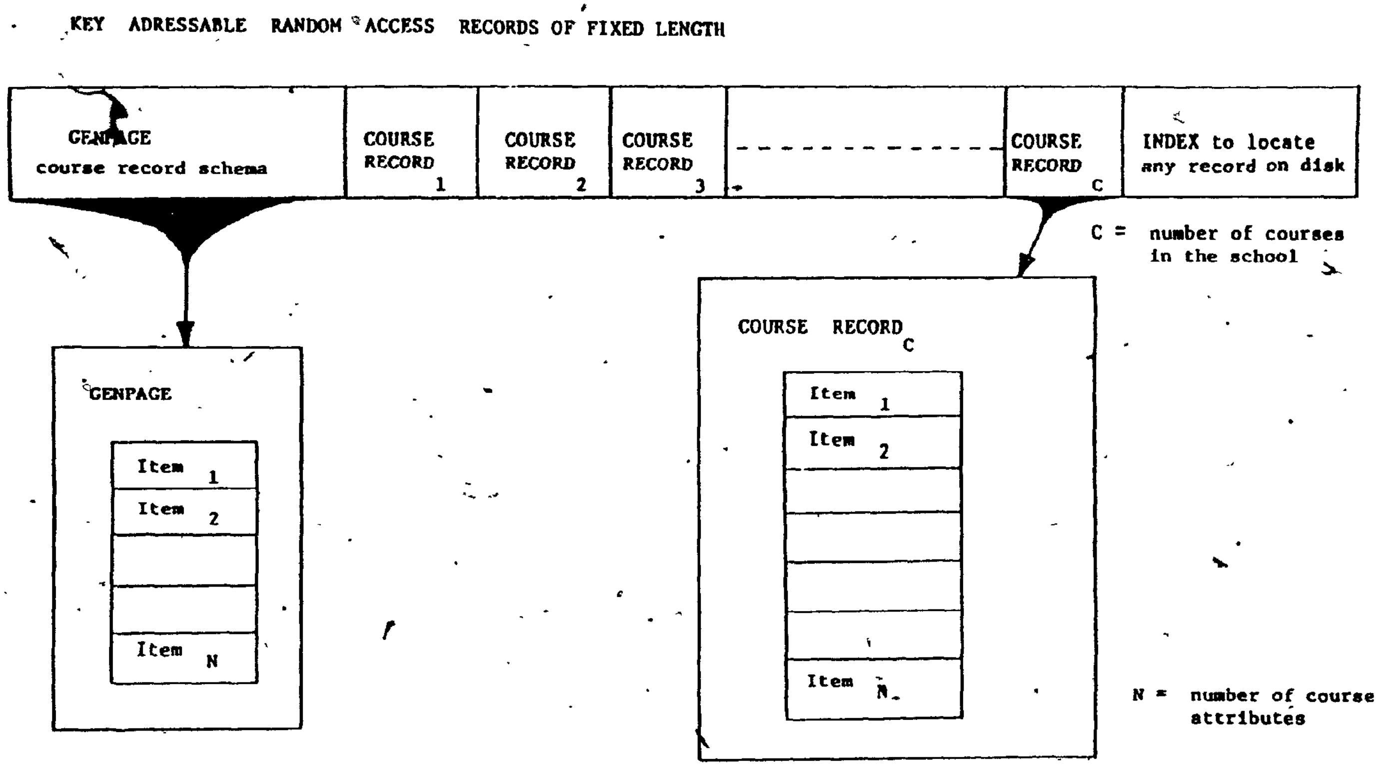


Figure 30 Course master file structure

This course master file is a key adressable random access file. Each record is of fixed length. The first record of this file called Genpage is created at the time when this file is set up. Genpage record items consist of special values, messages and pointers to the fields of the course record. The course record contains 24 data items (as can be seen on figure 50). Some of these will act as a critical value for the application programs of the attendance monitoring subsystem.

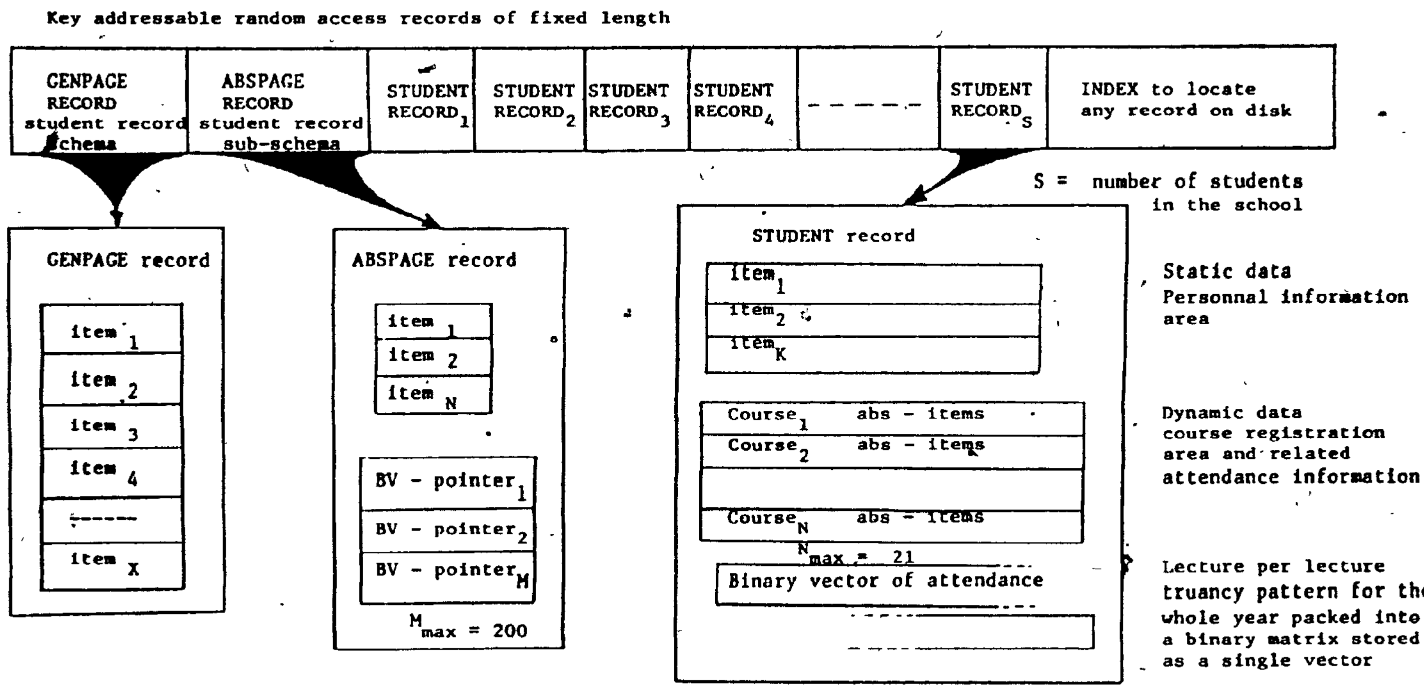


Figure 31 General structure of the student master file

See Figure 31-A for more details

Legend

GENPAGE = general schema for the student record

ABSPAGE = sub-schema for the attendance monitoring subsystem

BV - pointer = binary vector pointer to the student record

ABS - items = truancy related data items

The student master file is a key addressable random access file. Each record is of fixed length. The first record called GENPAGE is created when the file is set up and acts as the general schema for the student record. GENPAGE record items consist of special values, messages or pointers to the fields of the student record. The STUDENT record contains a fixed number of static data (the student's personnal information). A variable number of course related data that is frequently updated, course related pointers to the class master file and a binary storage area for attendance pattern which is equivalent to a sparse matrix storage system. The ABSPAGE record is created when the attendance monitoring subsystem's production cycle is set up. It acts as a sub-schema for the attendance monitoring subsystem. The ABSPAGE record items contain messages variable values and special limits for the ATS daily production cycle. Some of these are updated daily by the operator. It also contains an increasing set of BV-pointers (Binary vector pointers). Each binary vector pointer indicates a particular location on the binary vector attendance storage of the student record. Each day when the production cycle is run, a new binary vector pointer is computed from the day's and other data and then stored in the ABSPAGE record. The pointers in ABSPAGE and GENPAGE records are valid for all STUDENT records because the record subdivision identical for all variable per location. These records contain also non validated empty fields to be able to follow up an unforseen user requests that would require additionnal storage space.

Figure 31-A General structure of the student master file (follow up from Figure 31)

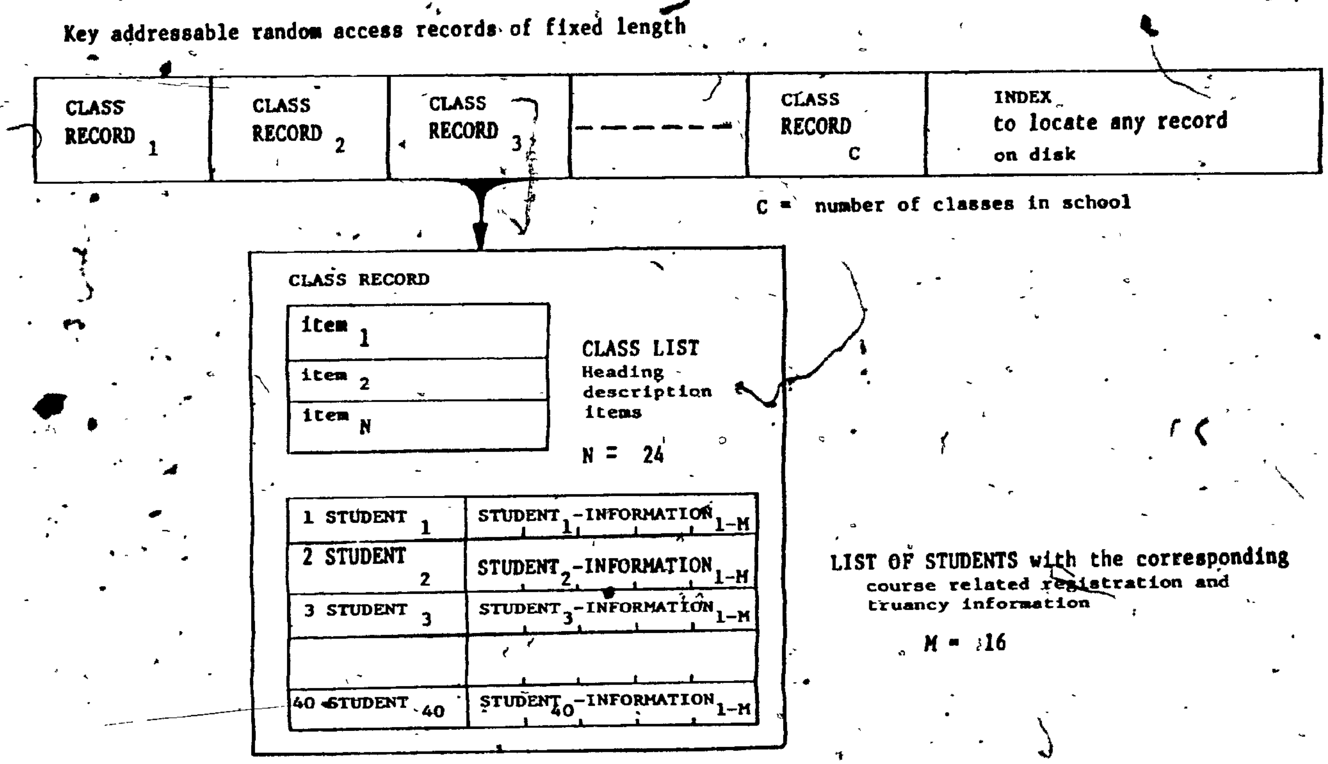


Figure 32 General structure of the class master file.

This file is created by a program from data in the student master file and in the Course master file. The key of each record is the same as the key for the records for the Course master file. Each record is a key addressable random access record of fixed length.

guard the data-base's integrity. Through the use of these cross-referenced files it is possible to check the daily attendance transaction file for inconsistencies and errors.

1.3 Set up all query and update subsystem interactive programs.

To permit the user to control his data-base we have to give him the facilities to display the content of the data-base and to edit the validated fields. For this we set up the following programs in form of procedure files:

course master file	update program
student master file	update program
course master file	query program
student master file	query program
class master file	query program

We have to train the different users to permit efficient use of the program corresponding to his or her task. (23)

1.4 Update allocated fields in the data-base. (3,49, 43, 50)

The three master files may be edited in interactive mode. This facility provides the user with the powerful tool needed to maintain the system's accuracy in monitoring the attendance of a large number of students. Throughout the whole year students keep changing or dropping courses. These changes continually effect all related class lists and course directories which makes it even more difficult to accurately track a student's attendance.

For reasons of access control and security, the course master file had its own editor which could be used only by the school's vice-principal in charge of the system.

