The University of Alberta

CHILDS:
A Library Environment for Children

by

Raymond Kim Ho

A thesis
submitted to the Faculty of Graduate Studies and Research
in partial fulfillment of the requirements for the degree
of Master of Science

Department of Computing Science

Edmonton, Alberta
Fall, 1987

ISBN 0-315-40979-7

The University of Alberta

Release Form

Name of Author: Raymond Kim Ho

Title of Thesis: CHILDS: A Library Environment for Children.

Degree: Master of Science

Year This Degree Granted: 1987

Permission is hereby granted to The University of Alberta Library to reproduce single copies of this thesis and to lend or sell such copies for private, scholarly or scientific research purposes only.

The author reserves other rights, and neither the thesis nor extensive extracts from it may be printed or otherwise reproduced without the author's written permission.

(Signed) ______________________

Permanent Address:

Flat G, 26/F, Block 3
Neptune Terrace
Lok Man Road
Chai Wan
Hong Kong

Date: October 15, 1987

The University of Alberta

Faculty of Graduate Studies and Research

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research, for acceptance, a thesis entitled **CHILDS: A Library Environment for Children** submitted by **Raymond Kim Ho** in partial fulfillment of the requirements for the degree of **Master of Science.**

Supervisor

Date: October 15, 1987

Abstract.

The CHildren's Interactive Library Display System (CHILDS) is a computerized library management system designed for elementary school libraries. CHILDS automates school libraries by providing self-serve loan/return, search and inventory control services. This thesis describes the second version of CHILDS which includes an efficient robust file system and an innovative computer interface for children. The elementary school library environment is described first, including the manual system which was used before the installation of the first version of CHILDS in spring of 1985. Then the first version is described and its failures are discussed. The system requirements and solution strategy of the second version are then specified. The software design of the second version is then described and some interesting results concerning data structure design and searching techniques are presented. From the problems encountered by children using both version of CHILDS, a set of basic principles which is especially important for constructing computer interfaces for children is observed.

Acknowledgements.

I would like to thank my supervisor, Dr. Duane Szafron, for the technical and moral support he has provided. I would also like to thank the school librarian at LaPerle Elementary School, Marilyn Dale, for providing expert advice on the system requirements. Finally, I would like to thank the students, teachers and parent volunteers for using an experimental system whose interface changed so often over the course of the year.

Table of Contents.

List of Figures.

Chapter 1.

Introduction.

In the past few years, much research has been conducted on the subject of human-computer interfaces for novices. However, very little work has been done on the problem of computer interfaces for children. The small amount of work which has been done focuses mainly on interfaces for teaching children to solve problems by programming [Gold81,Pape80]. One reason why researchers have shown little interest in computer interfaces for children is that in the past, most computer interfaces have been text oriented. Therefore, it was almost impossible to design an interface specifically for children due to their limited reading ability. However, advances in technology have provided a powerful tool, computer graphics, which potentially can be used to solve the problem.

In order to develop a set of principles for children-computer interfaces, one approach would be to use a computer system to gather information and test hypotheses. Such a system should be one which could be used by children regularly on a day-to-day basis. The results obtained on such a system would then be directly related to a wide range of children in a natural setting as opposed to a selected group of children in an artificial laboratory environment.

CHILDS (CHildren's Interactive Library Display System) was developed to study this subject. CHILDS provides self-serve loan/return, search, and inventory services for an elementary school library. The main objective of CHILDS is to enable children in an elementary school to perform loan/return and search operations without any adult supervision. A prototype implementation of the first version was installed at the LaPerle Elementary School library in Edmonton in spring of 1985. Although some preliminary results were derived from the system, its use was discontinued after six months due to a corrupted file system.

A second version of CHILDS was needed to continue the research. Because of the failure of the first version, a lot of time and attention was spent on the planning of the second version. The second version was developed using the following steps:

1. Identifying the problems with the first version.
2. Specifying new system requirements.
3. Developing a solution strategy.
4. Designing and implementing a new file system.
5. Designing and implementing a new user interface.
6. Testing the final system.

Chapter 2 describes the library environment at the LaPerle Elementary School library and the manual system which was used prior to the installation of the first version of CHILDS.

Chapter 3 describes the first version of CHILDS which was first introduced in spring of 1985. Both the file system and the user interface are discussed. Since the use of the system was discontinued after six months, the cause for the failure of the system is also outlined.

Chapter 4 specifies the system requirements for the second version of CHILDS. These requirements are based on functional requirements obtained through consultation with the school librarian, interface requirements necessary to enable children to use the system, and hardware constraints imposed by the school's budget and its existing equipment.

Chapter 5 describes the solution strategy which was used in the second version of CHILDS. The solution strategy was designed to meet the system requirements. This solution strategy was based on three basic principles: data minimization, simple data entry and information integration.

Chapter 6 gives a full description of the file system design of the second version of CHILDS. Considerable effort was spent on the design of the file system to ensure that it is simple, reliable and efficient.

Chapter 7 discusses the capacity and limitations of the file system of the second verion of CHILDS.

Chapter 8 describes the user interface design of the second version of CHILDS. The interface which is used by the adult staff is briefly described. The interface which is used by the children is described in more detail. Based on the observations of the children using both version of CHILDS, a set of principles which could be applied to the construction of computer interfaces for children is outlined.

Chapter 9 discusses the subject of children-computer interfaces and comments on possible future improvements to CHILDS.

CHILDS is an ongoing research project involving many individuals. The major contributions of my research efforts have been to create a simple, efficient and reliable file system and to make considerable improvements to the user interface.

Chapter 2.

Problem Domain.

The LaPerle Elementary School library has a collection of approximately 5000 items consisting of: books, audio cassettes, records, pictures, video cassettes, maps, kits, and films. The library serves approximately 360 students and 13 teachers. The students are in kindergarten through grade six. The library staff consists of one part time librarian and several parent volunteers. Due to a shortage of staff, the library required a system to provide:

1. self-serve item loan/return;
2. easy to use item search;
3. quick inventory control.

2.1 Loan/Return.

Since the librarian is at the school only three days per week and the parent volunteers are only present on a casual basis, the loan/return system must be a self-serve one.

In the spring of 1984, the school library introduced a manual self-serve loan/return system based on the one being used at the McKinley Elementary School in Aloha, Oregon [Szaf82]. For each class in the school, a folder was kept in the library. Each folder was labelled with the room number of one of the classes. Each folder contained one pocket for each student in the class. The pockets were labelled with student names and arranged alphabetically. Each pocket contained two bristleboard student cards labelled with the student's name and room number. In addition, each item in the library had a pocket which contained an item card bearing its title and all the other standard information together with spaces for borrowers' names, room numbers and due dates. Finally a stamp with the current due date, together with pink slips with the current due date stamped on them were available at a desk in the library.

A student was required to perform the following procedures in order to borrow an item:

1. Locate an item in the library.
2. Take the item card from the item pocket and write his/her name and room number on it.
3. Go to the desk, stamp the due date on the item card, take a pink due date slip and place it in the item pocket.
4. Go to the class folder, locate his/her pocket, and place the item card in the pocket.
5. Take a student card from the folder pocket and place it in the item pocket.

A student was required to perform the reverse actions when returning an item.

Although adults found these procedures easy to use, students at the school had difficulty with the system. Some of the problems encountered were:

1. The procedures were too complex for children in kindergarten and grade one.
2. It was impossible to find out who had borrowed a particular item without searching all the folders.
3. It was time consuming to sort through all of the pockets of a folder to generate overdue notices.

2.2 Item Search.

The library required students to use a standard card catalogue system to locate items. It was very difficult and sometimes impossible for elementary school children to use this procedure. In fact, three separate skills which are necessary to use a card catalogue were identified [Szaf87] :

1. **Conceptualization**: the ability to conceptualize a subject name, a keyword from a title, or the name of an author.
2. **Recognition**: the ability to visualize or recognize the character string representing the concept.
3. **Categorization**: the ability to locate the character string in an alphabetical listing.

Although the age at which these skills are developed depends heavily on individual children, each succeeding step requires a higher degree of abstraction and therefore is more difficult. In general, children in kindergarten and grade one should possess the conceptualization level of abstraction while children in grade two and three should have acquired the recognition level of abstraction. Children in grade four and up should have attained the categorization level of abstraction. Therefore, if a card catalogue system is used for item searches, almost all children from kindergarten to grade three are precluded from using such a system.

2.3 Inventory Control.

A good inventory control system is essential for any library. However, school libraries have special requirements that are not found in other libraries. As well as generating overdue notices and identifying missing items, it is often desirable to identify popular items as well as those children who use the library and those children who do not. Item popularity can be used as a mean to determine reading patterns which can then be used in purchasing new items and retiring old ones. Children's use records can be used by teachers to encourage students to use the library more often. Although the manual system could be used for some of these purposes, it was awkward and very time consuming.

Chapter 3.

CHILDS (First Version).

This chapter briefly describes the first version of CHILDS which was introduced to LaPerle Elementary School in 1985 and the problems associated with it. Most of the information given in this chapter can be found in the project report of the first version by H. C. Chan [Chan85].

The objectives of the first version of CHILDS were:

1. To allow children to borrow and return items without the assistance of library staff.
2. To allow children to search for items.
3. To provide an easy way for parent volunteers to update library information.
4. To allow easy access to all the library information.

The hardware consisted of:

1. an IBM^{TM} PC with 256KByte of main memory;
2. a 10 MByte hard disk;
3. a 360 KByte floppy disk drive;
4. a Caere^{TM} bar code scanner;

5. an Amdek^{TM} Video-300A monochrome graphics monitor;
6. an Epson^{TM} printer.

Since the loan/return sub-system was to be used by kindergarten children and they are unable to read, simple icons were used to communicate with the children and guide them throughout the system. To eliminate the need for the students to type, a bar code scanner was used. The system was divided into four main parts:

1. Loan/Return System

   Each item and user was assigned a unique ID (integer). Bar code labels which corresponded to the IDs were attached to the items and user cards. The loan procedure required the borrower to scan the bar code scanner across the bar code labels on the item and on the user card. To return an item, the user was required to scan only the bar code label on the item.

2. Item Search

   The student could search for an item by typing the item ID, keywords, authors, subject, or dewey number. A list of books meeting the specified criteria was then displayed on the screen.

3. Data Entry

   The input of item and user information was relatively easy and simple. Templates were displayed on the screen and the parent volunteers filled in the blanks with the appropriate information.

4. Inventory Control

The librarian was able to access all item and user information through very simple operations.

3.1 The File System.

The file system consisted of two master files, a set of directory files and a set of inverted files. The master item file and master user file contained all the information about every item and every user. Each master item record contained an item ID, a list of title keywords, a list of authors, a list of subject keywords, a publisher, a year of publication, an ISBN number, a price, a dewey decimal number, a borrower ID, a loan date, a loan count, an item code and the status of the item. Each master user record contained a user ID, a name, a class name and a user type.

The master files were sorted in ascending order by ID numbers. A search by ID required a binary search on the master item file. For all the other search functions, an inverted file structure was used. Keyword, author and subject files shared the same file structure. Figure 3.1 shows the file structure for keywords as an example. In this example, there are five books containing the keyword DOG in their title.

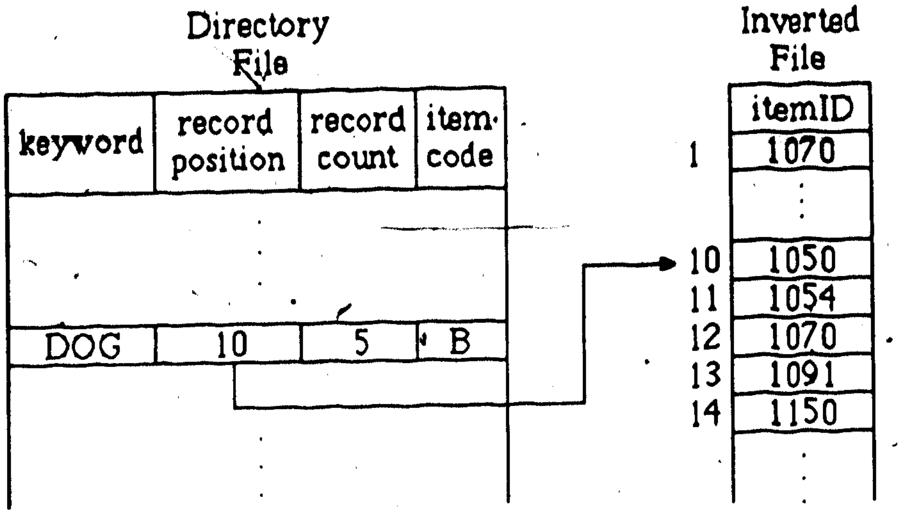


Figure 3.1 File structure of keyword files.

The directory file was sorted in ascending order of keywords. In each directory record, there was a pointer to the starting address of the item IDs in the inverted file. A record count kept track of the number of items that contained the particular keyword. For the section of the inverted file that was between two directory pointers, the item IDs were sorted in ascending order. A keyword search required a binary search on the directory file and all the items that contained the keyword could be found in the inverted file.

No directory file was used for dewey number since every item had a unique dewey number. Each record of the dewey inverted file contained the dewey number and the corresponding item ID. A binary search was required for a dewey number search.

An inverted file structure was also used to store the list of items that each user had borrowed. Figure 3.2 shows the structure of the borrower file. In this example, the student with ID equal to 9 has one item on loan and the ID of this item is 1234.