The student master file is updated by the record's office via another editor. The class master file is updated automatically whenever the other master files are edited. Figure 21 in chapter 3 outlines these details.

3. Set up the attendance monitoring subsystem (see table 4).

3.1 Give out accessories for the attendance monitoring subsystem.

Once the data-base is set up and its contents verified we are ready to set up the attendance monitoring subsystem. Attendance information is recorded through pseudo-batch* runs on card deck data. To obtain reliable transaction data in a short delay, we provide teachers with a data generation kit for each of their classes. From the data-base we produce a class list for each course offered as well as an interpreted punched data card for every student in each class. These punched cards are used as student identification cards by the teacher when he takes attendance (see Figure 33 for a sample card). Each class list and its corresponding deck of punched identification cards is placed in a three-pouch plastic container (see Figure 34). These (55) punched cards were designed with the following objectives in mind. Firstly they should provide teachers with a

---

* By pseudo-batch we mean a batch process which is controlled interactively by the operator on a previously defined data file.

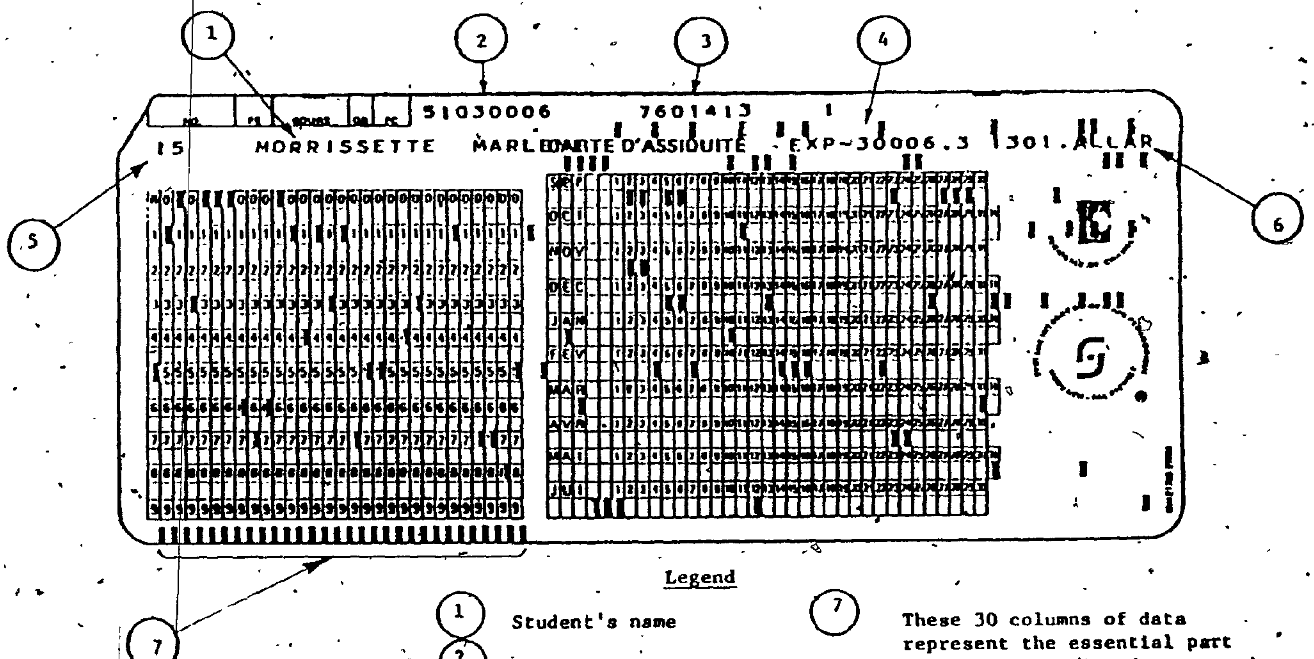
```
3.  SET UP THE ATTENDANCE MONITORING SUBSYSTEM ( # UTS. )          ===#                    < 3, 4 >

   3.1= GIVE OUT ACCESSORIES FOR ATTEND.MONIT.SUBSYSTEM.          ==    --THEN-GO-TO-CLUSTER-> 3
        3.1 IF  NOT  ATTENDANCE MONITORING SUBSYSTEM IS SET UP  AND IF
             DATA-BASE CONTAINS CORRECT NON TRUANCY INFORM.  AND IF
             NOT  TRANSACTION DATA GENERATION KITS ARE READY
                A42>  QUERY  CLASS MASTER FILE PSEUDOBATCH MODE
                A43>  PRODUCE # ABS-CARDS FOR THE CLASS
                A16>  PRODUCE TEACHERS DATA GENERATION KITS
                A17>  SET UP DATA COLLECTION ROUTES IN SCHOOL,
                A18>  INSTALL OPTICAL MARK READER ON TERMINAL
                A19>  VERIFY DATA DISTRIBUTION ROUTES
                A20>  DETERMINE DATA USAGE PARAMETERS

   3.2= SET UP #ATS.INTERACT.PROGS.FOR DAILY PRODUCTION CYCLE  ==    --THEN-GO-TO-CLUSTER-> 4
        3.2 IF  NOT  ATTENDANCE MONITORING SUBSYSTEM IS SET UP  AND IF
             TRANSACTION DATA GENERATION KITS ARE READY
                A21>  SET UP DAILY INITIALIZATION INTERACTIVE PROGRAM
                A22>  SET UP DATA COLLECTION PSEUDOBATCH PROGRAM
                A23>  SET UP. #ATS. DATA PROCESSING PSEUDOBATCH PROGRAM
                A24>  SET UP PROCEDURE TO PRINT SPOOLED OUTPUT
```

Table 4 - Implementation plan to set up the attendance monitoring subsystem.
(From ABL table SYSTALL).

fast, simple and transcription-free data generation process.

Secondly they should offer quick, error-free input through the card reader. Fifty of the eighty punched columns on the cards were not read into the transaction file. Sixty five columns contained information which was of interest and use to the teachers. These columns were interpreted to print the student's name and identification number, the course name and key and the teacher's name and identification number along the top edge of each card (see Figure 33). This information was useful to the teacher for easy card handling.

All essential transaction data (i.e. student record and course record retrieval keys and various pointers to the three data-base master files) were punched in the first 30 columns of each card. The first thirty columns of these cards were also printed with special clock marks. These clock marks activate the card reader to read only the specified columns and ignore all others. The use of this simple feature permitted a sixty per cent reduction in the time it took for the card reader to process the input data. The schedule of a course was not indicated on the card. Therefore updates to



Legend

1. Student's name
2. Class key
3. Student's id. no.
4. Class identifier
5. Student's rank on the class list
6. Teacher identifier
7. These 30 columns of data represent the essential part of the transaction data record to be transmitted on the card reader. The clock marks activate the card reader to read only the corresponding columns and ignore others.

Figure 33 This absentee card represents a transaction record in the data generation kit.

It contains 80 columns of punched data out of which 65 are interpreted on various parts of the card. The purpose of this is to permit easy visual identification and manipulation by teachers. Only 30 columns are essential of this transaction record to the computer. These are the class key, the student id., various matrix pointers, and a check sum. The purpose of this is to cut down input time.

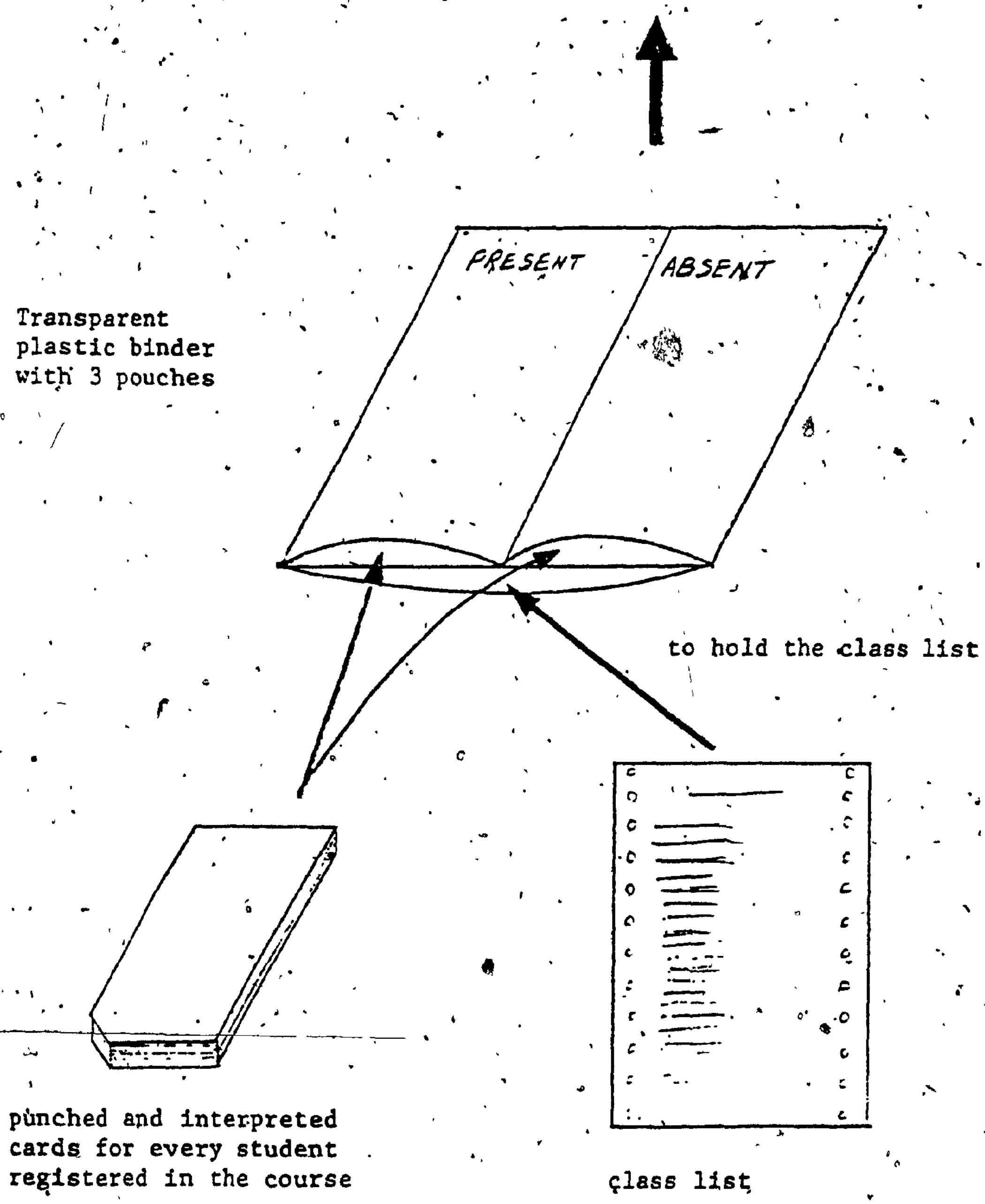


Figure 34 The teacher's data generation kit.

It is a plastic binder with 3 pouches, containing a class list, with a corresponding deck of punched and interpreted cards. At transaction data generation time the teacher simply selects the cards of those absent and inserts them into the indicated pouch. It is a simple, fast, transcription-free and error-free operation.

the master files reflecting a change in the course's schedule doesn't require the production of new student identification cards. These cards are punched once and remain valid throughout the entire academic year.

These attendance binders or data generation kits are picked up by the teachers each day in their mailbox room. The teacher carry these pouches around with them to class in order to record the attendance of each class that they teach.

The data collection routes must be well defined for this system to work. Figure 4 A, in chapter 1 gives an idea of just how extensive the data collection route is. The administration has to determine in advance who is going to collect the truancy transaction data from the classes at the beginning of every lecture. Since there are about 150 lectures at any one time which are widely scattered throughout different sections of the school the data collection route must be carefully planned. To meet the school's objectives all of this data must be collected whithin a 15 minutes period.

Before starting the regular run of the attendance monitoring subsystem we also have to carefully consider the length of the eventual output report. If it will be too long people will not read it, if it will be too short then truancy problems might escape the school administration's attention. In order to avoid this waste of human and technical efforts we have to specify some output selection parameters. Incorporated in the program at execution time these parameters will select for the output only those students whose truancy record match these parameters. This will be discribed in detail in sections 4.1 and 4.6 of this chapter.

3.2 Set up the interactive programs for the daily production cycle of the attendance system.

We have to choose and train an operator (23) for the regular running of the Attendance Monitoring subsystem's daily production cycle. That is we set up the on line programs to operate the production cycle. These are the daily initialization interactive program, data collection and verification pseudo-batch program, the data processing pseudo-batch program (which updates the data-base) and a procedure file to print the spooled output.

4. Production cycle or the regular daily run of the Attendance Monitoring subsystem (see table 5).

For each new day we have a new production cycle (as was described in chapter 3).

4.1 Daily data usage parameter definition. (43)

To start a new production cycle, the operator calls on the terminal a procedure file which invokes the interactive program ("CALL, ABSCOM"). This program sets up a new day's procedure. It clears the previous day's data buffer. The operator sets up the school day, the calendar day (Monday, Tuesday,...), the semester, the week. Then the operator enters output selection parameters such as no. of days missed during the current semester, total number of lectures missed during the semester, total number of lectures missed in any course during the semester. If any of these parameters are below the student's cumulated attendance record, the student's name will appear in report B at the end of the day. Error recovery is also possible in case of an absent-minded operator. This entire procedure takes no more than a few minutes (see Figure 35 for sample output).

```
4.  DO REGULAR RUN OF THE ATTENDANCE MONITORING SUBSYSTEM ($ ATS.)===$            < 4, 4, 4, 4, 4, 4, 5 >

    4.1= $ATS. DAILY DATA USAGE PARAMETER DEFINITION            ==   --THEN-GO-TO-CLUSTER-> 4
       4.1 IF  TASK IS ATTENDANCE MONITORING  AND IF
           ATTENDANCE MONITORING SUBSYSTEM IS SET UP  AND IF
           NOT  PRODUCTION CYCLE IS STARTED  AND IF
           USAGE PARAMETERS ARE DETERMINED
              A25> PERFORM DAILY SYSTEM INITIALIZATION

    4.2= $ATS. DAILY TRANSACTION DATA GENERATION                ==   --THEN-GO-TO-CLUSTER-> 4
       4.2 IF  TASK IS ATTENDANCE MONITORING  AND IF
           PRODUCTION CYCLE IS STARTED  AND IF
           PRODUCTION DAY IS INITIALIZED TO THE COMPUTER
              A26> PERFORM TRANSACTION DATA GENERATION

    4.3= $ATS. DAILY TRANSACTION DATA COLLECTION                ==   --THEN-GO-TO-CLUSTER-> 4
       4.3 IF  TASK IS ATTENDANCE MONITORING  AND IF
           PRODUCTION CYCLE IS STARTED  AND IF
           TRANSACTION DATA IS GENERATED
              A27> PERFORM TRANSACTION DATA COLLECTION

    4.4= $ATS. DAILY TRANSACTION DATA PROCESSING                ==   --THEN-GO-TO-CLUSTER-> 4
       4.4 IF  TASK IS ATTENDANCE MONITORING  AND IF
           PRODUCTION CYCLE IS STARTED  AND IF
           TRANSACTION DATA IS COLLECTED
              A28> PERFORM DATA PROCESSING (REPORTING AND STORAGE)
              A38> PRINT SPOOLED OUTPUT

    4.5= $ATS. DAILY DATA DISTRIBUTION                          ==   --THEN-GO-TO-CLUSTER-> 4
       4.5 IF  TASK IS ATTENDANCE MONITORING  AND IF
           PRODUCTION CYCLE IS STARTED  AND IF
           DATA IS PROCESSED
              A29> PERFORM DATA DISTRIBUTION

    4.6= $ATS. DAILY OUTPUT USAGE                               ==   --THEN-GO-TO-CLUSTFR-> 4
       4.6 IF  TASK IS ATTENDANCE MONITORING  AND IF
           PRODUCTION CYCLE IS STARTED  AND IF
           OUTPUT IS DISTRIBUTED  AND IF
           NOT  USAGE PARAMETERS ARE DETERMINED
              A30> VERIFY OUTPUT USAGE IN DECISION MAKING
              A20> DETERMINE DATA USAGE PARAMETERS
                                  V
```

Table 5 - Implementation plan to run the daily production cycle of the attendance system. (From ABL table SYSTALL).

```
CALL,ABSCOM
```

The operator calls the daily initialization procedure file.

```
ABSENCE CONTROL SYSTEM, DAILY INITIALIZATION.
VER. 14.28.22. 78/05/08.
FOR SCH.78/10/29

LAST ACCESS ...
    DATE ................   78/11/20.
    SCHOOL DAY...........   1
    CALANDER DAY.........   LUN
    WEEK ................   1
    SEMISTER ............   1

ENTER SCHOOL DAY IN I-2 FORMAT
? 02
ENTER CALANDER DAY IN I-1 FORMAT
? 2
ENTER WEEK IN I-1 FORMAT
? 1
ENTER SEMISTER IN I-1 FORMAT
? 1
```

The operator inputs the proper variable values so the system will be able to identify and select the desired schedules for the day's courses.

```
YOU ENTERED.......
    SCHOOL DAY...........   2
    CALANDER DAY.........   MAR
    WEEK ................   1
    SEMISTER ............   1

ENTER 'YES' OR 'NO'.
? YES
```

Error recovery possibility of the previous values if the input is "No".

```
OPTIONS SELECTIVES POUR L'IMPRESSION DU RAPPORT FINAL.

NO. DE PERIODES MANQUEES AU MINIMUM PENDANT LE MOIS COURANT ? (I2-FORMAT)
? 09

NO. DE PERIODES MANQUEES AU MINIMUM PERDANT LE MOIS COURANT DANS UNE MATIE
? 06

NO. DE JOURS MANQUES AU MINIMUM PENDANT LE MOIS COURANT? (I2-FORMAT)
? 04
```

The operator inputs the output selection parameter values which govern the length of Report "B".

```
SI LES ABSENCES DE L'ETUDIANT RENCONTRE UN DE CES
CRITERES SON NOM SERA SUR LA LISTE

YOU ENTERED ...........
NO. DE PERIODES MANQUEES AU MINIMUM PENDANT LE MOIS COURANT ..........
NO. DE PERIODES MANQUEES AU MINIMUM PENDANT LE MOIS COURANT DANSUN MATIERE
NO. DE JOURS MANQUES AU MINIMUM PENDANT LE MOIS COURANT ..................

ENTER 'YES' OR 'NO'.
? YES
```

Error recovery possibility of the previous values if the input is "No".

```
END DAILY INITIALIZATION.
NOTE. *** END OF ABSCOM ***
```



```
NOTE. *** END OF ABSCOM ***
```


Figure 35 Man-computer dialogue on the terminal during the daily initialization procedure of the production cycle.

4.2 Transaction data generation. (1, 7, 15, 32, 53)

Taking attendance with the pre-assembled data generation kits is a relatively quick and fool-proof procedure. Once in class, the teacher calls for attendance with the printed class list contained inside the 3 pouched plastic attendance binder. Any absent student has his attendance card removed from the deck of student cards corresponding to the class list. These cards are placed in a pouch reserved for absentees. Recording attendance for each class requires no more than a minute. Attendance error rates are drastically reduced because no transcription is required. The absence of a transcription process was a great asset because it is precisely in the beginning of a lecture that students are especially restless and most prone to distract the teacher thus causing errors. The only manual task required is the conscious but easy effort of separating student identification cards into two decks. The only errors observed to occur at this stage of the monitoring system were found to be malicious in nature. This way of generating transaction data presented another advantage in the form of a simplified error recovery. Should a student arrive late for class it takes no more than few seconds to retrieve his or her card from the absentee pouch. Errors resulting from late arrival of a student at a given lecture could also be quickly rectified by the school's attendance supervisors up to the point at which the data is actually entered into the machine. This facility improved the students punctuality. Late arrivals disliked having to go to their supervisor's office to retrieve their absentee card.

This manual operation was accepted with enthusiasm by the teachers. They were initially skeptical of the system's ability to adequately keep track of attendance in a very migratory student population. Teachers were perplexed that a set of cards when passed through the card-reader for example on Monday would register a missed lecture in period 5 while the same cards handed in on Tuesday would register a missed lecture in period 7. Only when it was understood that by cross-referencing the student file's course selection with the school's course master file, which contains time table information, the computer could find out exactly which course was given when. The system will thus be able to exactly register a missed lecture for a given course regardless of its daily schedule. The attendance system simply located the period that a given course is offered on a particular day of the week. Such a procedure will reject all cards handed in on the wrong day by mistake, if no lecture is scheduled for a particular course on the day in question. Such errors are flagged and a diagnostic message is subsequently sent to the system's operator at data collection time to avoid false positive errors. However the accuracy of the generated transaction data is also directly proportionnal to the accuracy of the data generation kit itself. In a highly dynamic environment where a lot of course changes occur throughout the year this could present a serious problem as far as false positive and false negative errors are concerned. The system's maintenance measures will be discussed in section 5.2 of this chapter.

4.3 Data collection.

Once the cards are separated and placed in their correct pouches the teachers send the entire folder to the supervisor's office. The folder protects the cards from handling usage. The supervisor takes only the cards placed in the absentee pouch from each folder. These card decks are then sent to the terminal room for input. It takes less than ten to fifteen minutes to prepare the entire school's transaction data file. The size of our transaction data file typically averaged between 300 to 500 records, or cards, per lecture period. Figure 4 A in chapter 1 illustrates the data collection route needed to collect the transaction file data from the school's 150 teachers. After the transaction decks are run through the card reader the card decks are sent back to the supervisor's office where they are inserted back into their proper binder. The class binders are then placed in the teacher mailboxes for use on the following day. This input procedure through the card reader provides transaction file data for only one lecture period. Thus this procedure must be repeated throughout the school day in order to avoid input bottlenecks with the card reader. Our implementation found it was best to repeat the input procedure for each lecture period of the day.(35) However the input process schedule may be modified to fit the daily activity of the school's clerk.

In practice it became important to insist that the attendance data for the first lecture period of each day was as complete and accurate as possible. This assured an accurate early morning truancy report.

The system's data base was updated only at the end of each school day. In this way the system was relatively insensitive to punctuality on the part of the school's teachers. This is an important factor within the school environment where a significant minority (10-15%) of teachers would often forget to send their data generation kit folders for processing until the afternoon. This tardiness did not effect the system's performance as long as the cards were processed on the same day.

The fragmented input procedure was implemented for three main reasons. Firstly it allowed the preparation of an accurate, early morning, truancy report based on attendance during the first lecture period. Secondly it eliminated input bottlenecks involved with card reading at the end of the school day. Finally it removed the need to either hire additional staff or pay school staff overtime in order to process bottlenecked data after school hours.

This fragmented input process has two parts. First we create a raw data-file, then we verify the data-file (see Figure 36).

In the implementation of this system we accelerated the process of card reading. While each card contains 80 columns of punched and interpreted data only 30 columns of selected values are read by the computer. This is achieved by the special card design which makes the card

```
NEW,MIKE
/TEXT
ENTER TEXT MODE

450441107604138l853 40   1 4688
4504411075532571243 40   1 4686
[illegible] 31   1 [illegible]
45451101745039O 187 34   1 1295
5780410373627031420 43   1 4881
[illegible] 43   1 [illegible]
5780410374522021050 46   1 4887
5780410374528611408 46   1 4885
[illegible] 34   1 [illegible]
4503220176606351146 34   1 7265
4503220176525001433 34   1 7266
[illegible] 34   1 [illegible]
4503220176044881531 34   1 7267
4503220175533181273 34   1 7265
[illegible] 37   1  619
57732001760O177  86 37   1  617
278400017602241 953 28   2 6098
[illegible] 28   2 6097
278400017706156 599 28   2 6094
2783000177037802075 28   1 6004
[illegible] 40   1 5756
2803200375541451631 31   1 5774
581320037651046 575 40   1 5750
1903200425537301445 25   1 6883

EXIT TEXT MODE.
NOSORT.
/PACK
PACK COMPLETE
/REPLACE,MIKE
/CALL,APKPLST(NOM=MIKE)
```

The operator starts a dummy file that is the temporary transaction data file name before the verification process.

The operator now inputs the transaction data generated by the teachers, by inputing the absentee cards through the card reader.

Only the first 30 columns of each card is read and transmitted to the computer.

The temporary transaction data is saved on disk.

The operator calls a procedure file for execution. This procedure will verify the temporary transaction data and collect if in the verified daily transaction file till the end of the day. This procedure will also produce a Report "A" at the end of the first lecture (see Figure 38)

Figure 36 Chain of procedures on the terminal at data collection time at the end of the first lecture.

reader ignore all but the specified 30 columns of data. This simple scheme cuts down card reading time to about a third of what it would take to read the entire card. This gain in speed is important in view of data traffic of 2000 to 3000 cards a day. Thus while all 80 columns are necessary for efficient card identification and subsequent handling by the teachers in class, only 30 columns contain information used by the computer to maintain attendance records. The 30 columns accessed by the computer contain the student id. number, the class key, and various pointers to set tables in the data-base and the programs. Columns 29 and 30 of the card are reserved for a check-sum of the first 28 fields. If they match, the data is correct. If they don't, data transmission error occured or the card has been damaged before reading. Incorporating such error detecting techniques saves us from having to check for the validity of these parameters at execution time. The schedule of the course on the card is verified to see if the given card was not handed in on the wrong day. The student's registration in the particular course is checked to see if he hasn't quit the course. Only some malicious errors have bypassed the system's security at this level. Error handling is enhanced by appropriate diagnostic messages (see Figures 37 A and 37 B for sample output). These are: transmission errors, wrong schedule for the course (cards handed in on the wrong day), outdated cards and cards registered twice. Transaction file verification consists of filtering the data through these checking routines. Only valid cards are transfered to the verified transaction file of the day. Error recovery is possible even after the cards have been entered in the system's transaction file. It is possible to use the system's text editor to delete records from the transaction file.