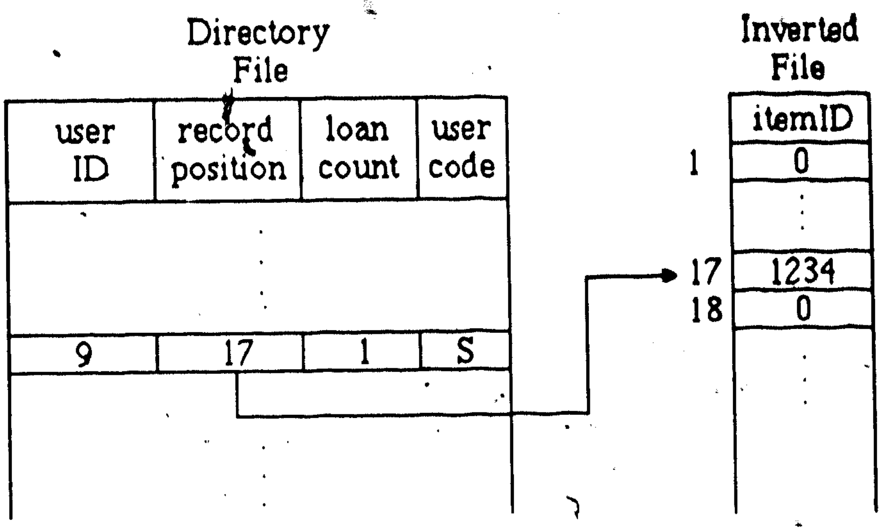


Figure 3.2 File structure of borrower file.

The number of items that a user borrowed was stored in the directory record and the IDs of the items that a user borrowed were stored in the inverted file. The size of the inverted file was changed when new users were added to the system (i.e., the loan limit of a user is fixed when the user is created).

Inverted files for classes and the last name of users were also included to allow the librarian to access information about all users in a particular class or about specific users by name. The class file has the same structure as the keyword file. The structure of the user name file was almost identical to the dewey number file except each record in the user name file contained the user name and the user ID.

3.3 File System Problems.

The major problem with the file system in the first version of CHILDS was its lack of flexibility. For example, as shown in Figure 3.1, the number of records corresponding to a specific keyword was fixed. Therefore, every time the librarian made a change to the title of an item, the whole inverted file for keywords had to be rebuilt. A similar problem existed with the borrower file due to the fact that the loan limits for students and teachers were set to be 2 items and 75 items respectively.

This problem was solved by storing changes and additions during the day and then performing the updates overnight (i.e., a batch update). At first, these updates were quite rapid, but by the time 5000 items had been entered, the daily update took over six hours.

Two separate aspects of this procedure contributed to the corruption. The first is that the update procedure was quite complex, involving the re-creation of the entire file system each night. Re-writing this much data increased the chances of media i/o errors.

Secondly, the batch update can leave the system in an inconsistent state. For example, consider the following case:

1. User ID 100 represents user A and user ID 200 represents user B.
2. In the morning, the librarian edits user ID 100 to represent user B and user ID 200 to represent user A.
3. During the day, user A comes in and borrows an item. Since the

update has not taken place yet, the item is loaned to user ID 100.

4. At the end of the day, the updates are performed.

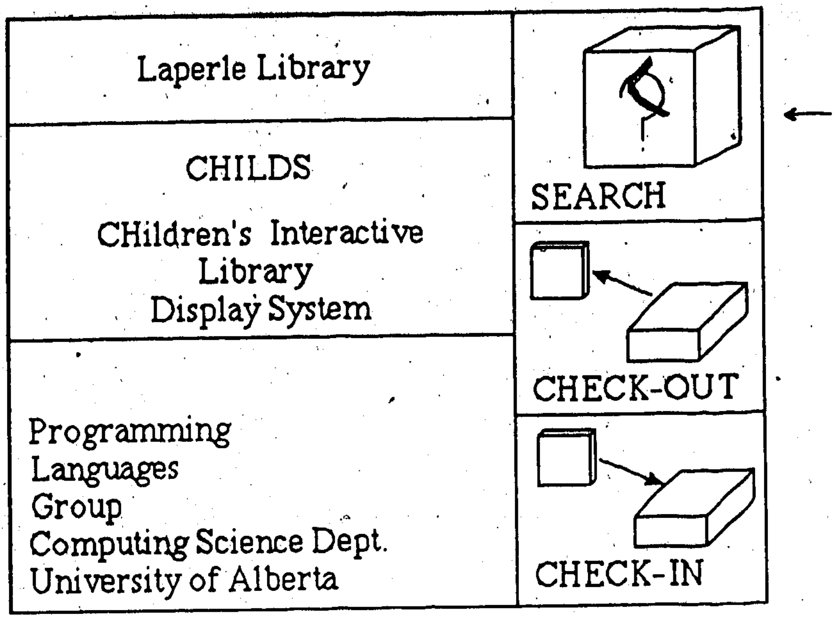
Now, the system will determine that user B is the borrower of the item rather than user A. The situation would be even worse if the information for two or more items was interchanged.

3.3 The User Interface.

The first version of CHILDS provided a wide range of library services which varied widely in complexity. This section describes only the loan/return and search sub-system.

The first version eliminated the need to read and type whenever possible. In situations where reading skills were required, the requirements were reduced to the absolute minimum. In situations where typing skills were essential, the input which the children had to provide was greatly simplified. Due to these simplifications, children in kindergarten were able to borrow and return items without any supervision and children in grade three and up were able to perform their own item search.

The loan, return and search procedures were combined into a single sub-system called the "loan sub-system". Figure 3.3 shows the main menu for the loan sub-system.



Laperle Library

CHILDS

CHildren's Interactive
Library
Display System

Programming
Languages
Group
Computing Science Dept.
University of Alberta

Figure 3.3 Loan sub-system main menu.

The arrow on the right hand side of the menu allowed the user to select one of the three functions. To select a function, the user was required to move the arrow to the appropriate function and press the back space key (which has a similar arrow on it). The up and down arrow keys were used to move the visual display arrow up or down.

3.3.1 Loan (Check Out).

To borrow an item, two pieces of information are needed: an itemID and a user ID. After selecting the CHECK OUT function from the main menu, icons of a bar code scanner and a book (Figure 3.4) are displayed.

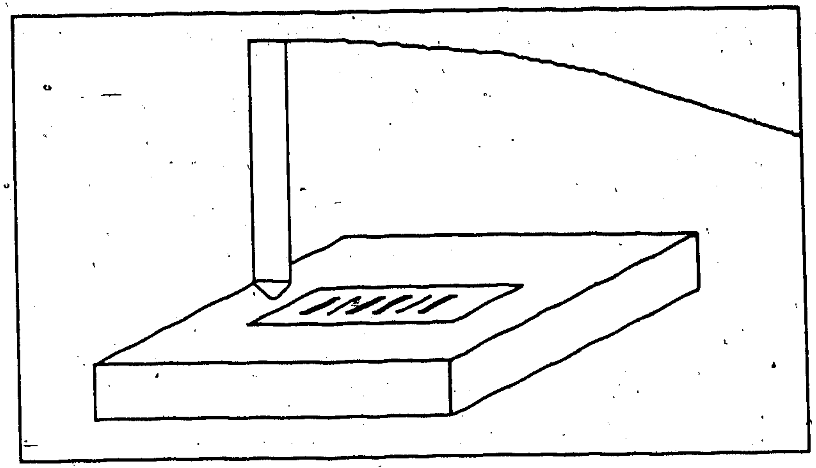


Figure 3.4 Input itemID icon.

After sliding the bar code scanner across the bar code label on the item, three possible situations could happen:

1. Item is OK;
2. Item is not found;
3. Item is already checked out.

3.3.1.1 Item Is OK.

In the case where the item is ok, icons of a bar code scanner and a library card (Figure 3.5) are displayed.

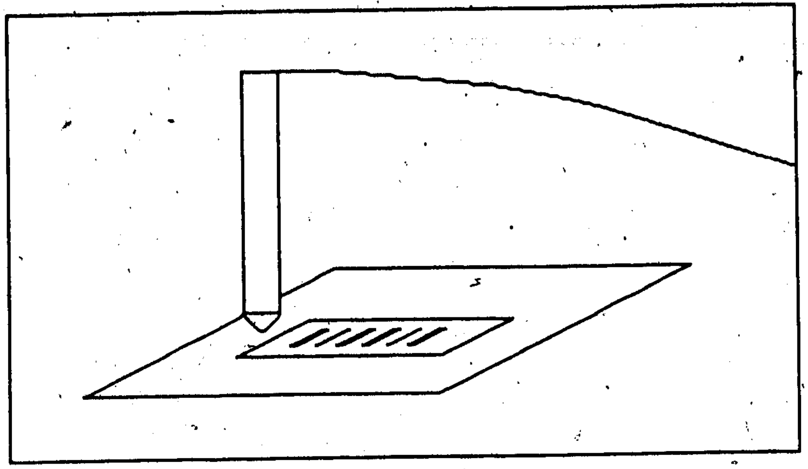


Figure 3.5 Input userID icon.

After sliding the bar code scanner across the bar code label on the user's library card, one of three possible situations could occur:

1. Successful loan;
2. User has too many books;
3. User is not found.

3.3.1.1.1 Successful Loan.

A successful loan means that the item has been successfully checked out by the user. An icon of a happy face (Figure 3.6) is displayed briefly, disappears and the system returns to Figure 3.4 after 3 seconds.

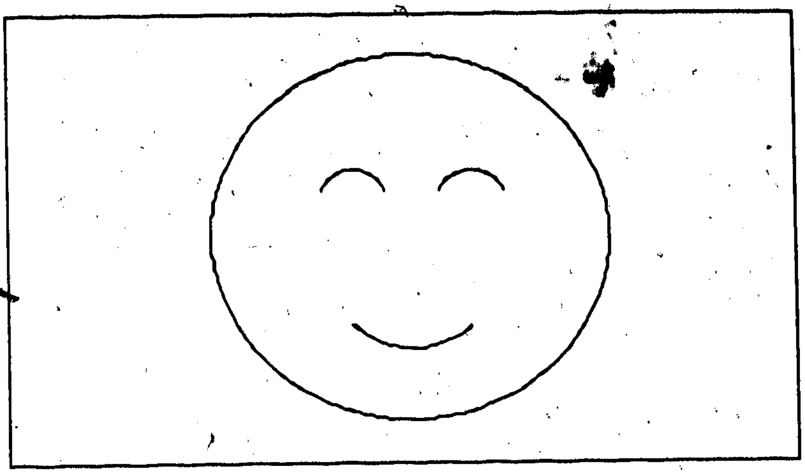


Figure 3.6 Success icon.

3.3.1.1.2 User Has Too Many Books.

If the user has reached his/her loan limit, an icon of six books with an X across them (Figure 3.7) is displayed.

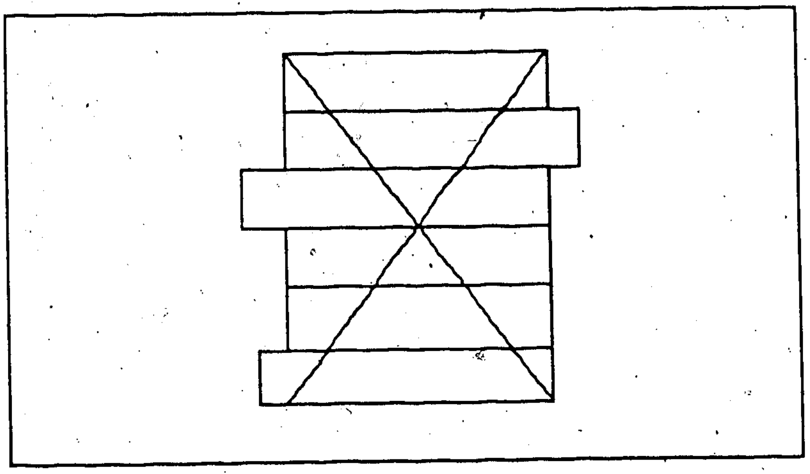


Figure 3.7 User has too many-books icon.

The user acknowledges this error by pressing any key.

3.3.1.1.3 User Is Not Found.

If there is no user registered in the library with the number scanned, an icon of a stick person and a question mark (Figure 3.8) is displayed.

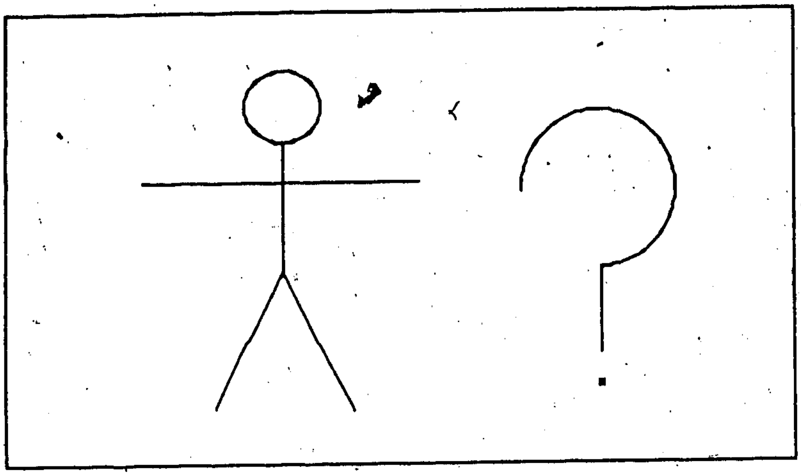


Figure 3.8 User is not found icon.