More drastic forms of error recovery involve aborting the transaction file and re-entering the entire day's data. Such error recovery is possible up to the point at which the data-base itself is updated. The data-base is updated at the end of each school day. This fragmented input procedure comes with 2 options: with or without reporting (except for error messages). Report A (see Figure 38) contains a listing of students absent from the given lecture with some other information from the data-base. The system produces such a report for each assistant principal. The day's first lecture is usually reported in full. However for the other lectures error messages are sufficient. Following the last lecture's fragmented input process we have the complete transaction

INFERIEURE

The operator records and saves the temporary transaction data file.

```
/REPLACE,MIKE
```

```
/CALL,ABKFNLT(NOM=MIKE)
```

The operator calls the transaction data collection and verification procedure with suppressed reporting except for error messages.

Each card image of the temporary transaction file is analysed. If no error is detected the card image is appended to the permanant transaction file "DABER" which contain the verified transaction data of the previous lectures. If an error is detected the transaction record is copied to the error listing with the appropriate error message.

Page heading

in the card deck

data transmission error

```
--SYSTEME INTEGRE DE GESTION PEDAGOGIQUE --      SCH.78/10/29
-- CONTROLE DES ABSENCES, DATA COLLECTION--      VER. 22.26.36. 78/11/13.
   ------ HORVATH & PALKIEWICZ -----              78/11/20.LUN, JOUR 1 SEM 1

AT LINE    0                                     * DATA-TRANS. ERROR.
AT LINE   15                                     * DATA-TRANS. ERROR.
AT LINE   61  4545110274517275 846 25   1 4806  WRONG SCHEDULE.
AT LINE   62  45451102770470619O9 25   1 480*  WRONG SCHEDULE.
AT LINE   63. 4545110273077671578 25   1 480*  WRONG SCHEDULE.
AT LINE   64  45451102745O756 365 25   1 480*  WRONG SCHEDULE.
EXIT. *** END OF ABKFNLT ***
```

transaction record

cards were handed in on the wrong day when no lecture was scheduled for this course.

Figure 37-A Chain of procedures on the terminal at data collection time at the end of the 2nd 3rd 4th 5th and the 6th lecture.

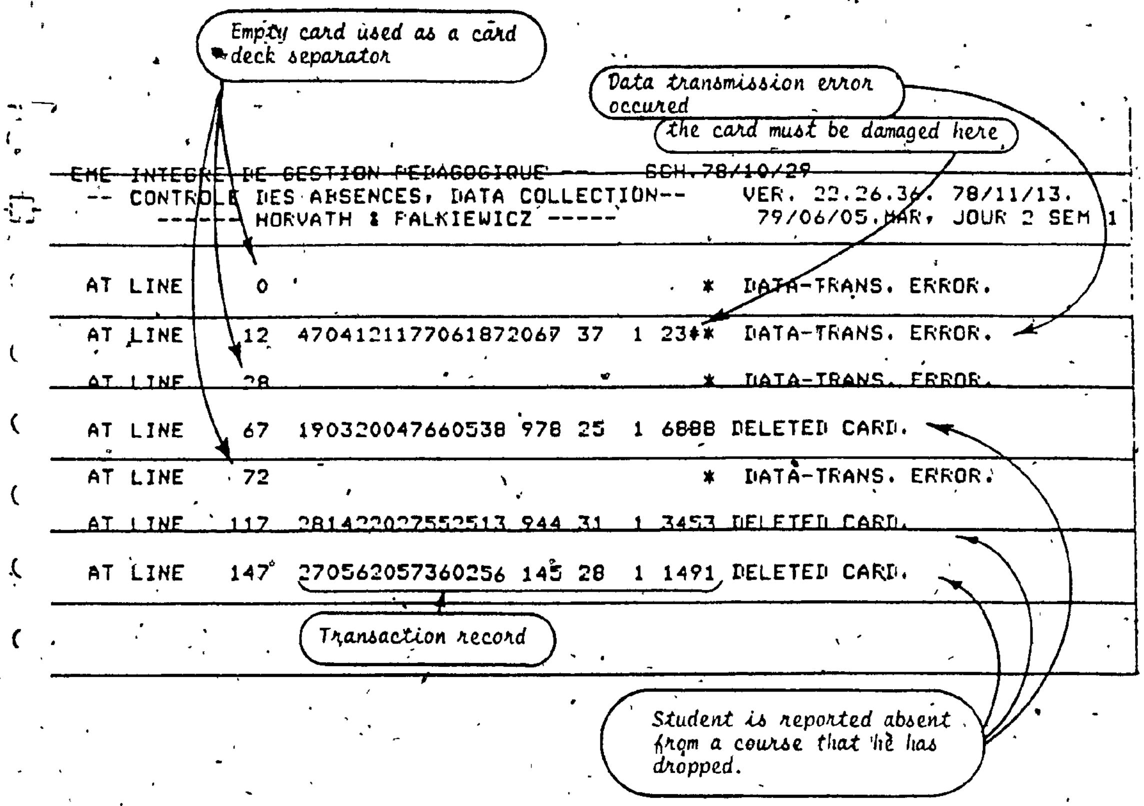


Figure 37-B Other examples of transaction data collection and verification.

```
--SYSTEME INTEGRE DE GESTION PEDAGOGIQUE --          SCH.78/10/29
-- CONTROLE DES ABSENCES, DATA COLLECTION--          VER. 11.46.15. 78/11/14. DIR=  9
    ------ HORVATH & PALKIEWICZ ----                 79/06/08.VEN, JOUR 5 SEM 1 SIM 2
7218273 ROY  MICHEL             653-3393  06/06   86/163  GEO-47051203.3/JOUR
7361679 FONTAINE  JOHANNE       653-7693  06/06   92/229  MAT-28152205.3/JOUR
7403502 GAUDET  DENIS           656-2341  06/06   68/110  REL-48042105.3/JOUR
7450624 BOULANGER  ODETTE       653-3134  06/05   39/109  FRA-58052105.3/JOUR
7451812 FRADETTE  LINDA         653-0495  06/06   78/152  FRA-58052105.3/JOUR
7453756 PERRON  ANDRE           653-3704  06/04   86/136  HIS-47541212.3/JOUR
7453890 POTVIN  RICHARD         658-1944  06/06  102/264  GEO-47051203.3/JOUR
7454537 THIBAULT  SANDIA        653-4747  06/01   39/ 86  REL-48042105.3/JOUR
7454683 VACHON  CLEMENT         653-2353  06/06   64/132  ANG-57852105.3/JOUR
7531346 BRASSARD  DANIELLE      658-1502  06/06   61/125  HIS-47541212.3/JOUR
7601933 GODBOUT  STEPHANE       653-2794  06/05   41/ 55  MAT-28142214.3/JOUR
7800405 JOSEPH  MICHELE         676-9611  06/01   24/ 24  FRA-58052105.3/JOUR
```

<u>Legend</u>

1	*Heading*	8	*Student id. number*
2	*Date*	9	*Student's name*
3	*Calendar day*	10	*Student's home phone number*
4	*School day*	11	*Last date missed from school (month, day)*
5	*Week*	12	*Total number of partially or entierely missed days since the beginning of the year*
6	*Semester*	13	*Total number of lectures missed since the beginning of the year*
7	*Target user (director 9 for this listing)*	14	*Course missed*

Figure 38 This is a page from Report "A" or the partial output of the fragmented data collection procedure for the first lecture. It contains a list of students absent from the first lecture. Each director has his own report "A" for the students under his supervision. The report "A" reaches its destination within an hour at the beginning of the first lecture and it contains enough information to permit to be phoned.

file for the day and we are ready for transaction data processing.

4.4 Transaction data processing. (18)

This process involves: updating the data-base with the transaction file, storage of each student's truancy record and synthesis of a truancy report B for the day. In this process each student's record is retrieved and updated as follows:

1) The student record's course selection field is used to store the last date on which student missed a lecture in that particular course.

The course selection field in the student's master file in overwritten with the day's date:

2) The field which stores the accumulated total number of lectures missed in that course is incremented.

3) The field of the student's record which is used to store the number of days partially or entirely missed is incremented. (A logical check prevents incrementation of this sum if this total has already been updated for that day).

In addition to these updates of the student master file the class attendance record is updated with information on the total number of students who missed that particular lecture. In addition to this simple book-keeping the system computes and stores a 7-bit binary vector for every student on each day of the school year. The value of i-th bit of this vector allows one to determine wether or not the student missed his i-th lecture during that day. For example if the student was absent from lectures 3 and 5 the third and fifth bits are given a value of "1" while the others remain "0". This binary vector when used with the student's course schedule on a given day allows one to determine the lecture by lecture attendance pattern of any student throughout the year. These binary vectors are stored daily in a section of each student's record. (37)

The daily operation of the data processing phase is simple. When all transaction data has been collected in the validated transaction file, the operator calls a procedure which will submit a batch job to the system. He will enter "call, absoir" on the terminal (see Figure 39). He will receive a short message from the system on his terminal but the actual output from this procedure will be spooled. The operator may then go home. There is no need to wait for some lengthy output which will be only used on the next day. This whole operation lasts no more than a few minutes and can be completed during

At the end of the day when all transaction data have been verified and collected, the operator calls a special procedure file for execution.
This will update the data-base and will produce a Report "B" which is a daily truancy report summary.

```
CALL,ABSOIR
  13.29.07.AH1ABYV
 NOTE.  END ABSOIR
 *EXIT.
 /
```

Submitting the job is on line.
Execution is a batch job off line.

```
IWXMESS.  *** TRAVAIL COMPLET ***
```

Successfull submitting will generate this message to the operator. After this he may go home as the output from this execution is spooled untill the next morning.

The output is then listed in the morning on the terminal.

Figure 39 Daily transaction data processing procedure on the terminal.

regular school hours. On the following morning the operator retrieves this spooled output from disk-storage by entering "call,ablist" on the terminal (see Figure 40). The output, or report B, is produced for each assistant principal. Report B includes an error listing of outdated cards (see Figure 40 for sample output), a 3 line long mini-report for each absent student under the assistant principal's responsability (see Figures 42 A and 42 B), and an overall, or statistical, view of the truancy level for that day in the school. This report is prepared for the principal (see Figure 41). Report B lists only the names of those students whose daily lecture absences surpass the level initially set by the system's output selection parameters. The objective of this report was to initiate a face to face meeting between the "problem" student and the assistant principal who was responsable for his or her supervision. We found that no matter how many students were absent, the assistant principal could not meet more than 10-15 students a day. The judicious use of the system's output selection parameters made it possible to select the 10-20 worst truancy cases for the focused attention of the assistant principals. This feature enabled them to more effectively use their time and cut down substantially

```
CALL:ABSLIST
```

Next morning the operator calls a procedure file to list the spooled output from the previous day's run.

LISTING no 1 contains error messages to each director to take out outdated absentee cards from the corresponding data generation kits. These cards do not produce errors because they are rejected before the data-base is updated.

```
--SYSTEME INTEGRE DE GESTION PEDAGOGIQUE --
-- CONTROLE DES ABSENCES, RAPPORT FINAL --        DIR= 100
+----- HORVATH & PALKIEWICZ ----

CARTE A ENLEVER ........

ART-57113001  1102.CAPUANO.MICHEL   7750224  BEAUDOIN  GILLES
FRA-58113001  1104.CHAREST.ADRIEN   7750224  BEAUDOIN  GILLES
ANG-57721001  1105.COOSSA.V.        7804974  LEVESQUE  SYLVIE
ANG-57721001  1105.COOSSA.V.        7804834  ADAM  YVES
ACT-99922002  1109.HAMELIN.CLAUDE   2205922  CHALIFOUR  JEAN-CLAU
MUS-57521001  1111.LABELLE.DENIS    7654048  VERVILLE  GILLES
GYM-19312003  1332.VANIER.CAROLE    7801510  BEAUDRY  FRANCINE
GYM-19322003  1332.VANIER.CAROLE    7804974  LEVESQUE  SYLVIE
```

Target user director no 10

Class pouch identifier — *Teacher identifier* — *Student identifier*

Figure 40 Daily transaction data processing output Report "B" part 1.