In this case, the user should ask the librarian for help. This error can be cleared by pressing any key.

3.3.1.2 Item Is Not Found.

If there is no item in the library with the number scanned, an icon of a book and a question mark (Figure 3.9) is displayed.

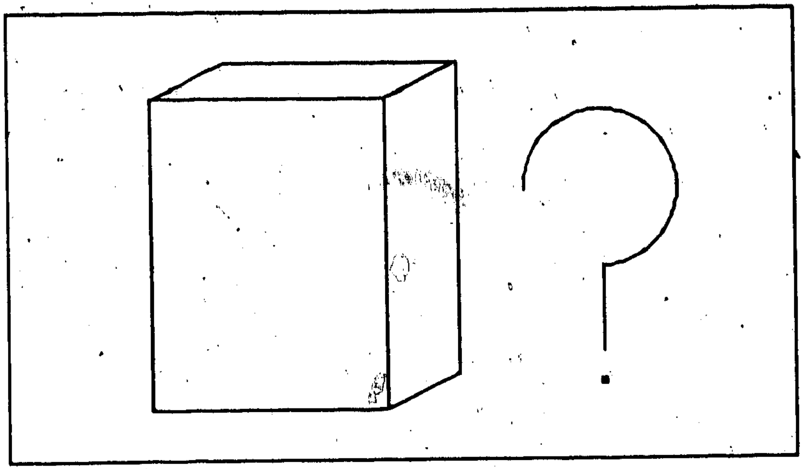


Figure 3.9 Item is not found icon.

In this case, the user should ask the librarian for help. This error can be cleared by pressing any key.

3.3.1.3 Item Is Already Checked Out.

If another user has already checked out the item, an icon of a sad face (Figure 3.10) is displayed.

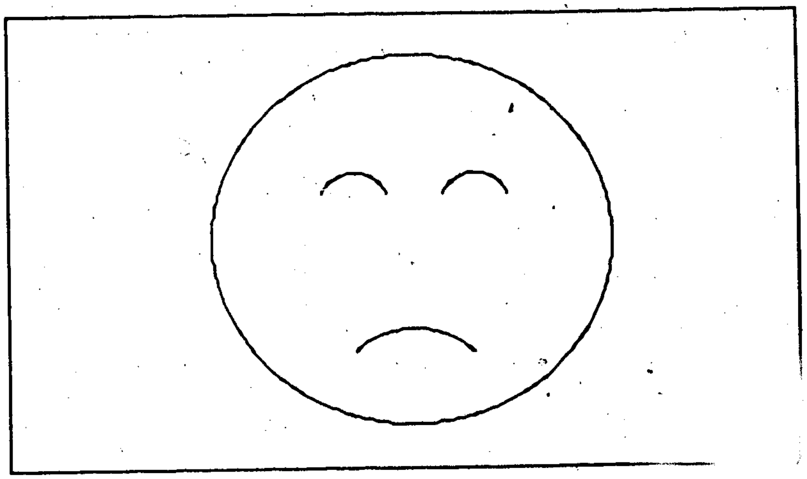


Figure 3.10 Failure icon.

In this case, the user should ask the librarian for help. This error can be cleared by pressing any key.

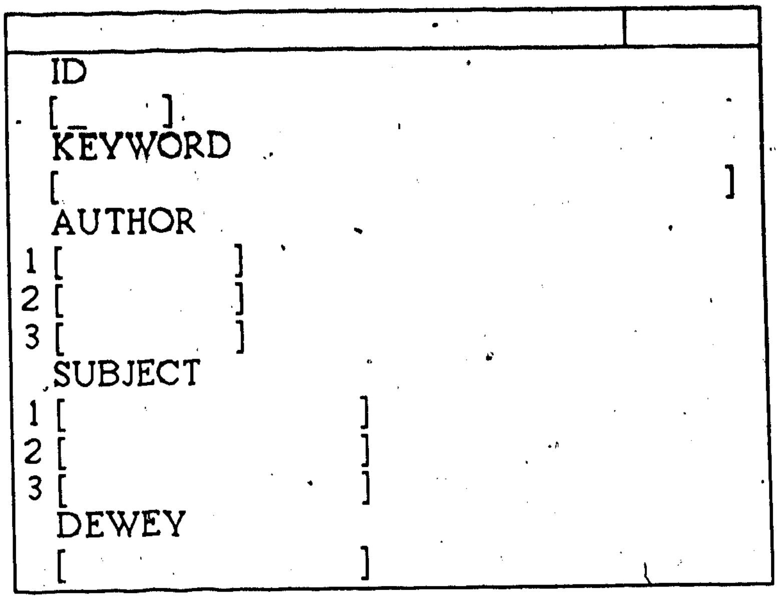
3.3.2 Return (Check In).

To return an item, only one piece of information is needed: an item ID. After selecting the CHECK IN function from the main menu, the icons of a bar code scanner and a book (Figure 3.4) are displayed. After sliding the bar code scanner across the bar code label on the item, three possible situations could happen:

1. Return is successful and the icon of a happy face (Figure 3.6) is displayed.
2. The item is not found and the icon of a book and a question mark (Figure 3.9) is displayed and the user should ask the librarian for help.
3. The tem is not checked out and the icon of a sad face (Figure 3.10) is displayed and the user should ask the librarian for help.

3.3.3 Search.

A user can search for items by item ID, keywords from title, authors, subjects, or dewey number. After selecting the SEARCH function from the main menu (Figure 3.3), the search menu (Figure 3.11) is displayed.



```
  ID
  [_    ]
  KEYWORD
  [                                          ]
  AUTHOR
1 [          ]
2 [          ]
3 [          ]
  SUBJECT
1 [                 ]
2 [                 ]
3 [                 ]
  DEWEY
  [                 ]
```

Figure 3.11 Search menu.

A small blinking character, the cursor, is also displayed. The user can move the cursor anywhere in the menu by using the cursor control keys (i.e., the up, down, left and right arrow keys). To request a search for an item, the user must type in the appropriate information in the corresponding field on the search menu and press the return key. For example, to search for a list of items about dogs, the user types DOGS in the first line under SUBJECT and presses the return key.

After the user has typed in the information and pressed the return key, the list of item types (Figure 3.12) is displayed.

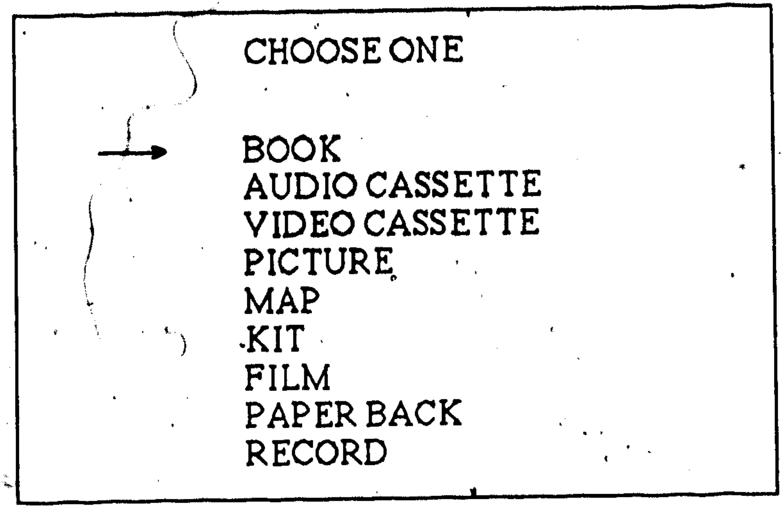


Figure 3.12 List of item types.

To select a particular item type, a user moves the cursor to the desired type and presses the return key.

If the system finds items that match the search criteria, the information for the items is displayed. If the system cannot locate an item with the given information, an icon of a book and a question mark (Figure 3.9) is displayed.

3.4 User Interface Problems.

Besides the problems associated with the file system, there were problems with the user interface of the first version. Some of the problems encountered by the children were:

1. Children were sometimes confused as what they were supposed to do in a particular situation.
2. Children sometimes thought they had successfully checked out an

item when actually they had not.

3. Some children were confused about their loan limit.

4. Children in kindergarten and grade one were unable to perform searches because the search procedure was text oriented and they had difficulty reading the instructions.

The first problem was caused by using the same icons for different meanings in different situations. For example, the input item ID icon (Figure 3.4) was used for inputting an item ID both during the loan and return procedures. This caused confusion when children intermixed loan and return operations during a library period. This problem was solved in the second version of CHILDS by displaying a small icon in the top right hand corner to distinguish between the two different operations.

The second problem was caused by the fact that some children ignored error icons. For example, consider a child who is attempting to borrow an item. Instead of scanning the bar code label on the item card (as shown on the display), suppose that the child scans the bar code label on the user card. The icon indicating that the item is not found (Figure 3.9) is displayed. If the child ignores the error depicted by the icon and scans the bar code label on the item, the error is cleared. This is because the hardware does not differentiate between input from the keyboard and input from the scanner. After the error is cleared, the system redisplays the input item ID icon (Figure 3.4) and treats part of the second scan as an item scan. This results in another error similar to the previous one. This problem was solved in the second version of CHILDS by introducing an audio cue (a beep) to attract the child's attention.

The third problem can occurr when a child attempts to borrow an item after reaching the loan limit. In this case, the too-many-books icon (Figure 3.7) is displayed. Some children actually interpreted the icon literally (i.e., they thought that their loan limit was five items rather than two). This problem led to an important principle in designing computer interfaces for children which will be discussed in more detail in Chapter 7.

The fourth problem occurred due to the fact that the search procedure was text oriented. That is, the first version of CHILDS required recognition level of abstraction [Szaf87] for searching. The search interface in the second version of CHILDS was completely re-designed. It provides graphical cues as well as textual cues. In some cases, it reduces the level of abstraction from recognition to conceptualization. Thus children in kindergarten and grade one can perform some limited item searches.

Chapter 4.

System Requirements.

The system requirements for the second version of CHILDS are based on functional requirements developed in consultation with the school librarian, interface requirements neccessary to ensure the system's accessibily to the children, and hardware requirements imposed by the school's budget and its existing equipment.

4.1 Functional Requirements.

The functional requirements are divided into four groups: loan/return, search, data entry/update and inventory. The loan/return and search capabilities must be available to all users (children, teachers and library staff). The data entry/update and inventory capabilities need only be available to library staff (the librarian and parent volunteers).

Each item must include the following information: a unique item ID number, a title, an authors list, a subjects list, a publisher, a year of publication, an ISBN number, a price, a unique dewey number, a current borrower ID number, a current loan date, an item type, a loan count and a current loan status. Each user must include the following information: a unique user ID number, a user name, a class name, a user type, a loan limit and a list of borrowed items.

The system must be capable of storing at least 10,000 items and serving at least 500 users (to allow future growth). The loan/return procedures must include the following capabilities:

1. Loan by unique item ID and unique user ID.
2. Return by unique itemID.
3. Support different user classes with independent loan limits.

The search procedure must include the following capabilities:

1. Display the unique item, given the item ID.
2. Display zero or more items, given one or more title keywords.
3. Display zero or more items, given the surnames of one or more authors.
4. Display zero of more items, given one or more subjects.
5. Provide a printed record of any of the above.

The data entry/update procedures must include the following capabilities:

1. Enter a new item or a new user.
2. Delete an existing item or an existing user.
3. Update the information of an existing item or an existing user.

The inventory procedures must include the following capabilities:

1. Display the unique user, given the user ID.
2. Display zero or more users, given a user name.
3. Display all users in the school sorted by class.
4. Display all users in the school sorted by user ID.
5. Display all users in the school sorted by surname.
6. Display all users in a class sorted by surname, given a class name.
7. Display all users within an ID range , given the range of IDs.
8. Display the unique item, given the item ID.
9. Display all (popular) items checked out at least N times, given N.
10. Display all (unpopular) items checked out not more than N times, given N.
11. Display all (overdue) items with loan date older than a given date.
12. Display all items loaned to a particular user, given a user ID.
13. Provide a printed record of any of the above.

4.2 Interface Requirements.

All loan/return operations must use graphical cues instead of textual ones so that all school children can understand the directions and perform loans and returns without any adult supervision. In addition, the system must be capable of loaning and returning 50 books in less than 10 minutes. This is due to the fact that children arrive in classes of about 25 students, with two books per student during a library period, and only 10 minutes are allowed for loans and returns at the end of the library period.

As mentioned in section 2.2, there are three abstraction levels involved in an item search using a card catalogue: conceptualization, recognition, and categorization [Szaf87]. Therefore, the text used in the search procedures must be minimized to reduce the level of abstraction from categorization to recognition. This should allow children in the second and higher grades to perform searches. In addition, some searching capabilities should be included to allow children in kindergarten and the first grade to perform limited searches.

The data entry/update procedures may use textual rather than graphical cues since they will be used by adult library staff. However, the interface must be simple and consistent since most of the users are computer novices.

4.3 Hardware Constraints.

The hardware used for the first version of CHILDS was retained for the second version since the school has very limited financial resources. The equipment consists of:

1. an IBM^{TM} PC with 256KByte of main memory;
2. a 10 MByte hard disk;
3. a 360 KByte floppy disk drive;

4. a Caere™ bar code scanner;
5. an Amdek™ Video-300A monochrome graphics monitor;
6. an Epson™ printer.

Although more suitable equipment is available today, financial and other factors often exist in real world situations which constrain system developers.

Chapter 5

Solution Strategy.

The system requirements specified in the previous chapter were satisfied by applying the following three fundamental principles:

1. Users must only be required to supply the minimum data necessary to accomplish a task.
2: This data must be easy for the users to enter.
3. Item and user information must be integrated and organized for easy access by multiple keys.

5.1 Minimum Data.

The loan operation requires two pieces of information: a unique item ID and a unique user ID. The return operation requires only one piece of information: a unique item ID. This data minimization is essential to ensure usability by the youngest children. This principle does not restrict the form of these IDs except that they must be unique (they need not be numerical).

The item search operation requires one or more words from the title, the surname of one or more authors, or one or more subjects. In general, this operation requires only recognition level abstraction [Szaf80]. However, there are some special cases in the second version of CHILDS where conceptualization level abstraction is sufficient. These cases will be described later.

The data required by the inventory control operations depends on the individual operation. For example, generating class lists for the whole school does not require any input data, whereas generating a single class list requires the class name. Although the intended users for the inventory control operations are the library staff, data minimization is still an advantage because most of them are computer novices.

5.2 Data Entry.

To eliminate typing errors, a bar code scanner is included in the system. Many kindergarten and grade one children would not be capable of using the loan and return operations if item IDs and user IDs had to be entered using a keyboard. The probability of typing errors is too high and the process would be too slow to meet the 50 books in 10 minutes requirement. Users are issued cards with bar code labels which represent user IDs and all items have bar code labels which represents item IDs. Integer IDs were chosen in both cases for simplicity. Although names could be used for users, the librarian preferred numbers since the user library cards could then be re-used.

A bar code scanner is essential but not sufficient to ensure the system's usability by the youngest children. The interface must use graphical cues instead of textual ones to guide the children throughout the system. The interface for the loan and return operations must be completely graphical and the amount of text used in the interface for the search operation must be minimized.

5.3 Information Integration.

The four sub-systems (loan/return, search, data entry/update and inventory) must share common information. The information may be organized in a custom designed file system or a custom data base system. Either approach can provide the simple queries required for the system. A commercial data base could not be used since it could not be integrated with the custom interface. A file system was chosen for two reason. Firstly, it is more space efficient, so a smaller hard disk is required. This is very important for elementary schools because of their limited budget. Secondly, it could be designed to optimize the transactions (loan/return) which must be fast.

5.4 System Structure.

The system can be naturally decomposed into four separate sub-systems according to the functional requirements: loan/return, search, data entry/update, and inventory. This suggests that four separate programs which operate on a common file system are required. The reason for using separate programs instead of one single program is to reduce the amount of main memory which is required. This reduction is essential due to the 256 KByte memory restriction.

However, since users often want to intermix loan and search operations, it is quite reasonable to combine the loan/return and search sub-system into a single sub-system. Therefore, the system is divided into three sub-systems: loan/return and search, data entry/update, and inventory.

The design of each of the three programs (loan/return and search, data entry/update, and inventory) is divided into two parts: the file system and the user interface. The file system is fairly stable since system capabilities are essentially fixed. The user interface, on the other hand, changed with user feedback.

Chapter 6.

File System Design.

The failure of the first version of CHILDS showed that it is very important to have well designed data structures to support a software system. In the design phase of the second version of CHILDS, a lot of time and attention has been spent on the file system design.

6.1 Data Abstraction.

The file system of the second version of CHILDS was designed based on the concept of data abstraction. Data abstraction is a powerful concept in software design [Fair85,Ford85]. An essential feature of data abstraction is information hiding, which is the ability to define the essential concepts without specifying the implementation details. This ability is very important because it allows the implementation of a data type to be changed at any time. This is a definite advantage as some of the abstract data types were re-implemented along the way. In addition, applying data abstraction results in a cleaner and better organized software design for easier testing and simplified maintenance.

6.2 What is an Encoded String?

The ultimate goal of the file system is to store information about every item and user. Therefore, the file system consists of two master files, the master item file and the master user file. Each master item record contains all the information about a specific item (Figure 6.1).

Title
Authors List
Subjects List
Publisher
Year
Price
ISBN
Dewey
Type
Status
Borrower ID
Loan Date
Loan Count

Figure 6.1 A Master Item Record.

Each master user record contains all the information about a specific user (Figure 6.2).

Last Name
Other Name
Class
Type
Loan Limit
Borrowed Items

Figure 6.2 A Master User Record.

In order to incorporate all the information in a compact manner, an integrated design was required for the file system. One of the observations made from studying the library environment was that there are multiple occurrences of common words (character strings). For example, many items have "DOG" as one of the keywords in the title. It is therefore reasonable to keep one master copy of every character string, to **encode** each of these character strings, and to use the encoding in place of the string wherever it is needed. The second version of CHILDS uses a hash table and a hash function to encode every distinct character string as a unique positive integer. Since each positive integer requires two bytes of storage and most character strings are longer than two characters, this resulted in substantial savings in storage.

As well as storing character strings, the file system must also support searching by character strings. The encoding scheme is augmented by maintaining a reference list of the item numbers of all the items which contain the encoded string. Since there are cases where certain types of searching does not make sense, not all encodings require a reference list. For example, searching by a user's given name is one of these cases since a user's given name is not always unique. Separate encodings are maintained for the titles, authors, and subjects since the system supports three different search types. For example, if a user wants to search on all the items written by BROWN, he does not want items with the keyword BROWN in the title to appear in the search list.

6.3 Abstract Data Types.

The abstract data types are constructed hierarchically. That is, each major abstract data type is composed of several more basic ones which are composed of even simpler ones. To understand this structure, it is necessary to discuss the simplest ones first.

6.3.1 Stream.

A Stream is a random access file of records. Each record can be defined to be of any form and size. Only the record number is needed to access a record.

6.3.2 List Array.

A List Array is a random access collection of singlely linked lists stored ~~in a Stream~~. Figure 6.3 shows the structure of a List Array.

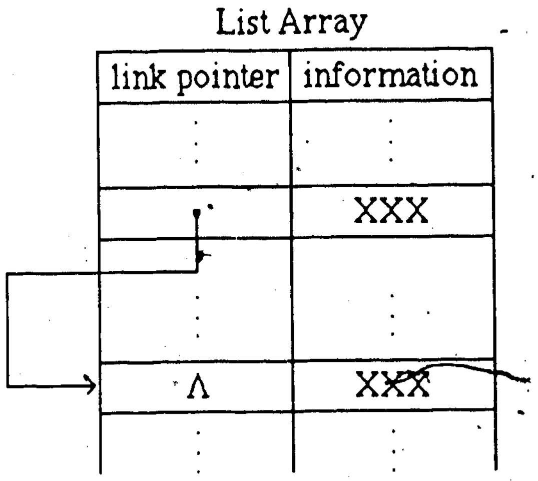


Figure 6.3 Structure of List Array.

Each record in the List Array consists of a link pointer that points to the next record in the linked list and an information field which can contain literally any type of data. The size of a List Array is dynamic (i.e., it can grow when it is full). The List Array is indexed by positive integers which are the record numbers of the first record in each of the lists contained in the array. Note that not all positive integers are valid indexes into the List Array since they do not necessarily correspond to the first record in a list. When a new list is added to the List Array, its index is returned so it can be stored and accessed at a future time.

6.3.3 String Table.

A String Table is a Stream of characters. String Tables are needed to keep a master copy of every character string. Figure 6.4 shows an example of a String Table.

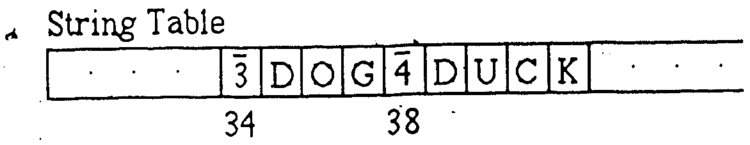


Figure 6.4 Structure of String Table.

The first element before each character string in the string table specifies the length of the character string. The bar on top of the length indicates that this particular element is a length rather than a character. This can be done by turning on the most significant bit to indicate a length since the characters are stored in ASCII (Note that only the lower 7 bits are used for characters in ASCII).

The idea of using a bit to indicate the first byte of a character string is an example of an important issue concerning abstract data type design: **An abstract data type should be able to identify and correct (if possible) errors in its own structure.**

For example, assume the information in location 38 is erased by an i/o error and replaced by some character, say an "A". If we ask the string table to give us the character string starting at location 38, then the string table can return an error message because location 38 does not have a one as its most significant bit (i.e., the string table knows that location 38 is not a length) and therefore there is an error in the string table. Without this mechanism, it would blindly interpret the "A" as a length of 65 (the ASCII code for an "A") and return a character string of this length.

One possible solution to correct this error is to step back through the string table one character at a time until we hit a length (location 34) and use that length (3) to step through the previous character string (DOG). Now, we are at location 38 and we know that location 38 should be a length, so we count the number of characters in the character string starting at location 39 and it turn out to be 4 (because location 43 is a length). Now, we can go back to location 38 and put in a bar and a 4.

If the length indicated by location 34 is wrong (say 5), then this solution will not work. Therefore, while it is relatively easy to identify errors, it is quite difficult to correct the errors. Nevertheless, this example illustrates the importance of the abilities to identify and correct errors within an abstract data type.

6.3.4 Encoded String.

An encoded string is implemented using three other abstract data types: a String Table, a Stream, and a List Array. The String Table is needed to store a copy of every character string. The Stream is used as a hash table. The List Array is used as a collision list to resolve collisions within the hash table. Figure 6.5 shows an example of an Encoded String. In this example, the character strings DOG and CAT both hashed to 25 but DOG is encoded as 31 and CAT is encoded as 45.

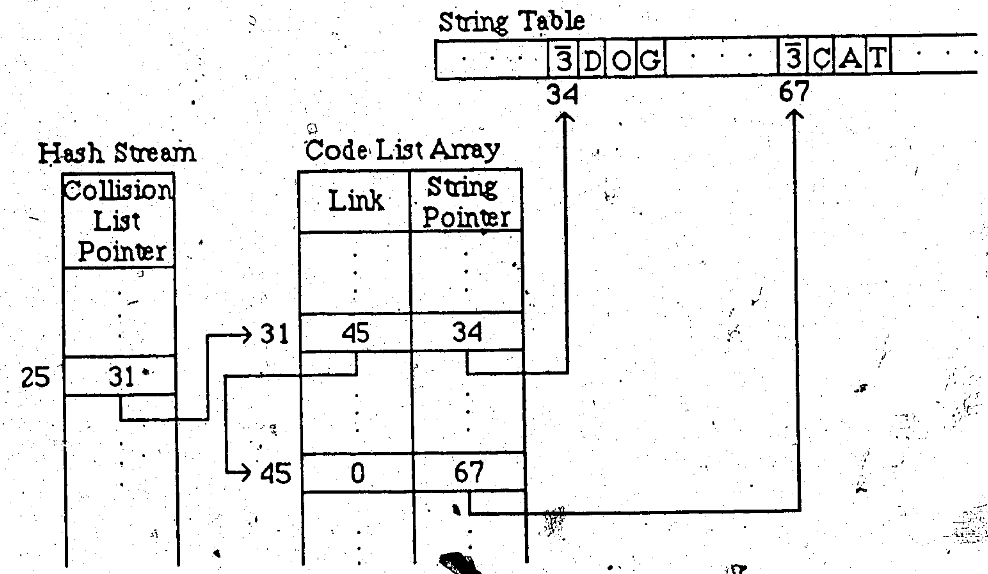


Figure 6.5 Structure of Encoded String.

Each entry in the Hash Stream (Stream of non-negative integers) contains a pointer that points to the first record of the collision list in the Code List

Array. Each record in the collision list consists of a pointer that points to the next record in the collision list and a pointer that points to the corresponding character string in the String Table.

Note that the index of the record in the List Array is used as the encoding of the character string rather than the index of the record in the Hash Stream (i.e., the hashed value) because the hashed value is not unique. Note also that this encoding scheme can be changed as long as every character string is encoded to a unique form (they need not be numerical). This is the essence of data abstraction. Encoded Strings can be used by higher level constructs without the knowledge of this implementation.

6.3.5 Referenced Encoded String.

Since the file system has to support rapid searching, it is necessary to keep a reference list for those character strings which are going to be searched. A Referenced Encoded String is used to accomplish this task. Figure 6.7 shows the structure of a Referenced Encoded String. In this example, item number 1000 and 2000 have references to DOG whose encoding is 31.