This part contains an error listing to each director to warn them to take out outdated cards from the data generation kits.

```
--SYSTEME INTEGRE DE GESTION PEDAGOGIQUE --    SCH.70/10/29
--CONTROLE DES ABSENCES, RAPPORT FINAL   --    VER. 17.48.10. 70/11/14. EXE#   4  DIR=
          HORVATH & FALKIEWICZ                 70/11/17.VEN. JOUR-5-SCH-1-SHIR-1-PAGE    1

  --- SUMMARY ---

   827 CARD(S) READ.    406 ABSENT FROM SCHOOL.

   227 STUDENT(S) ABSENT FOR  1 LECTURE(S).
    65 STUDENT(S) ABSENT FOR  2 LECTURE(S).
    30 STUDENT(S) ABSENT FOR  3 LECTURE(S).
    37 STUDENT(S) ABSENT FOR  4 LECTURE(S).
    22 STUDENT(S) ABSENT FOR  5 LECTURE(S).
    15 STUDENT(S) ABSENT FOR  6 LECTURE(S).
     0 STUDENT(S) ABSENT FOR  7 LECTURE(S).
   176 STUDENT(S) ABSENT FROM LECTURE  1.
   133 STUDENT(S) ABSENT FROM LECTURE  2.
   122 STUDENT(S) ABSENT FROM LECTURE  3.
    95 STUDENT(S) ABSENT FROM LECTURE  4.
     0 STUDENT(S) ABSENT FROM LECTURE  5.
   140 STUDENT(S) ABSENT FROM LECTURE  6.
   155 STUDENT(S) ABSENT FROM LECTURE  7.
```

827 transaction records

But only 406 students missed at least one lecture

Students playing hookey

Probably artefact due to negligence by some teachers in handing in his cards

Only 15 students were reported absent all day.

Late arrivals to school

Students who skip the last lecture

Figure 41 Listing #2 of Report "B". Daily statistics for each vice principal.

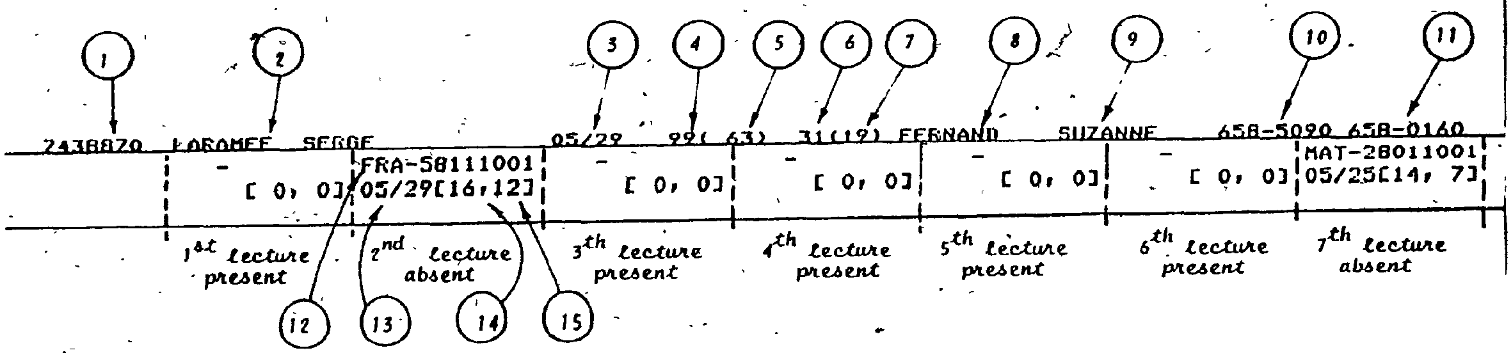
This one is for the principal. It gives the overall picture of the whole school. Each vice principal has an identical statistical report for the students under his supervision.

heading

execution nc

```
--SYSTEME INTEGRE DE GESTION PEDAGOGIQUE --      SCH.78/10/29
-- CONTROLE DES ABSENCES, RAPPORT FINAL  --      VER. 15.18.51. 79/04/05. EXE# 108  DIR=  1
   ------ HORVATH & PALKIEWICZ ----              79/05/31.JEU, JOUR 4 SEM 1 SMTR 1 PAGE  1

7750338  BELIVEAU  DENIS            05/29  120( 66)  43(22) ONIL        ESTELLA     678-7415 678-7415
             -           FRA-58132003
               [ 0, 0] 05/10[27,13]

7751153  CHARRON  ALAIN             05/30  133( 65)  52(27) REMI        MARIE-PAUL  658-8007 658-8007
             -           FRA-58132003
               [ 0, 0] 05/29[27,13]

7851211  COMTOIS  SONIA             05/30  180( 82)  54(25)             BLANCHE     678-2481 678-2481
         CUR-48312001      -        ACT-99912002     -           -          GYM-19312003
         05/30[11, 7]      [ 0, 0]  05/29[30,13]     [ 0, 0]     [ 0, 0] 05/29[ 5, 4]

7438870  LARAMEE  SERGE             05/29   99( 63)  31(19) FERNAND     SUZANNE     658-5090 658-0160
             -           FRA-58111001   -            -            -           -     MAT-28011001
               [ 0, 0] 05/29[16,12]     [ 0, 0]      [ 0, 0]      [ 0, 0]     [ 0, 0] 05/25[14, 7]

7754498  MOORE  DANY                05/30  200( 70)  58(25)             MONIQUE     678-5681 678-5681
         CUR-48312001 MAT-28012001 ACT-99912002
         05/30[12, 8] 05/29[35,14] 05/29[29,10]

7703346  ROBERGE  NINON             05/30  133( 82)  54(30)             RITA        658-4201 658-4201
             -            -            -        GYM-19322003
             [ 0, 0]      [ 0, 0]      [ 0, 0] 05/07[11, 6]
```

Figure 42-A   This is listing#3 of Report "B". It is a synthesis of the daily truancy information about students whose overall attendance record fall beyond the specified output selection parameters. Each assistant director has a listing like this for the students under his supervision. (For more information see Figure 42-B). This report reaches its target user by 10 o'clock the following day.



<u>Legend</u>

1. Student's id. number
2. Student's name
3. Last date missed
4. Total number of lectures missed since the beginning of the year
5. Total number of days missed since the beginning of the year
6. Total number of lectures missed since the beginning of the semester
7. Total number of days missed since the beginning of the semester
8. Father's first name
9. Mother's first name
10. Home phone number
11. Emergency phone number
12. Identifier of the missed lecture
13. Last date this course was missed
14. Total number of times this lecture was missed since the beginning of the year
15. Total number of times this lecture was missed since the beginning of the semester

Figure 42 B Detailed explanation of the information on Report "B".

on the data processing costs. Report B always reaches the assistant principal within 24 hours after the beginning of a production cycle.

4.5 Data distribution (47)

Data distribution is a complex task. It involves distributing the output listing and the data generation kits used for the system's card input. We had to meet hard deadlines. The absentee cards must be returned to their respective data generation folders and these folders placed in the teacher's mailboxes by 4 PM to enable the next morning's production cycle to begin. This deadline is one of the reasons why we opted for the fragmented data input process described earlier in section 4.3. This fragmented process allows for input data and information distribution in a continuous manner during the whole day. We steered clear from earlier unsuccesfull methods of output distribution. These attempts provided large impressive looking reports which were printed in alphabetical order and deposited for consultation in the records office. These reports were inconvenient and were consulted only in extreme cases.

We adopted the design principle that information had to be handed out to the proper decision makers in readily usable form within a useful time delay. For this reason we used a simple standard format for our listings (report A and B). Each student has a personal supervisor. This supervisor is an assistant principal. Each vice principal has about 300 students under his or her supervision. Each student record contains a number which identifies the student's supervisor. Thus the system's output may be directly forwarded to the student's supervisor. This facility is analogous to a computerized mailbox. Should a student change his supervisor, the secretary, by changing the supervisor's number on the student record, will automatically channel all output concerning this student to the new supervisor. Thus the system output is sequentially sorted before printing and it needs no manual sorting afterwards. With this method we could deliver the morning listing (report A) before the day's first lecture had ended. Report B followed within 24 hours after the production cycle has started. Once the school administration receive report A they phone the parents of truant students. In more serious cases the parents are invited to the school and given a computer report on the spot. In all cases the system is used to enhance a highly personnalized face to face contact based on accurate information.

The system was designed to be always able to give comprehensive information should the need arise. To be able to help in all kinds of situations (parental visit, truant student counselling, management meeting, teacher-principal relations, etc) an elaborate interactive query subsystem was developed. This system is discussed in detail in section 6 of this chapter.

4.6 Data and information usage. (2, 44, 46)

Data and information usage study is the most important consequence of the production cycle. This feature permits successful refinement of the modelling and implementation process described in this thesis. While the initial model and implementation of the system is based on assumptions in a very hypothetical way, we may define it in terms of its success in providing and handling the system's data. If we define data usage in terms of the portion of the computer processed output that is actually used in decision making, then it is possible to isolate three major factors effecting this utilization. These factors are: volume, accuracy and delays. If we provide too much information we not only waste precious resources but we saturate the user. If the information is not accurate, then it's not dependable. Furthermore truancy information is really useful for only a short period of time. By studying the rate of use of the various computer reports, we can detect major oversights and flaws in the logical design of the system long before the user becomes frustrated by its shortcomings and rejects the system altogether. In order to circumvent these problems we paid attention to the delay and accuracy problems in the logical design of the system. Computer assistance was introduced into all steps of the production cycle to help alleviate such potential difficulties. Data usage studies permitted us to closely tune the reporting part of the system to the real and often unexpressed needs of the user in his decision making task. In this way we can optimize output usage while minimizing output volume: such optimization also minimizes data processing expenses. The output selection parameter feature was incorporated into the reporting part of the system to facilitate this process. While output usage differs from one output to the other, we always have to establish the manner in which such reports will receive attention and use. The morning report (report A for the first lecture) was useful if most parents affected by it got a phone call from school. Initially we had thought that report A for the day's subsequent lectures would also be used in a similar way. We were wrong and report A on subsequent lectures became optional. The daily report B was initially quite voluminous. Its volume resulted from the initial requirements of the school administration. However after the initial enthusiasm dissipated, it was established that report B was simply too large to get the attention it required. Report B was to aid the assistant principals in isolating and meeting the persistently truant student. By introducing output selection parameters the report's volume dropped by about 90%. Concomitantly the number of counselling sessions with truant students increased sharply. The computer outputs provided by the interactive query subsystem also add evidence, objectivity and weight to such encounters. The personal interactions between parents and the school's administration, students and the parents, student and supervisor, as well as teachers and the administration all benefited from this feature. We can evaluate the functional importance of these reports by the number of times they were actually used. Ironically our studies show that the batch reporting at the end of each semester requested by the administration got the least attention.

This last point illustrates another fact which makes data usage studies necessary. Users sometimes request information they won't use or need. Sometimes they will also fail to request information that they do need.

5. <u>Maintaining the system's data accuracy (see table 6)</u>.

The accuracy of the system's routine performance is directly dependent on that of its data base and its data generation kits. Both the system's data base and the teacher's data generation kits had to be accurately and continually updated to maintain the system utility in a highly dynamic environment and to avoid false positif or false negatif errors.

5.1 Updating of non attendance related data.

The interactive updating subsystem has been discussed earlier in section 1 of this chapter. In the update subsystem any authorized edits to the system's data base were immediately cross referenced and updated on the class and student master files. In this way it was possible to accurately track any changes in a student's schedule. Cross referencing the transaction file record with the student's course registration area detected inconsistencies in the transaction file records that resulted from course changes. This detection process effectively eliminated false positive truancy errors encountered whenever a student modified his course schedule. This technique prevented penalizing the student for missing lectures in a course which he or she had officially dropped.

5.2 Updating of attendance related data.

The date generation kits were updated in a semi-automatic manner. The act of registering a student for a course automatically formatted his or her data generation kit card, or ticket. The ticket's "card image" was then automatically appendended to a special card image file. This file would be periodically punched and interpreted on the specially designed computer cards (stored at the computer center at Concordia University). Damaged or lost absentee cards could be replaced by a single command in the class query subsystem described in section 6 of this chapter.