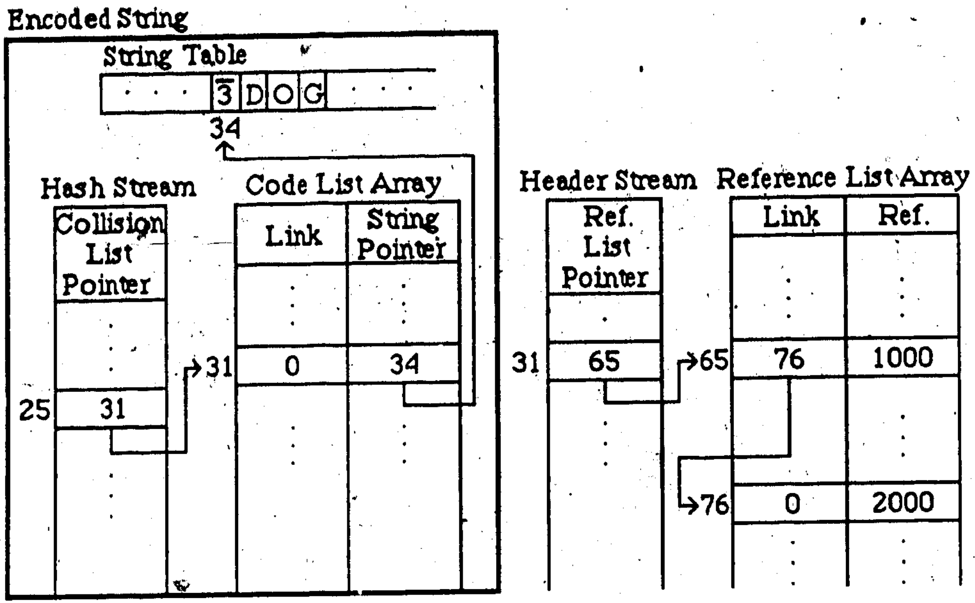


Figure 6.6 Structure of Referenced Encoded String.

As shown in Figure 6.6, a Referenced Encoded String is constructed by combining an Encoded String, a Stream and a List Array of references. The Header Stream is "parallel" to the Code List Array in the Encoded String. Each entry in the Header Stream points to the first record of a reference list in the Reference List Array. It is therefore correct to view the Header Stream as another column in the Code List Array (i.e., every record in the Code List Array would contain a third field). Each record in the Reference List Array consists of a pointer that points to the next record in the reference list and a pointer that points to the corresponding reference (item ID).

Note that if more than one character string is hashed to the same value (i.e., the collision list has more than one element), a linear search through the collision list is required to find the reference list of a given string. This is quite acceptable if the average length of the collision list is small. This problem is discussed in more detail in section 6.3.12.

There are cases where certain words in the title are too common to be used in a search. For example, it does not make sense to perform a search on all the items with the word THE in the title. It is therefore inappropriate to keep a reference list for the common words. Therefore, the entries in the Header Stream for all common words are specially marked to indicate that they are common words.

6.3.7 Encoded String List.

There are some cases where a list of character strings is needed. For example, assume that the name of a user is "JOHN A. DOE". The character string DOE is stored in a Referenced Encoded String since the system supports searching for users by their last names. However, it is also desirable to have both JOHN and A. encoded separately since JOHN and A. are common names but JOHN A. is not. An Encoded String List is used to accomplish this task. Figure 6.7 shows the structure of an Encoded String List.

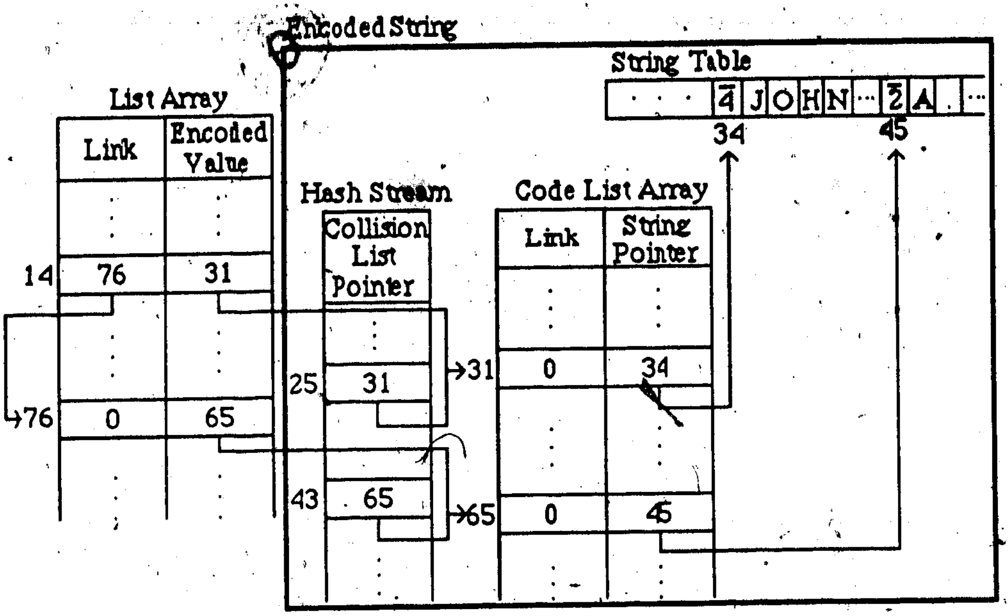


Figure 6.7 Structure of Encoded String List.

As shown in Figure 6.7, an Encoded String List is constructed by combining a List Array and an Encoded String. Each record of the List Array consists of a pointer that points to the next record in the list and a pointer that points to the corresponding record in the Code List Array.

6.3.7 Referenced Encoded String List.

There are cases where a list of character strings with references is needed. For example, if the title of an item is "NEW BOOK", then both NEW and BOOK have to be encoded separately but the title must be stored as "NEW BOOK". A Referenced Encoded String List is used to accomplish this task. Figure 6.8 shows the structure of a Referenced Encoded String List.

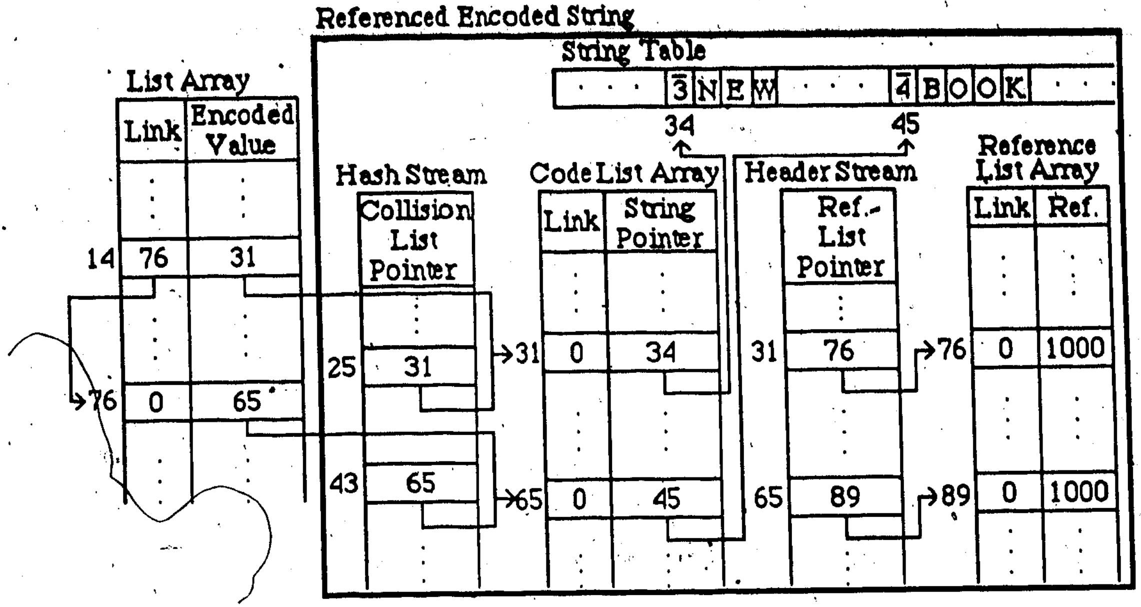


Figure 6.8 Structure of Referenced Encoded String List.

As shown in Figure 6.8, a Referenced Encoded String list is constructed by combining a List Array and a Referenced Encoded String. However, it can also be constructed by combining an Encoded String List, a Header Stream and a Reference List Array. The decision is an implementation one rather than a logical one since data abstraction hides it. The first combination is simpler so it was used in the implementation.

6.3.8 Authors List.

Since many items have more than one author, another abstract data type was required to represent them. Figure 6.9 shows an example of an Authors List. In this example, JOHN A. DOE and JACK SMITH are both authors of the same item.

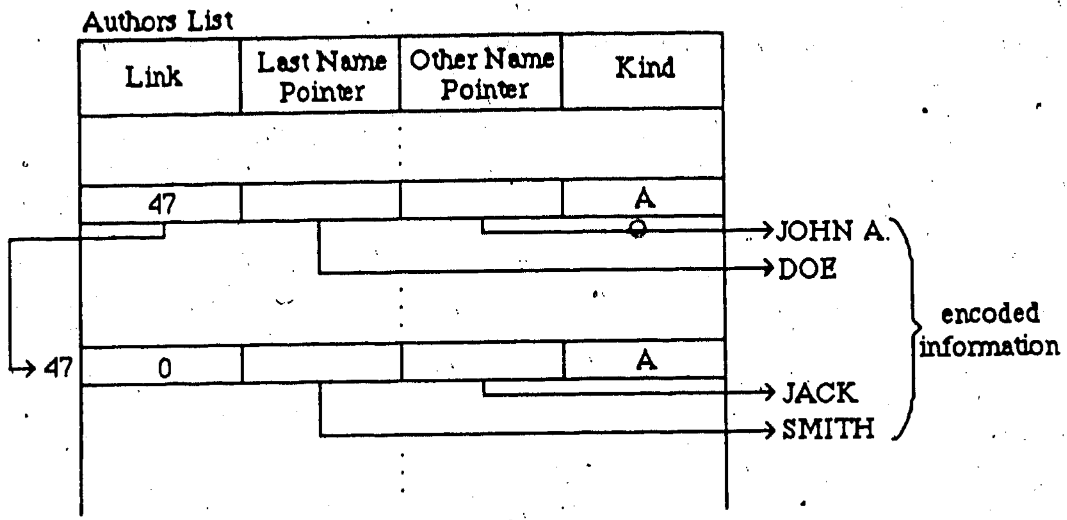


Figure 6.9 Structure of Authors List.

Each Record in the Authors.List consists of a link pointer that points to the next author, a last name pointer that points to the author's last name (a Referenced Encoded String), an other name pointer that points to the author's other name (an Encoded String List), and a character which indicate the author's kind (author, illustrator, or editor).

6.3.9 Loan List.

A loan list is needed to represent the items that each user has borrowed. However, a new abstract data type is not needed since a List Array can be used. Figure 6.10 shows the structure of a Loan List. In this example, items 1000 and 2000 are loaned to the same user.

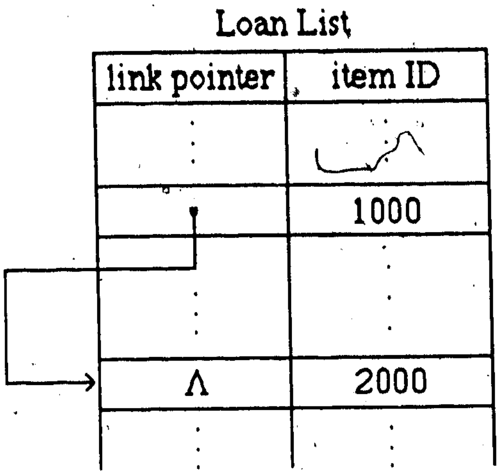


Figure 6.10 Structure of Loan List.

Each record consists of a link pointer that points to the next borrowed item by the same user and the item ID of the borrowed item.

6.3.10 Master Item.

As described previously, a master item file is a file of master item records. Each master item record contains all the information about an item. The item ID is used as the record number within the file. Searching by ID is simply a single random access into the file. Figure 6.11 shows the complete structure of a master item record.

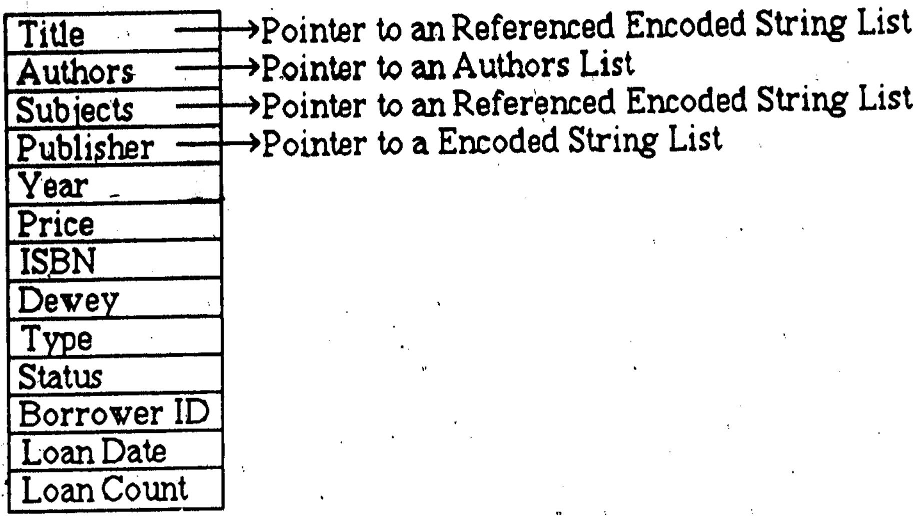

Title
Authors
Subjects
Publisher
Year
Price
ISBN
Dewey
Type
Status
Borrower ID
Loan Date
Loan Count
Pointer to an Referenced Encoded String List
Pointer to an Authors List
Pointer to an Referenced Encoded String List
Pointer to a Encoded String List

Figure 6.11 Structure of Master Item Record.

As shown in Figure 6.11, the title, authors, subjects, and publisher fields of a master item record are pointers that point to other abstract data types. All the other information is stored directly in the master item record. This makes the master item file quite compact as compared to the one in the first version.

6.3.11 Master User.

The master user file is a file of master user records. Each master user record contains all the information about a user. The user ID is used as the record number within the file, so a search by ID is a single random access. Figure 6.12 shows the complete structure of a master user record.

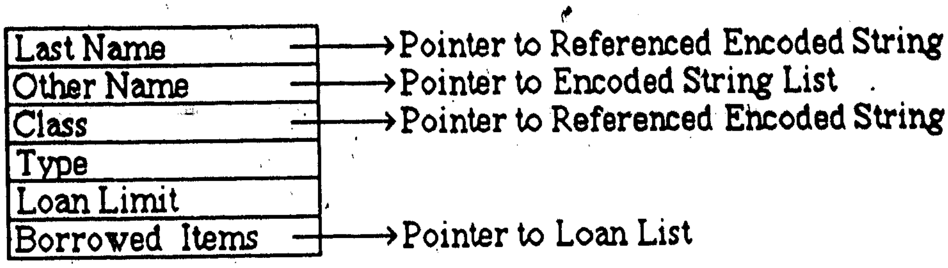

Last Name
Other Name
Class
Type
Loan Limit
Borrowed Items
Pointer to Referenced Encoded String
Pointer to Encoded String List
Pointer to Referenced Encoded String
Pointer to Loan List

Figure 6.12 Structure of Master User Record.

As shown in Figure 6.12, the last name, other names, class, and borrowed items fields of a master user record are pointers that point to other abstract data types. All the other information is stored directly in the master user record. Again, this makes the master user file more compact than in the first version.

6.3.12 Implementation Notes.

Due to the limitation of the maximum number (15) of files which can be opened at any time on the IBM^{TM} PC, there are a few adjustments in the implementation of an Encoded String. The Hash Stream and the Code List Array of the Encoded String were combined into one structure called a Code Table. Figure 6.13 shows the structure of this implementation. In this example, the character strings DOG and CAT are both hashed to 25 but DOG is encoded as 31 and CAT is encoded as 45.

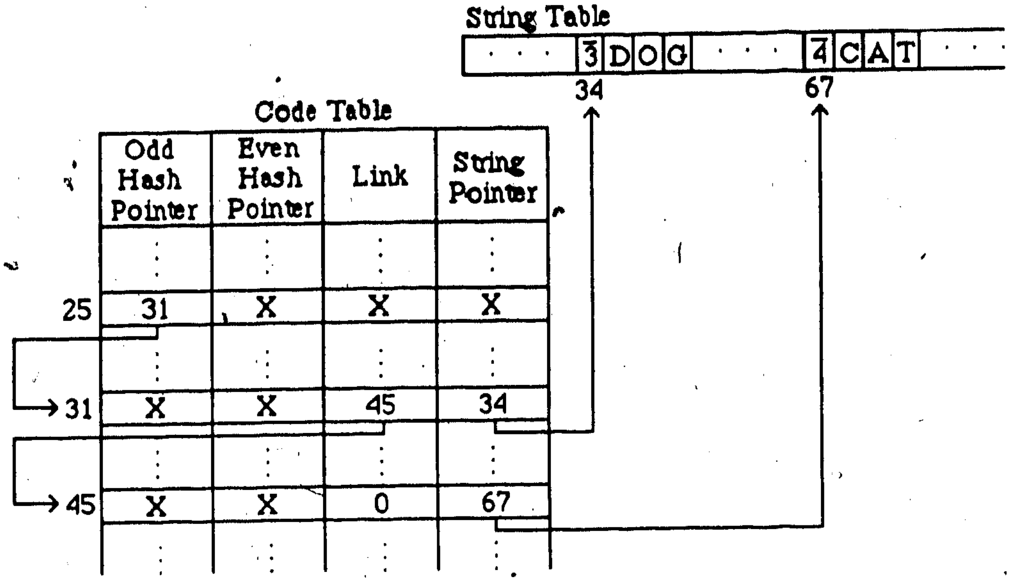


Figure 6.13 Implementation of Encoded String.

If the hash value is 50, the even hash pointer of record number 25 is used. This structure makes the size of the hash table twice the size of the Code Table.

While the original design of the Encoded String requires the size of the hash table to be fixed, this implementation allows the size of the hash table to be enlarged if the code table is full. This increases the efficiency of the searches since the size of the hash table will always be at least twice the number of hashed items. By reducing the number of collisions, the average length of the collision lists is smaller, so the search time is reduced. Figure 6.14 shows the average length of the collision lists for each of the Code Tables for 6000 items.

Code Table	Average length of collision lists
Title	1.277
Author's last name	1.344
Author's other names	1.311
Subject	1.385
Publisher	1.255

Figure 6.14 Average length of collision lists.

As shown in Figure 6.14, the average length of the collision list is slightly greater than one. Therefore, the time required for searching is relatively small.

On the other hand, enlarging the hash table requires all the items to be re-hashed. This is a definite disadvantage if the size of the code table is large. However, since the number of accesses (i.e., searching) is considerable larger than the number of updates (i.e., editing), it is worth the trade off.

Chapter 7.

Capacity and Limitations.

This chapter discusses the capacity and limitations of the second version of CHILDS. Since all the indexes in the file system are encoded as 2-byte cardinal numbers (i.e., positive integers less than or equal to 65535), the number of records in a file cannot exceed 65535.

Since the number of users is relatively small as compared to the number of items in an elementary school library, the sizes of the item files will be the limiting factor. The size of each data file for items was recorded at 1000, 2000, 3000, 4000, 5000, and 6000 items to determine the growth curve for each data file and the expected capacity of the file system.

7.1 String Tables.

The data in Figure 7.1 was recorded for the growth of each String Table. The data was then plotted and exponential curves were fitted to the data. The equation of the curves were used to calculate the number of items necessary for the record count to reach its maximum value of 65535. These curves are shown in Figures 7.2 to 7.6.

Number of Items Number of Records	1000	2000	3000	4000	5000	6000
Title	9307	17112	23397	27430	32609	36816
Author surname	4010	7377	10281	12298	14965	17160
Author other names	2845	4495	5805	6606	7569	8382
Subject	4106	7569	9914	11423	14743	17367
Publisher	3130	4735	7160	8852	10758	14031

Figure 7.1 Data for growth of String Tables.

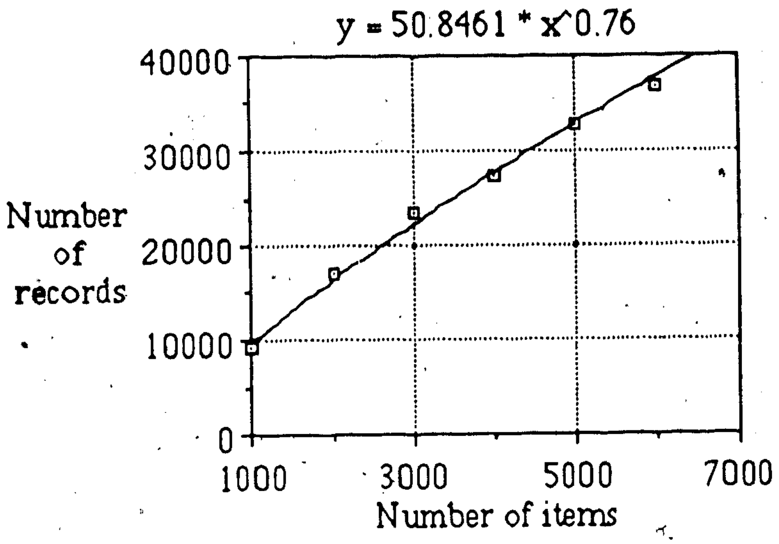


Figure 7.2 Growth of String Table for title.

When the number of records in the title String Table reaches 65535, the number of items will be 12320.

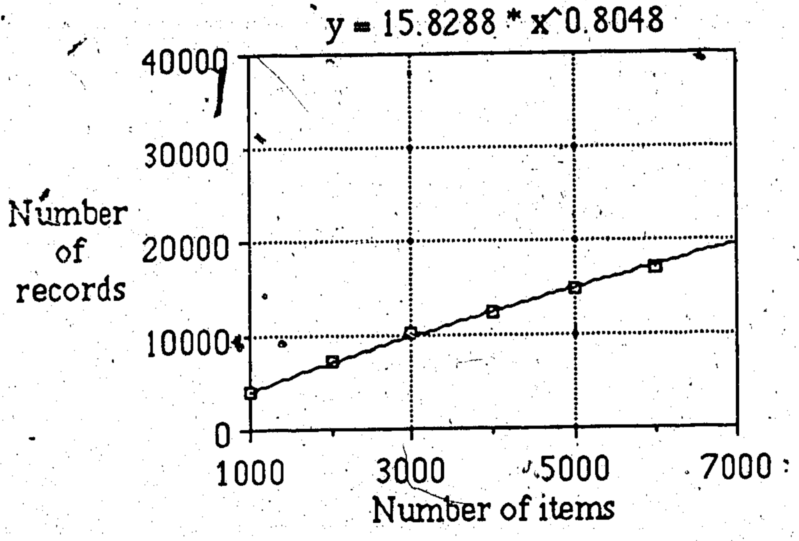


Figure 7.3 Growth of String Table for author surname.

When the number of records in the author surname String Table reaches 65535, the number of items will be 31211.

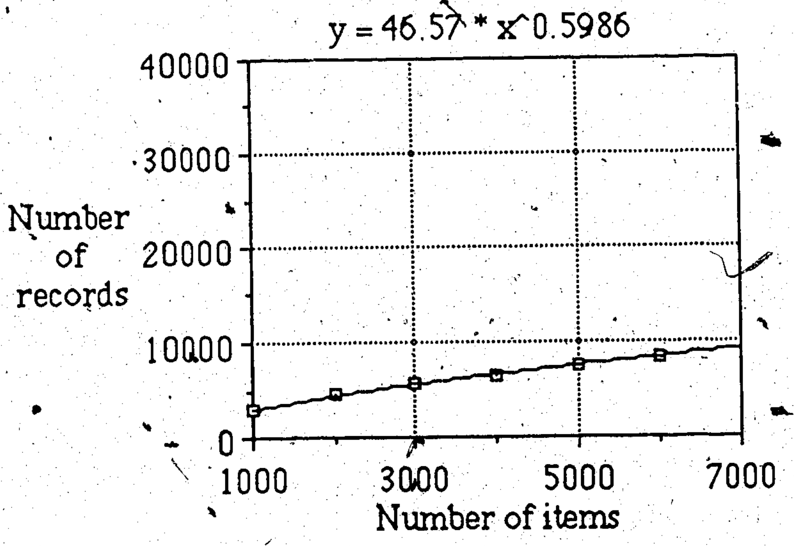


Figure 7.4 Growth of String Table for author other names.

When the number of records in the author other names String Table reaches 65535, the number of items will be 181781.

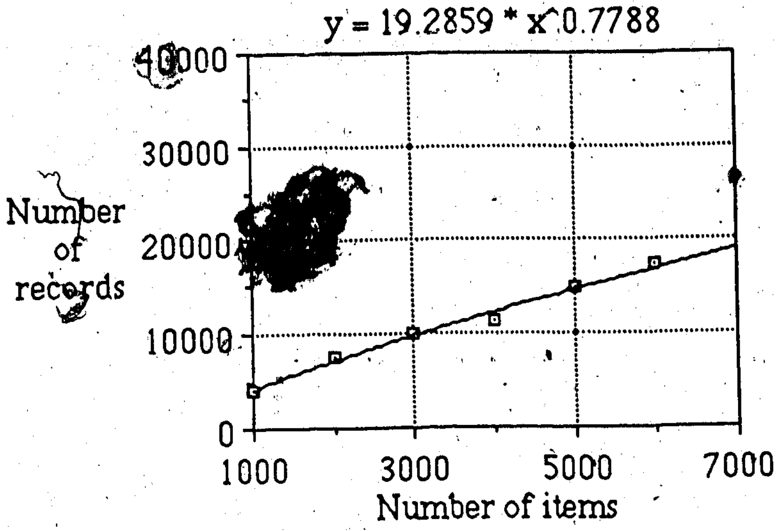


Figure 7.5 Growth of String Table for subject.

When the number of records in the subject String Table reaches 65535, the number of items will be 34213.

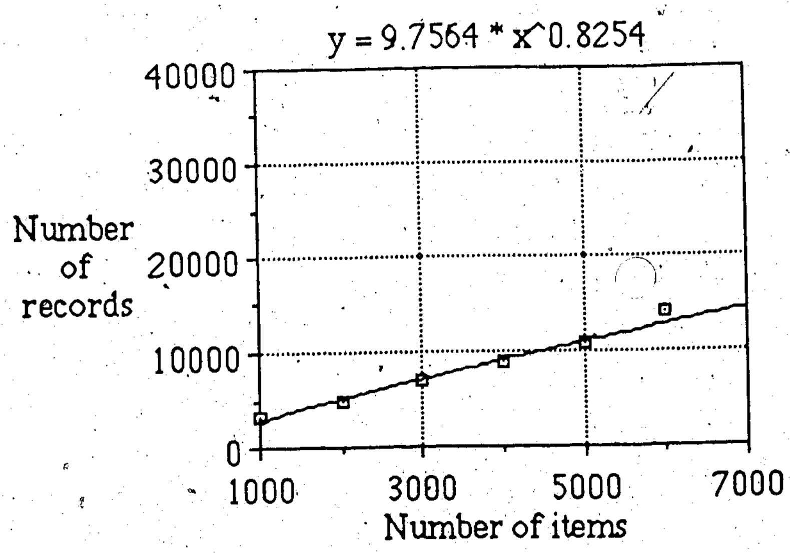


Figure 7.6 Growth of String Table for publisher.