For example the interactive commands:

```
PUNCHS, Class-key, student-no

    will "punch" replacement absentee cards
    for the student ( = student-no )
    in the class     ( = class-key  )

PUNCH, Class-key

    will "punch" replacement absentee cards
    for the whole class.
```

These commands were developed to give the school's staff the ability to easily maintain the school's data

```
5. MAINTAIN THE EXACTNESS OF THE SYSTEMS DATA ( $ UPS. )        ===$                      < 4, 4, 6 >

   5.1= UPDATE NON ATTENDANCE RELATED DATA                       ==    --THEN-GO-TO-CLUSTER-> 4
      5.1 IF  TASK IS UPDATING THE DATA-BASE  AND IF
            CHANGE INVOLVES NON $ATS. RELATED DATA
               A10>  UPDATE COURSE MASTER FILE INTERACTIVELY
               A11>  UPDATE STUDENT MASTER FILE INTERACTIVELY
               A12>  UPDATE CLASS MASTER FILE INTERACTIVELY

   5.2= UPDATE ATTENDANCE RELATED DATA                           ==    --THEN-GO-TO-CLUSTER-> 4
      5.2 IF  TASK IS UPDATING THE DATA-BASE  AND IF
            CHANGE INVOLVES $ATS. RELATED DATA
               A10>  UPDATE COURSE MASTER FILE INTERACTIVELY
               A11>  UPDATE STUDENT MASTER FILE INTERACTIVELY
               A12>  UPDATE CLASS MASTER FILE INTERACTIVELY
               A33>  QUERY  COURSE MASTER FILE INTERACTIVELY
               A31>  PUNCH NEW $ ABS-CARDS
               A32>  UPDATE TEACHERS DATA GENERATION KITS
```

Table 6 – Implementation plan to maintain the system's data accuracy (From ABL table SYSTALL).

generation kits in the face of theft, loss and damage. As we have seen in the above example, the command "PUNCH" in the interactive utility program could be used to replace the data generation kit for the whole class when given the class key. The conversational utility program writes student cards directly to the card image file. The card image file has to be punched every two weeks to maintain the system's false negative error rate under a 10% threshold.

The system's flexibility is a function of cross referencing techniques. For example, if for some reason the course's schedule, the room number, or even the teacher has been changed, only the course record of the course master file has to be updated. Cross referencing the transaction data with the updated course record at processing time permits the new updated information to be printed on all reports without modifying the class's data generation kit.

```
6. INFORMATION EXTRACTING AND REPORTING ( # QUS. )            ===          < 4, 4, 4, 4, 4, 4 >

   6.1= QUERY  COURSE MASTER FILE INTERACTIVELY               ==   --THEN-GO-TO-CLUSTER-> 4
        6.1 IF  TASK IS QUERY  THE DATA-BASE  AND IF
              COURSE INFORMATION IS NEEDED  AND IF
              SMALL AMOUNT OF OUTPUT IS NEEDED
                A33>  QUERY  COURSE MASTER FILE INTERACTIVELY

   6.2= QUERY  STUDENT MASTER FILE INTERACTIVELY              ==   --THEN-GO-TO-CLUSTER-> 4
        6.2 IF  TASK IS QUERY  THE DATA-BASE  AND IF
              STUDENT INFORMATION IS NEEDED  AND IF
              SMALL AMOUNT OF OUTPUT IS NEEDED
                A34>  QUERY  STUDENT MASTER FILE INTERACTIVELY

   6.3= QUERY  CLASS MASTER FILE INTERACTIVELY                ==   --THEN-GO-TO-CLUSTER-> 4
        6.3 IF  TASK IS QUERY  THE DATA-BASE  AND IF
              CLASS INFORMATION IS NEEDED  AND IF
              SMALL AMOUNT OF OUTPUT IS NEEDED
                A35>  QUERY  CLASS MASTER FILE INTERACTIVELY

   6.4= QUERY  COURSE MASTER FILE PSEUDOBATCH                 ==   --THEN-GO-TO-CLUSTER-> 4
        6.4 IF  TASK IS QUERY  THE DATA-BASE  AND IF
              COURSE INFORMATION IS NEEDED  AND IF
              LARGE AMOUNT OF OUTPUT IS NEEDED
                A36>  WRITE A FILE IN COURSE MASTER FILE QUERY  LANGUAGE
                A37>  QUERY  COURSE MASTER FILE PSEUDOBATCH MODE
                A38>  PRINT SPOOLED OUTPUT

   6.5= QUERY  STUDENT MASTER FILE PSEUDOBATCH                ==   --THEN-GO-TO-CLUSTER-> 4
        6.5 IF  TASK IS QUERY  THE DATA-BASE  AND IF
              STUDENT INFORMATION IS NEEDED  AND IF
              LARGE AMOUNT OF OUTPUT IS NEEDED
                A39>  WRITE A FILE IN STUDENT MASTER FILE QUERY  LANGUAGE
                A40>  QUERY  STUDENT MASTER FILE PSEUDOBATCH MODE
                A38>  PRINT SPOOLED OUTPUT

   6.6= QUERY  CLASS MASTER FILE PSEUDOBATCH                  ==   --THEN-GO-TO-CLUSTER-> 4
        6.6 IF  TASK IS QUERY  THE DATA-BASE  AND IF
              CLASS INFORMATION IS NEEDED  AND IF
              LARGE AMOUNT OF OUTPUT IS NEEDED
                A41>  WRITE A FILE IN CLASS MASTER FILE QUERY  LANGUAGE
                A42>  QUERY  CLASS MASTER FILE PSEUDOBATCH MODE
                A38>  PRINT SPOOLED OUTPUT
```

Table 7 - Implementation plan of information extracting from the data-base. (From ABL table SYSTALL)

6. The query subsystem (see table 7) (3, 49, 43,50, 18)

The query subsystem was subdivided into three query languages to suit the query requirements and access organization criteria of each of the system's master files (see Figures 20 et 18 in chapter 3 for more details). The logical access and security levels of the data-base (9) has been defined earlier (see Figure 18). The query subsystem should permit on authorized user to quickly obtain comprehensive information regarding students, classes and courses. To do this in a simple manner, we developed three conversational query languages. Programs supporting these languages have similar structures. After identifying the key of the record to be displayed on the terminal, the record would be retrieved from the random access file into a buffer area in central memory. Then a menu of commands is offered to the operator. Upon entering any command, a defined set of record items would be displayed according to a specific format. These programs provided a man-machine interface that was essential to our application. They formatted and displayed data in a way that was understandable and useful to the system's users.

By far the most widely used of these programs was the student master file query subsystem. It had the following commands and display functions:

ID, student-key	:	retrieve student record from the file (no = student key) (see Figure 43)
PERSONNEL	:	display all personal information items (see Figure 43)
COURS	:	display all registered courses of the student (see Figure 44)
HORAIRE *	:	display all registered courses of the student with the full time table schedule information (see Figure 45)
ABSENCE	:	display attendance record in all registered courses of the student (see Figure 46)
DETAIL	:	display all the student's absences in a chronological order (see Figure 47)
FIN	:	terminate the query session

---

* The commands are in French.

```
ID.7601989
7601989  KALBAKJI   RITA                 SUPPORT = 600

ID,PERSONNEL,HORAIRE,COURS,ABSENCE,DETAIL,NOTES,FIN.
?

PERSONNEL

  RENSEIGNEMENTS PERSONNELS%
  --------------------------

  11...... NO. DE FICHE%.................7601989
  12...... QUE. CODE%....................KALR22586408
  13...... NOM ET PRENOM%................KALBAKJI   RITA
  14...... DATE DE NAISSANCE,AGE,SEX%..22086     F
  21..26.. ADRESSE%..................... 5636    TERRASSE MORENC
                                          ST-HUBERT
  27..28.. TEL., TEL. D"URGENCE%........676-9159, 676-9159
  31...... PRENOM DU PERE%...............JOSEPH
  32...... OCCUPATION%...................MENUISIER
  33...... PRENOM DE LA MERE%............MARIE
  34...... OCCUPATION%...................
  35...... NO.D"ENFANTS DANS LA FAMILLE% 04
  36...... RANG DANS LA FAMILLE%........04
ID,PERSONNEL,HORAIRE,COURS,ABSENCE,DETAIL,NOTES,FIN.
?
```

After entering this command the student record is retrieved and a menu of display options is available.

Legend

1. *Commands given by the operator*
2. *Menu of on line commands available*
3. *Student's supervisor*
4. *All personnal items of the student*
5. *Student's name and id.*

Figure 43 Man-computer dialogue while using the student master file query language to display a student's personnal information.

```
COURS

COURSE INFORMATION.
--------------------
GYM-19242107    3
CHI-27046201    3
MAT-28143202    3
PHY-29042207    3
HIS-47544202    3
REL-48042108    3
ANG-57842110    3
ANG-57892110    3
FRA-58043103    3
FRA-58093103    3

IO,PERSONNEL,HORAIRE,COURS,ABSENCE,DETAIL,NOTES,FIN.
?
```

Once the student's record is retrived from disk to central memory we can display it in a number of different ways.

Legend

1. *On line command given to display the student's registered courses*
2. *Course information*
3. *3 letter course mnemonique identifier*
4. *8 digit course key*
5. *Semester identifier*
   - *1 = 1st semester*
   - *2 = 2nd semester*
   - *3 = 1st and 2nd semester*
6. *Menu of commands*
7. *System requests next command*

Figure 44 Man-computer dialogue using the student master file query language to display a student's course registration area.

The student's record is previously retrieved from disk to central memory.

HORAIRE ← 1

—SMTR-- I --SIM-- I		JOUR 1	JOUR 2	JOUR 3	JOUR 4	JOUR 5
GYM-19242107.3/J 1323.DIAZ-ARENAS.M.	(D143)	6			3	
CHI-27046201.3/J 3202.CHARTRAND.G.	(C215)	1	6	2		3
MAT-28143202.3/J 3215.ST-PIERRE.G.	(B203)		4	1	6	2
PHY-29042207.3/J 3203.CHAURETTE.RENE	(C210)	2		3	7	4
HIS-47541202.3/J 3212.LANDRY.ESTELLE	(B216)	4	1		2	6
REL-48042108.3/J 3102.BERGERON.M.	(A211)		2			7
ANG-57842110.3/J 3214.ST-JACQUES.J.	(A204)	7	3		4	1
ANG-57892110.3/J 3214.ST-JACQUES.J.	(A204)	7	3		4	1
FRA-58043103.3/J 3204.CLERMONT.M.	(A215)	3	7	4	1	
FRA-58093103.3/J 3204.CLERMONT.M.	(A215)	3	7	4	1	

ID,PERSONNEL,HORAIRE,COURS,ABSENCE,DETAIL,NOTES,FIN.
?

1 *on line command to display the schedule*
2 *semester*
3 *week*
4 *Monday*
5 *Tuesday*
6 *Wednesday*
7 *Thursday*
8 *Friday*
9 *lecture number during the particular days*
10 *course identifier*
11 *teacher identifier*
12 *room number*

Figure 45 Man-computer dialogue while using the student master file query language to display the student's course schedule for the week.

```
ABSENCE

DERN.ABSENCE LE  04/09, NO. TOTAL DE JOURS  48, NO.TOTAL DE PERIODES    86

 GYM-19242107   3      03/29    10
 CHI-27046201   3      03/28    13
 MAT-28143202   3      01/16     3
 PHY-29042207   3      04/02    11
 HIS-47541202   3      04/09     8
 REL-48042108   3      03/16     8
 ANG-57842110   3      04/03    16
 ANG-57892110   3                0
 FRA-58043103   3      02/27     6
 FRA-58093103   3                0
ID,PERSONNEL,HORAIRE,COURS,ABSENCE,DETAIL,NOTES,FIN.
?
```

Legend

1. *On line command given by the operator*
2. *Last date the student was reported absent*
3. *Total number of days where at least one lecture was missed*
4. *Total number of lectures missed by the student in all his courses since the beginning of the year*
5. *Course identifier*
6. *Last date this course was missed*
7. *Total number of times this course was missed*

Figure 46 Man-computer dialogue while using the student master file query language to display the student's course related truancy information (totals and subtotals).

The command "DETAIL" unpacks the "binary vector" stored in the student record and displays the overall absences of the student in a chronological ordered calendar. This particular feature is quite unique. Most attendance monitoring systems (28, 1, 7, 15, 32, 55, 14) increment only totals or subtotals of lectures missed. This particular form in which the "binary vectors" are displayed can allow one to establish a pattern in the absences of individual students. This feature permitted more efficient student counselling. For example the study of the attendance record of nearly 300 drop-outs of our database was used to establish a progressive pattern in the absences of these students. This information could be used to help problem students.

Each command provides the user with specific information which would be useful to him or her in a particular situation. The commands are easy to remember by any user because they are formed from a small vocabulary of meaningful words. The secretaries at the records office, the assistant principal and their secretaries and individual teachers have direct access to the system to provide them with appropriate information needed to face their students.

The class master file query subsystem had the

```
ID,PERSONNEL,HORAIRE,COURS,ABSENCE,DETAIL,NOTES,FIN.
? DETAIL,79/05/21,79/06/01

DETAIL ABSENCE DISPLAY FROM  79/05/22 TO 79/06/01
I----------I----------I----------I----------I----------I----------I
I          I   -LUN-  I   -MAR-  I   -MER-  I   -JUE-  I   -VEN-  I
I----------I----------I----------I----------I----------I----------I
I 79/05/21 I          I    1     I   12     I          I    2     I
I----------I----------I----------I----------I----------I----------I
I 79/05/28 I    4     I    1     I    2     I    2     I    2     I
I----------I----------I----------I----------I----------I----------I
```

<u>Legend</u>

1. *Interactive command to display detailed absentee information from 79/05/21 to 79/ 06/01.*
2. *Report line corresponding to the week of 79/05/21*
3. *Monday 79/05/21. Student was present*
4. *Tuesday 79/05/22. Student missed 1^{st} lecture*
5. *Wednesday 79/05/23. Student missed 1^{st} and 2^{nd} lecture*
6. *Thursday 79/05/24. Student was present*
7. *Friday 79/05/25. Student missed the 2^{nd} lecture*
8. *Report line corresponding to the week of 79/05/28*
9. *Monday 70/05/28. Student missed the 4^{th} lecture*

Figure 47-A Man-computer dialogue while using the student master file query language to display the student's detailed absentee information.

The command detail will unpack part of the binary vector (where the student's absences are stored) from a certain date to a certain date. It will then display all lectures missed in a chronological order corresponding to this period.

```
ID,PERSONNEL,HORAIRE,COURS,ABSENCE,DETAIL,NOTES,FIN.
?,DETAIL,79/04/02,79/06/01
```

DETAIL ABSENCE DISPLAY FROM 79/04/02 TO 79/06/01

	-LUN-	-MAR-	-MER-	-JUE-	-VEN-
79/04/02	34	1	12	2	12 6
79/04/09	4	....	....	....	....
79/04/16	....	1	12	1	2 6
79/04/23	1 4	1 6	....	2	2
79/04/30	12 4	1	12	2	2 6
79/05/07		1	1		....
79/05/14	4	1	12	2	2
79/05/21	....	1	12	[illegible]	2
79/05/28	4	1	2	2	2

Why is this student always absent

on Mondays from the 4th lecture

on Tuesdays from the 1st lecture

on Wednesdays from the 1st and 2nd lectures

on Thursdays from the 2nd lecture

on Fridays from the 2nd lecture

Figure 47-B The interactive command "DETAIL" helps the student's supervisor to detect the pattern of the student's absences. It subsequently permits easier studies and helps to establish the possible reasons for various truancy problems.

following menu of display commands and display functions

COPYCL, class-key	:	retrieve record corresponding to the class-key and print a full class list (see Figure 48)
COPYMR, class-key	:	retrieve record corresponding to the class-key and print a full class list with the cumulated attendance record of each student during the session (see Figure 49)
PUNCHS, class-key student-no	:	"punch" replacement absentee card for the student (= student-no) in the class (= class-key)
PUNCH, class-key	:	"punch" replacement of absentee cards for the whole class
FIN	:	terminate session

This subsystem allowed each assistant principal or teacher to print an up-to-date class list any time, or to replace lost cards in the data generation kit.

The course master file query language had more flexibility than those which we have just discussed. It allows one to display selective items of a single record defined set of records. It had the following commands and display functions:

```
MAT-152201   PROF. 3305 DUFOU LUC  LOC. C219 11/01 03/15 79/04/17 LISTE#  0
#GARS=  12 #FILLES=  11  #DET   5  #INS  28

  1 7450258     BEDARD  MARIE JOSEE        11/01           0
  2 7450270     BELANGER  ALAIN            11/01           0
  3 7704697     BLAIS  CHRISTIAN           11/01           0
  4 7450523  ***BONHOMME  DENIS            11/01  11/23    0
  5 7450555     BOUCHARD  DIANE            11/01           0
  6 7450601     BOURDAGES  LUC             11/01           0
  7 7450829     CARIGNAN  JOELLE           11/01           0
  8 7450991     CHOQUETTE  MARIO           11/01           0
  9 7451111  ***COTE  DENIS                11/01  11/23    0
 10 7412132     DEZIEL  MARIO              11/01           0
 11 7452129     GOUGEON  JEAN YVES         11/01           0
 12 7452233     GUILLET  LYNE              11/01           0
 13 7452252     HACHE  LINE                11/01           0
 14 7602059     LACOURSE  BRIGITTE         11/01           0
 15 7703856     LACOURT  SYLVIE            11/01           0
 16 7452679     LAMPRON  FRANCE            11/01           0
 17 7452736     LANGLOIS  JEAN LOUIS       11/01           0
 18 7452789     LAPOINTE  JACQUES          11/01           0
 19 7452894     LAVILLE  GUYLAINE          11/01           0
 20 7424561     LEDUC  MARIO               11/01           0
 21 7424426     MARTEL  RICHARD            11/01           0
 22 7602012     MAURICE  PIERRE            11/01           0
 23 7800498     POMONE  ARMAND             11/01           0
 24 7363763  ***PREVOST  MARC              11/01  03/15    0
 25 7700940  ***RABY  CHRISTIAN            11/01  03/15    0
 26 7454196     ROY  LOUISE                11/01           0
 27 7601964  ***TREPANIER  LISE            11/01  02/19    0
 28 7804789     DAIGNEAULT  NANCY          01/24           0
COPYCL, COPYHR, PUNCH, MRCARD, ZERO, PUNCHS, FIN.
?
```

Legend

1. *Course identifier*
2. *Teacher identifier*
3. *Room number*
4. *Date*
5. *Number of boys*
6. *Number of girls*
7. *Deleted students*
8. *Total registered*
9. *Date registered*
10. *Date dropped out*
11. *Student id. number*
12. *Student name*
13. *Menu of interactive commands.*

Figure 48 Man-computer dialogue while using the class master file query language to display a class list.

CHI-27046201.3/JOUR LOC. C215 11/01 02/13 79/04/17 LISTE# 0
#GARS= 15 #FILLES= 17 #DET 1 #INS 33 PROF. 3202.CHARTRAND.G.

Heading

1	7450069	***AUBE DANIEL	11/01	02/13	0
2	7550133	BAILLARGEON THERESE	11/01		1
3	7604131	BEAUCHEMIN PASCALE	11/01		0
4	7550803	BOULAIS NICOLE	11/01		1
5	7550937	BROSSEAU PIERRE	11/01		1
6	7300513	BROUILLARD BRUNO	11/01		4
7	7551434	COTE SUZANNE	11/01		8
8	7551582	DANSEREAU JULIEN	11/01		0
9	7552142	FAUCHON DANIEL	11/01		1
10	7700949	FONT NATHALIE	11/01		4
11	7552239	FORTIER HELENE	11/01		1
12	7552578	GEOFFRION LUCIE	11/01		1
13	7552811	GROLEAU JOHANNE	11/01		2
14	7601909	KALBAKJI RITA	11/01		13
15	7553742	L ECUYER MARCEL	11/01		8
16	7362954	L HOMME NORMAND	11/01		0
17	7553326	LAGACE JOANNE	11/01		6
18	7553549	LAREAU GAETAN	11/01		0
19	7703841	LEPAGE DENIS	11/01		0
20	7554043	MAISONNEUVE GISELE	11/01		7
21	7538905	MALONEY FRANCE	11/01		0
22	7554174	MASSICOTTE BENOIT	11/01		1
23	7554252	MEILLEUR LOUISE	11/01		4
24	7539125	MONAST JOSEE	11/01		0
25	7554753	PERREAULT NICOLE	11/01		3
26	7554846	PLANTE GUYLAINE	11/01		6
27	7800481	RATHE LUC	11/01		2
28	7555098	RENE DE COTRETMARTIN	11/01		9
29	7555307	ROYAL ANNIE	11/01		1
30	7555554	SOUCY GUY	11/01		2
31	7704759	SPIEGLE ROBERT	11/01		4
32	7454518	THERIAULT YVES	11/01		12
33	7555753	THIBAULT GUY	11/01		10

Number of lectures missed by the student from this course

Figure 49 Man-computer dialogue while using the class master file query language to obtain a class list with the cumulated truancy data for this particular course.

DISREC, course-key	:	retrieve course record and display all items on it (see Figure 50)
DIS, course-key, $item_x$, $item_y$, $item_z$	:	retrieve course record but display only the selected items on it ($item_x$, $item_y$, $item_z$). The record contains 24 numbered items from 1 - 24 (x, y, z vary from 1 - 24). This is a short display command.
FIN	:	terminate the session

This program contains a powerful looping feature. The course-key was a unique 8 digit number constructed in a very special and meaningful way. (51) For example, the course-key: 28122013

$\underbrace{2\ 8\ 1}_{a}\quad \underbrace{2}_{b}\quad \underbrace{2\ 0}_{c}\quad \underbrace{1\ 3}_{d}$ could be decomped into the following elements:

a : 3 digit subject code (281 = mathematics code)

b : high school level indicator (1, 2, 3, 4, or 5)

c : 2 digit course level indicator

d : 2 digit group number

Let us say the principal wanted a list of all the mathematics teachers in the school. He would use the short display command "DIS". But even with the command "DIS", he would have to remember the exact course keys of all his classes, a difficult task in a school with 1200 classes. So to permit easy queries of this type, a course-key masking option was introduced. By masking all unwanted portions of a course-key in the command, the system could display the desired record item for the subset of records whose course key element match the unmasked portion of the course-key.

For example:

DIS, 281*****, 3

would display all mathematics class teachers of the school (281 = mathematics code, 3 = item-no. for teacher's name).

DIS, ***5****, 3

would display all staff teaching in the school at level 5

(3 = item-no. for teacher's name)

DIS, 281*****, 2

would display all rooms where mathematics is being taught

(item no. 2 = room number)

The course master file query language proved to be a powerful tool for the vice-principal in charge.

```
 1  .......CODE OFFICIAL%.........................     58042103
 2  .......CODE DE LA MATIERE ET LOCAL.......          FRA-A216
 3  .......NOM DU PROFESSEUR%.....................     3213.PIERRE.M.
 4  .......SIM 1%.................................     58092103
 5  .......SIM 2%.................................     03
 6  .......SIM 3%.................................     03
 7  .......SEQ%...................................             183
 8  .......PRIORITE%..............................
 9  .......CAPAC. TOTALE%.........................             34
10  .......NO. MAX. DE GARS%......................             34
11  .......NO. MAX. DE FILLES%....................             34
12  .......NO. DE GARS%...........................             19
13  .......NO. DE FILLES%.........................             13
14  .......NUMBER OF STUDENTS WITHDRAWN%..........              1
15  .......NUMBER OF STUDENTS REGISTERED%.........             33
16  .......SCHEDULE%..............................
           J 1 (                    )
           J 2 (          4         )
           J 3 (   1                )
           J 4 (               6    )
           J 5 (      2             )
17  .......EXT - SCHEDULE%........................

           J 1 (                    )
           J 2 (                    )
           J 3 (                    )
           J 4 (                    )
           J 5 (                    )
18  .......DATE RECORD CREATED%...................       78/10/29.
19  .......DATE LAST MODIFIED%....................       78/10/29.
20  .......DATE LAST ACCESSED%....................       79/12/19.
21  .......DATE DELETED%..........................
22  .......SCHOOL TERM%...........................              3
24  .......JOUR/NUIT%.............................      JOUR

DISREC     DIS      MOD      CRE      DEL      UNDEL     FIN
```

<u>Legend</u>

1. *Record item number*
2. *Nature of the record item*
3. *Value of the record item*
4. *Menu of interactive commands available to the operator.*

Figure 50 Man-computer dialogue while using the course master file query language to display all items in the course master file record.

Part of his job involved the organization and supervision of the allocation of students among teaching, spatial and time table resources.

These three query languages allowed one to query the data-base in many ways. For large volume queries a pseudo-batch process in recommended. A "program" is written in the commands of a particular query language with the help of the system's text editor, and stored until execution time. Then they act as a driver to the main compiled data-base query program. The output from this process is spooled. The spooled output is then disposed to Concordia University computer center's high speed printer. This feature was used whenever new class lists are needed or at the end of the school semester when a report on all attendance records were required. This flexibility of information access was essential to satisfy all facets of the administration's needs.

CHAPTER 5

CONCLUSION

The Attendance Monitoring System described in this thesis was used to effectively combat truancy problems in three diverse educational environments. The system was first implemented in Ecole Mont St. Gabriel in St. Bruno which is a small school with a total student population of 500 students. This system was subsequently implemented in the College de Longueuil, which is a moderately large school with 1200 students. It was finally implemented in the Ecole Polyvalent Andre Laurendeau which is a large comprehensive high school in St. Hubert with a student population of 3300. Various studies showed that it successfully met the multiple objectives of the department of education, the school administration and teachers. (10, 24, 27, 33, 39)

The system eased many administrative problems and conflicts that arose between the school's administrators and its teaching staff. The incidence of problem turancy cases dropped by 80% over the first year of the system's implementation. As a consequence of these dramatic changes the teachers overwhelmingly voted (98%) to reinstate the system for a subsequent year of operation. The school's administration gained back control over the truancy problems that had undermined its credibility with its teachers, student population and their parents.

The students at first attempted to defeat the system which finally gained their respect and the nickname the "stooling machine".