When the number of records in the publisher String Table reaches 65535, the number of items will be 43327.

From the data recorded, the String Table for title is the fastest growing file of the String Tables and the expected maximum number of items that it can represent is 12320.

7.2 Code Tables.

The data in Figure 7.7 was recorded for the growth of each Code Table. The data was plotted and the exponential curves are shown in Figures 7.8 to 7.12. The curves were used to compute the expected maximum record counts.

Number of Records \ Number of Items	1000	2000	3000	4000	5000	6000
Title	1411	2458	3286	3833	4477	5014
Author surname	542	994	1372	1639	1987	2277
Author other names	451	706	898	1018	1160	1285
Subject	544	980	1268	1452	1868	2196
Publisher	233	353	515	625	742	915

Figure 7.7 Data for growth of Code Tables.

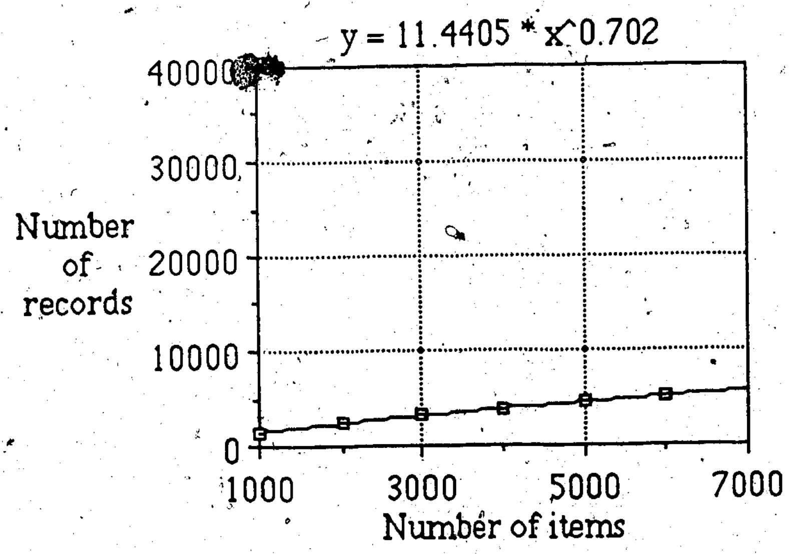


Figure 7.8 Growth of Code Table for title.

When the number of records in the title Code Table reaches 65535, the number of items will be 225588. However, since the Code Table contains a Hash Stream which is twice the size of the Code Table, the number of items will be 84026 when the number of records in the Hash Stream reaches 65535 (i.e., the number of records in the Code Table is 32677).

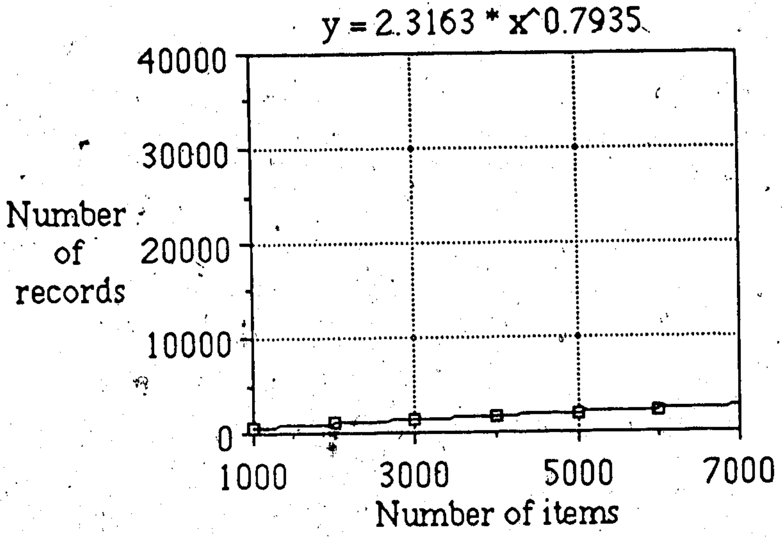


Figure 7.9 Growth of Code Table for author surname.

When the number of records in the author surname Code Table reaches 65535, the number of items will be 407549. When the number of records in the Hash Stream reaches 65535 (i.e., the number of records in the Code Table is 32677), the number of items will be 170112.

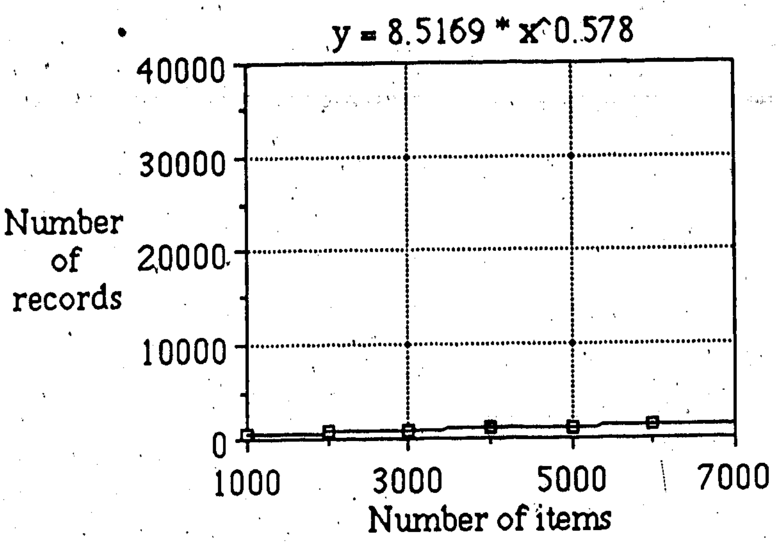


Figure 7.10 Growth of Code Table for author other names.

When the number of records in the author other names Code Table reaches 65535, the number of items will be 5290724. When the number of records in the Hash Stream reaches 65535 (i.e., the number of records in the Code Table is 32677), the number of items will be 1594405.

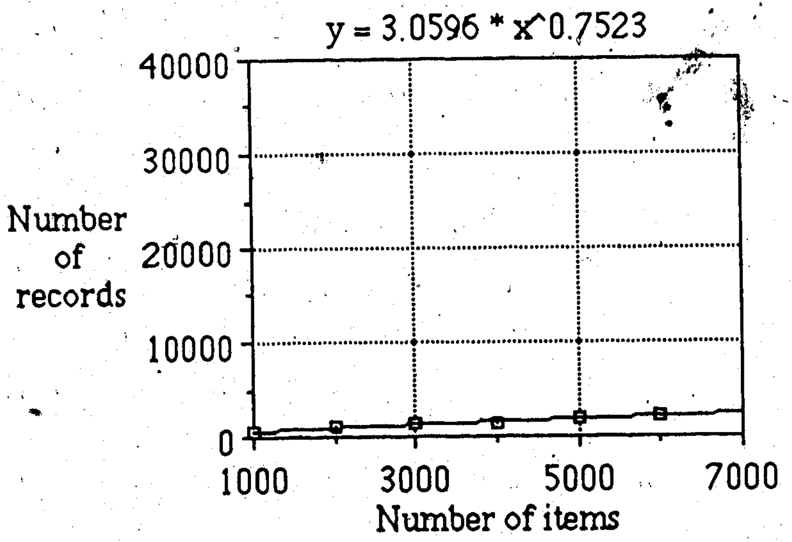


Figure 7.11 Growth of Code Table for subject.

When the number of records in the subject Code Table reaches 65535, the number of items will be 571159. When the number of records in the Hash Stream reaches 65535 (i.e., the number of records in the Code Table is 32677), the number of items will be 227264.

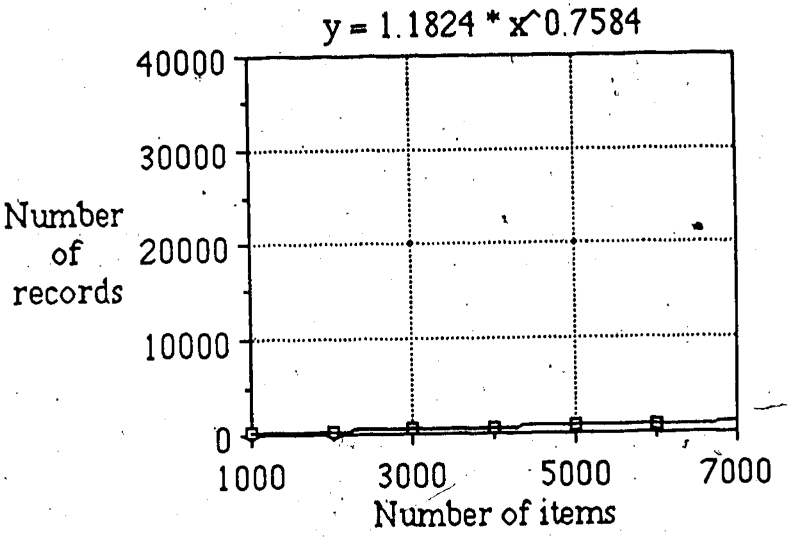


Figure 7.12 Growth of Code Table for publisher.

When the number of records in the publisher Code Table reaches 65535, the number of items will be 1798417. When the number of records in the Hash Stream reaches 65535 (i.e., the number of records in the Code Table is 32677), the number of items will be 720914.

From the data recorded, the Code Table for title is the fastest growing file of the Code Tables and the expected maximum number of items that it can represent is 84026.

7.3 Header Streams.

As mentioned before in section 6.3.5, the Header Stream is parallel to the Code Table, therefore, the data for the Code Tables also applies to the Header Streams. However, there are only Header Streams for title, author surname, and subject because the file system has to support these three search types.

7.4 Reference List Arrays.

The data in Figure 7.13 was recorded for the growth of each Reference List Array. The data was plotted and the exponential curves are shown in Figures 7.14 to 7.16. These curves were used to compute the expected maximum record counts.

Number of Items Number of Records	1000	2000	3000	4000	5000	6000
Title	2530	5060	7870	10590	13280	15900
Author surname	1230	2350	3450	4470	5540	6640
Subject	1960	4210	6230	7390	9750	11870

Figure 7.13 Data for growth of Reference List Arrays.

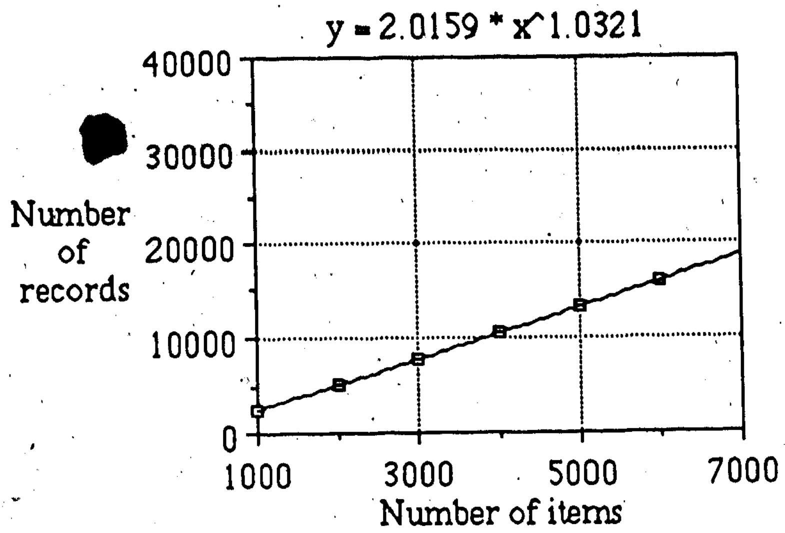


Figure 7.14 Growth of Reference List Array for title.

When the number of records in the title Reference List Array reaches 65535, the number of items will be 23532.

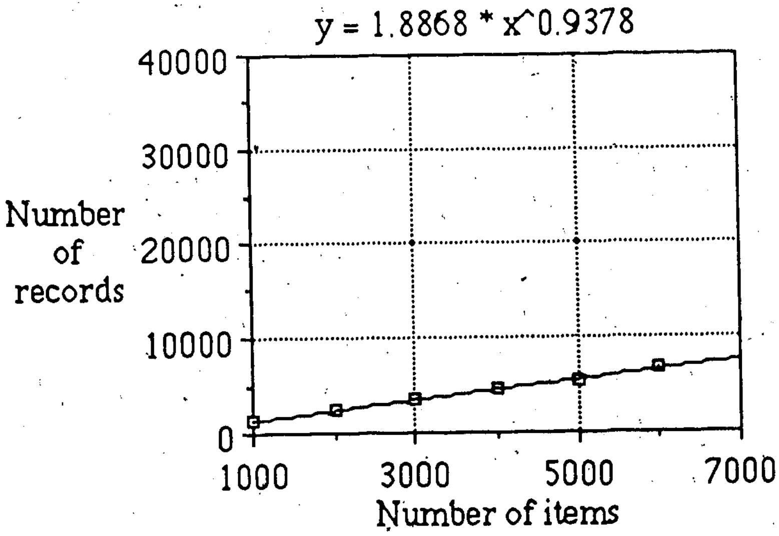


Figure 7.15 Growth of Reference List Array for author surname.