Sociological research work undertaken with the help of the system demonstrated that the 10% of the student population that drop out of high school each year definately follow a characteristic truancy pattern before hand. This research allows one to develop and test intensive counselling strategies aimed at helping the problem student. By the end of second year of the system's operation the majority of the teachers wanted to use the system's database query options to get involved in student counselling themselves in the home room environment. In fact over twenty teachers wanted to enhance their use and understanding of the system by designing and writing programs for their own classroom needs.

The logical design of this system required a software complexity that clearly demonstrates the value of ABL. In such a demanding task, ABL also provided a methodology for the design and implementation of the system's physical characteristics. It is a most effective tool to achieve the functional breakdown of a system to its components and achieve a logical simulation of it.

Though many problems arose in this implementation four guidelines helped overcome these difficulties. The first of these was to work closely with the user. By working closely with the user we mean that it is necessary to observe or research the user's needs as well as continously communicating to him the evolving design of his system. By establishing such close contact with the future users of a system one can assure the realistic modelling of the overall system and an adequate design of the system's software.

A second guideline involved the assessment of the system's performance after its implementation. Such a study allowed us to detect shortcomings and insufficiencies which should be corrected before the system's users would become alienated or disenchanted with the system.

A third guideline, which we followed, involved designing a system which was flexible enough to be easily expanded to incorporate new features requested by its users in the future. In particular the data base was designed with ample room for the storage of additional data in its records.

The fourth guideline involved considered the skill and capabilities of its users. The system was designed to be operated by the employees of the school. Therefore its job control language and logic had to be clear and concise.

These four guidelines allow us to avoid serious technical problems in the system's design and implementation.

A major problem did arise however from a totally (25) unpredictable human factor, which ironically traumatized the middle executive level of the school's administration. The system provided a great deal of useful data, some of which clearly demonstrated the relative effectiveness of various vice-principals. This statistical information which was available to the school's administration played on the insecurity and paranoia of those individuals who seemed to be less effective in the job of student and truancy control. Two of the affected vice principals opted out of using the system for awhile. It took a great deal of diplomacy and objectivity from senior administors to reassure the staff that the intended use of the system did not involve their surveillance but rather that of the student population. The vice-principals finally accepted the good faith and reassurances of the school's administration on this point.

Minor technical problems arose in assuring the accurate scheduling of the data collecting and processing phases of the system. Further refinements were needed in the system's reporting facilities and replacement procedure for lost or stolen cards. The last two difficulties were detected by our own information utilization study. This study demonstrated that while the system's detailed report (report B) was praised by all concerned, it did not facilitate the vice-principle's ability to confront and help truant students. Essentially, this report was too long: it offered detailed information on too many students. This problem was resolved by introducing output selection parameters which drastically reduced the report's volume by assuring an automatic selection of most serious truancy problems for reporting and subsequent counselling.

In final analysis we can say that the system's information had a positive catalitic effect on the school's administration.

About 58 Quebec school administrators came to visit our pilot project and said that they would like to have such a system at their disposal. During three consecutive years, high school administrators from France were also visiting us.

However comprehensive monitoring and control systems such as the one described here, can be used to violate the personal freedom and privacy of all concerned. The responsibility for the administration of the school's data base is often fought over. Information in these systems can be used as weapons of confrontation in times of hostility between the government, the school boards and its civil servant unions. Partisan interests can misuse their authority through the ugly and powerful manipulation of such flexible control systems. This system was discontinued as a result of a policy decision that was independent of its technical performance.

Further work would involve extending the system's utility so that it could handle test scoring, mark accounting, mark reporting as well as course scheduling and student registration in a distributed data entry environment. Further work on the system would also make it applicable to attendance monitoring problems elsewhere. (45) Certain generic characteristics of the system make its use applicable to problems found in other record management situations for example those encountered in the medical profession. (20)

BIBLIOGRAPHY

(1) Barbadora, B. M. "A Brief Description of the School Information System of the Cincinnati Public Schools." Cincinnati Public Schools, Ohio, Apr. 1973, 7 p., Papers presented at the Association for Educational Data Systems, Annual Convention, New Orleans, Louisiana.

(2) Carlson, E. D. "Evaluating the Impact of Information Systems." Auerbach Publishers Inc., Systems Development Management 1976.

(3) Chai, D. T. "An Information Retrieval System Using Keyword Dialog." Information Storage Retrieval, Volume 9, pp. 373-387 Pergamon Press 1973, Printed in Great Britain.

(4) Date, C. J. "An Introduction to Data-base Systems." Addison-Wesley Publishing Company, Don Mills, Ontario, 1976.

(5) Descheneaux, R. "Evaluation du Système de Contrôle des Absences par Informatique." Commission Scolaire Régionale de Chambly, Service de l'Informatique, Rapport non publié, Juin 1973.

(6) Dickson, G. W. "Control Systems for Information Systems Development Projects." AEDS Journal, Volume 4, December 1970, Number 2.

(7) Egbert, R. L. "Buena Vista High School - Systems Analysis". System Development Corp., Santa Monica, Calif. Feb. 66, 49 p.

(8) Fiddleman, R. H. and Gorman, M. M. "A Management Information System for local School Districts." AEDS Journal, Volume 5, Number 3, March 1972.

(9) Gehrke, Ch. "Protecting Privacy When Using the Computer in Educational Research." AEDS Journal, Volume 6, Summer 1973, Number 4.

(10) Gendron, G. "Le Refus Scolaire et Comment y Remédier." Thèse de Maîtrise, Faculté des Sciences de L'Education, Section d'Administration Scolaire, Université de Montréal, Juin 1981.

(11) Gitomer, J. "Translating Functional Specifications Into Detail Design." Auerback Publishers Inc., Systems Development Management 1977.

(12) Gitomer, J.. "Defining Detailed Systems Modules." Auerbach Publishers Inc., Systems Development Management 1978.

(13) Goldberg, A. L. "Information Systems Analysis and Design (Observation from a Field Experience)." AEDS Journal, Volume 11, Spring 1978, Number 3.

(14) Gruman, A. J. "On line System for Student Master File Records and Attendance." Papers presented at the Annual Conference of California Educationnal Data Processing Association, San Diago, California, Dec. 1973.

(15) Hartig, W. R. "RAMA - Recursive Attendance Monitoring and Analysis." Trenton Public Schools, N.J., Aug. 72, 18 p.

(16) Heaps, H. S. "Computational and Theoretical Aspects of Information Retrieval." Unpublished Preliminary Version, December 1974, Computer Science Department, Concordia University, Montreal.

(17) Heintz, T. J. "How Small College Administrators Can Evaluate the Potential Need for Computerized Information Processing." AEDS Journal, Volume 11, Summer 1978, Number 4.

(18) Hunter, B. "Designing Information Displays." Auerbach Publishers Inc., Systems Development Management 1976.

(19) Jaworski, W. M. and Hinterberger, H. "Controlled Program Design by Use of the ABL Programming Concept." Angewandte Informatik, 7/81.

(20) Jaworski, W. M. and Williams, A. J. "A Representation of Therapeutical Strategies in Dermatology." Conference of Canadian Dermatological Association, July 2 - July 6, 1982, Edmonton, Canada.

(21) Jaworski, W. M. and Francott, T. "Table Based Constructs for Programming Languages." Paper published by the Department of Computer Science, Concordia University, Montreal, December 1975.

(22) Kirby, E. "Analyzing the Existing System." Auerbach Publishers Inc., Systems Development Management, 1978.

(23) Kirby, E. "Conducting User Training." Anerbach Publishers Inc., Systems Development Management, 1978.

(24) Maire, H. "Mémoire de l'Ecole André-Laurendeau au sujet d'un Système Intégré de Gestion Pédagogique et Administrative Assistée par Ordinateur en Mode Interactif." Rapport, Polyvalente André-Laurendeau, Commission Scolaire Régionale de Chambly, 1979, P.Q.

(25) Martin, J. "Principles of Data-Base Management." Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1976.

(26) Mensh, S. "Identifying User Requirements." Auerbach Published Inc., Systems Development Management, 1977.

(27) Meynard, F. and Chevalier, G. "Système Intégré de Gestion Assisté par Ordinateur -- Rapport d'Appréciation." Ministère de l'Education, Service Général des Moyens d'Enseignement, Mars 1978.

(28) Ministère. "SIGNE (Système Informatique de Gestion des Elèves) Les "ABSENCES"." Analyse Fonctionnelle, Gouvernement du Québec, Ministère de l'Education, Service de l'Informatique, 1976.

(29) Mitchell, J. E. "Implementation of MSEIP'S (Midwestern States Educational Information Project) Integrated Information System." AEDS Journal, Volume 5, Number 3, March 1972.

(30) Morgan, A. H. "An Engineering Approach to Problem Analysis." A master thesis in the department of Computer Science at Concordia University, Montreal, Quebec, Canada, May 1981.

(31) Murdick, R. G. and Ross, J. "Information Systems for Modern Management." Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1971.

(32) Muscat, E. J. and Lorton, P. "Cinch: An Attendance System for Small Time-Sharing Systems." Woodrow Wilson High School, San Francisco, Ca., U.S.A. 1976 AEDS International Convention Proceedings - Today's Revolution, Computers in Education.

(33) Legendre, R. "Une Education à Eduquer." Editions France-Québec, Montréal, 1979.

(34) Palkiewicz, J. and Horvath, A. "Communication Educative." Info Recherche, no. 31, Commission Scolaire Régionale de Chambly, 1975.

(35) Peck, P. H. "NASA Begins Countdown on School Problems." School Management, 15, 9, 32., Sept. 71.

(36) Poly, Groupe (200 members). "Rapport du Groupe Poly, L'Organisation et le Fonctionnement de l'Ecole Secondaire Polyvalente." Gouvernement du Québec, Ministère de l'Education, 1974.

(37) Pooch, U. W. and Niedler, A. "A Survey of Indexing Techniques for Sparse Matrices." Computing Survey, Vol. 5, No 2, June 1973.

(38) Powers, M. J. "A Management Information System for Academic Administrators." AEDS Journal, Volume 11, Summer 1978, Number 4.

(39) Robert, G. "Evaluation du Terminal à l'Ecole Secondaire." Fédération des Principaux de Québec, 1979.

(40) Ross, D. T. "Structured Analysis (SA): A Language for Communicating Ideas." IEEE Transactions on Software Engineering, January 1977.

(41) Ross, G. "Rethinking Data Base Systems for Education." AEDS Journal, Volume 9, Winter 1976, Number 2.

(42) Ross, T. D. and Schoman, E. E. "Structured Analysis for Requirements Definition." IEEE Transactions on Software Engineering, January 1977.

(43) Rubin, M. "Developing Interactive Data Formats and Procedures." Auerbach Publisher Inc., Systems Development Management 1976.

(44) Sadek, K. E. "The Human Element - Critical Factor in Systems Design." Canadian Data systems, September, 1980.

(45) Sire, P. W. "Basis: Business Administration Student Inventory System." AEDS Journal, Volume 3, December 1969, Number 2.

(46) Spuck, D. W. and Bozeman, W. C. "A Design for the Evaluation of Management Information Systems." AEDS Journal, Volume 14, Fall 1980, Number 1.

(47) Taylor, H. L. "Planning Data Communications Functions." Auerbach Publishers Inc., Systems Development Management, 1976.

(48) Thierauf, R. J. "System Analysis and Design of Real-Time Management Information Systems." Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1975.

(49) Thompson, D. A. "Interface Design for an Interactive Information Retrieval System: A Literature Survey and a Research System Description." Journal of the American for Information Science, November-December 1971.

(50) Vallee, J. F. and Askevold, L. "Information Organization for Interactive Use: Design Implications in Data-Base Systems." Journal of the American Society for Information Science, July-August 1973.

(51) Weaver, B. N., Ruffner, A. B., Ellis, C., Anderson, H., and DeLorenzo, S. X. "An Automated Storage and Retrieval System for Clinical Pharmaceutical Research Documents." Journal of the American Society for Information Science, November-December 1970.

(52) Wen, T. and Iyengar, S. "Application of Linked List Technique for the Enhancement of Traditional Random Access Files." Computers & Education, and International Journal, Volume 4, Number 3, 1980.

(53) Whitney, F. L. "The Minicomputer in Student Accounting in the High School." Glendale Union High School District, Glendale, Az, U.S.A. 1976 AEDS International Convention Proceedings - Today's Revolution, Computers in Education.

(54) Winings, J. W. "A Data Structure for Information Retrieval.", Journal of the American Society for Information Science, March-April, 1970.

(55) Wolmut, P. "A Study of the Optimal Answer Sheet Identification Grid Design." AEDS Journal, Volume 11, Summer 1978, Number 4.

(56) Wong, Y. "Computer Simulation of Student Mobility Patterns." AEDS Journal, Volume 2, June 1969, Number 4.