When the number of records in the author surname Reference List Array reaches 65535, the number of items will be 69488.

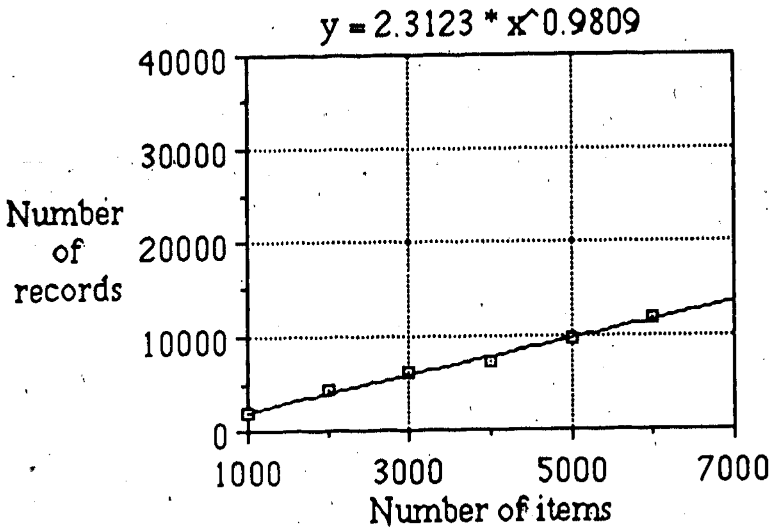


Figure 7.16 Growth of Reference List Array for subject.

When the number of records in the subject Reference List Array reaches 65535, the number of items will be 34604.

From the data recorded, the Reference List Array for title is the fastest growing file of the Reference List Arrays and the expected maximum number of items that it can represents is 23532.

7.5 List Arrays.

The data in Figure 7.17 was recorded for the growth of each List Array. The data was plotted and the exponential curves are shown in Figures 7.18 to 7.21. These curves were used to compute the expected maximum record counts.

Number of Items Number of Records	1000	2000	3000	4000	5000	6000
Title	4300	7650	11870	15850	19830	23650
Author surname	1234	2356	3445	4479	5535	6646
Author other names	1380	2730	4000	5210	6470	7680
Subject	1960	4210	6230	7390	9750	11870

Figure 7.17 Data for growth of List Arrays.

Note that the List Array for author surname is actually an Authors List.

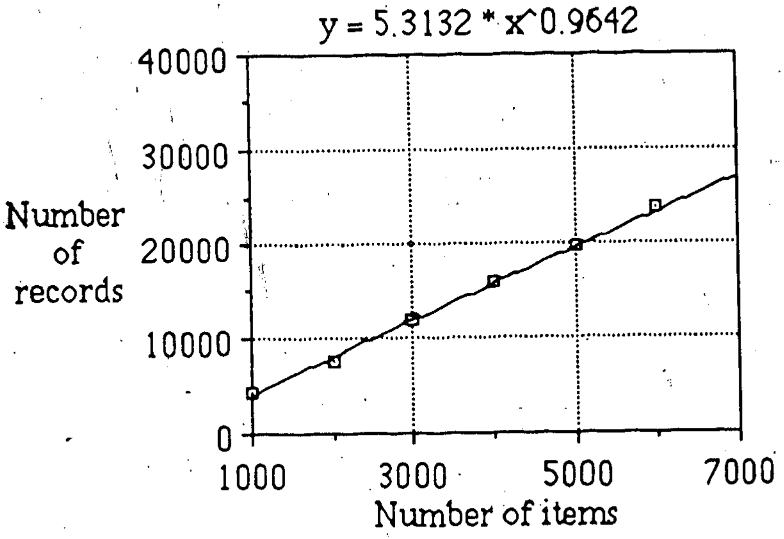


Figure 7.18 Growth of List Array for title.

When the number of records in the title List Array reaches 65535, the number of items will be 17499.

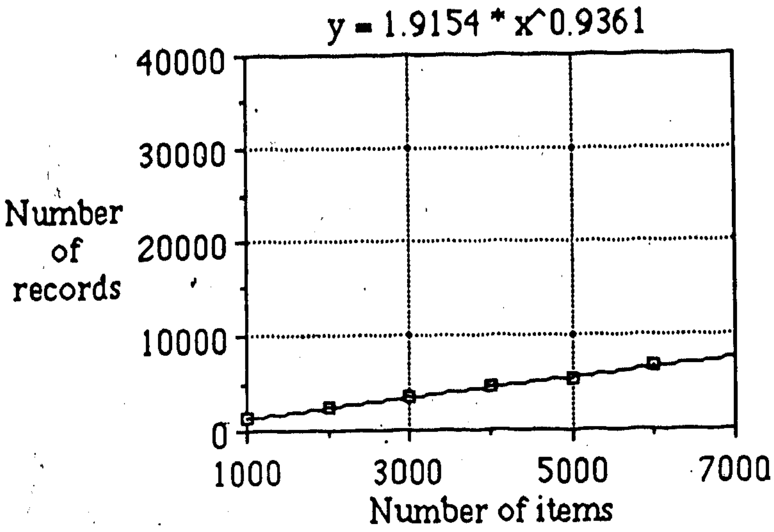


Figure 7.19 Growth of Authors List.

When the number of records in the Authors List reaches 65535, the number of items will be 69779.

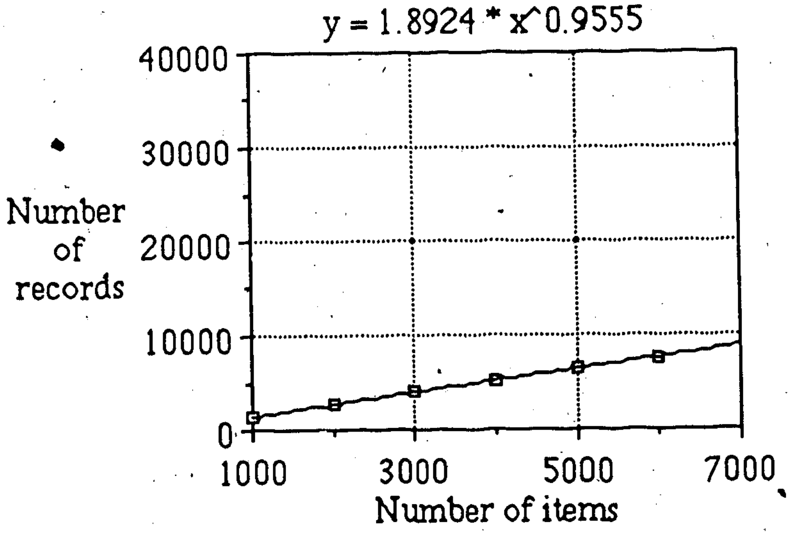


Figure 7.20 Growth of List Array for author other names.

When the number of records in the author other names List Array reaches 65535, the number of items will be 56347.

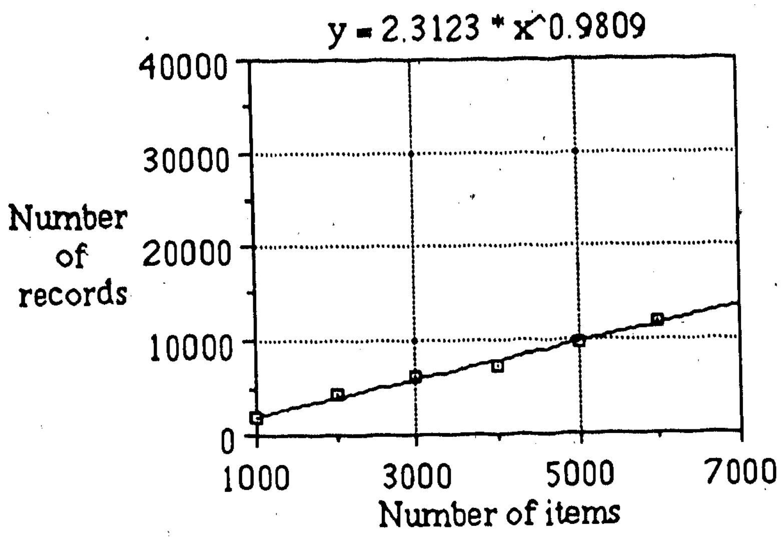


Figure 7.21 Growth of List Array for subject.

When the number of records in the subject List Array reaches 65535, the number of items will be 34604.

From the data recorded, the List Array for title is the fastest growing file of the List Arrays and the expected maximum number of items that it can represent is 17499.

7.6 Results.

From the data recorded, the fastest growing file of the entire file system is the String Table for title. It is expected to have 65535 records when the number of items reaches 12320. However, it still satisfies the system requirement of storing 10000 items.

Therefore, if the system is required to manage more than 12320 items, it is advisable to use 4-byte cardinals instead of 2-byte cardinals in the implementation of the file system. However, the size of the file system will be doubled due to the change and the savings in storage that the encoded string provides will be decreased.

Chapter 8.

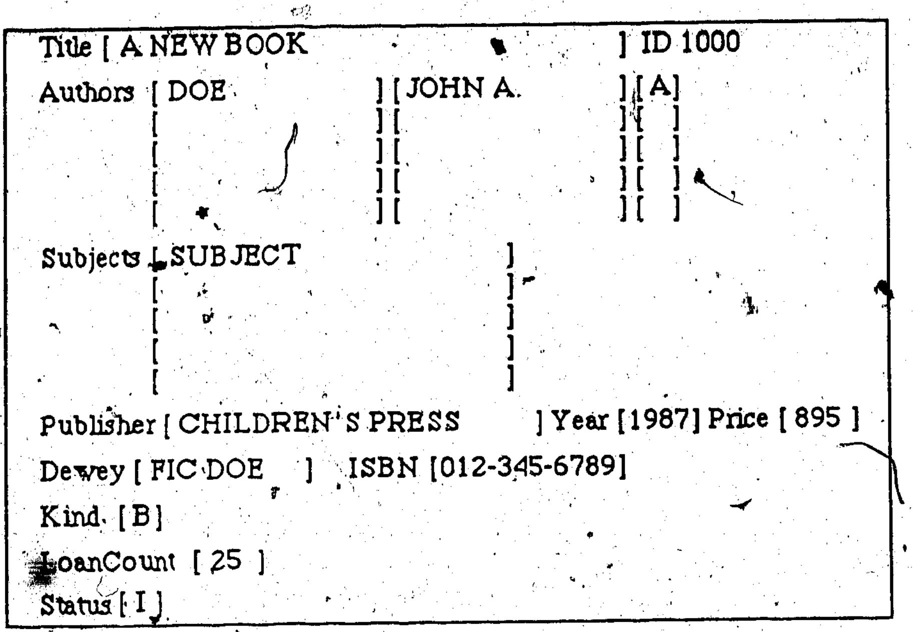
User Interface Design.

Although CHILDS is divided into three sub-systems: loan/return and search, data entry/update, and inventory, the interface design is actually divided into two parts according to the two major types of users: adults and children. The loan/return and search sub-system is graphical since it is used by children. The data entry/update and inventory sub-systems are used only by library staff, so they are more text oriented.

8.1 User Interface For Adults.

The interface designs of the data entry/update and inventory sub-systems are straightforward since they are used by adults only. The main objective is to provide a simple and consistent interface for the librarian and the parent volunteers. The users are given menus for selecting commands and templates for entering information.

Nevertheless, there are several improvements over the design of the first version of CHILDS. For example, in the first version, an item template contained only five fields for authors and five fields for subjects. Figure 8.1 shows an example of an item template.



```
Title [ A NEW BOOK                        ] ID 1000
Authors [ DOE            ][ JOHN A.        ][A]
        [                ][                ][ ]
        [                ][                ][ ]
        [                ][                ][ ]
        [                ][                ][ ]
Subjects [ SUBJECT                  ]
         [                          ]
         [                          ]
         [                          ]
         [                          ]
Publisher [ CHILDREN'S PRESS        ] Year [1987] Price [ 895 ]
Dewey [ FIC DOE  ]  ISBN [012-345-6789]
Kind [B]
LoanCount [ 25 ]
Status [ I ]
```

Figure 8.1 An Item Template.

The librarian indicated that there are some items with more than five authors (including editors and illustrators) and some items with more than five subjects. This is an illustration of an important principle of interface design: **flexibility**. An interface should not impose any restriction on the size of the information. In the present system, a user can use the PageUp and PageDown keys to scroll up and down within the authors list or the subjects list. Therefore, the system supports an unlimited number of authors and subjects.

8.2 User Interface For Children.

The first step in designing the interface of the loan/return and search sub-system is the design of the icons. In the past few years, icon design has evolved into a new area of interest in computer science. The main question pertaining to this research effort is "Which icons can be understood by children and which can not?". More research needs to be conducted to answer this question in general.

However, in the specific application of elementary school libraries, a set of understandable icons has been designed. Although there was some confusion and misinterpretation, the icons used in the first version of CHILDS were understandable by the children. Therefore, it was decided that the same type of icons with appropriate changes and improvements, would be used in the second version of CHILDS. These changes and improvements were needed for a better representation of the situation and solving some of the problems encountered by the children when using the first version. In addition, a set of fundamental principles for designing interfaces for children is postulated although experiments need to be conducted to verify these principles.

When the loan/return and search sub-system starts, the main menu (Figure 8.2) of the loan/return and search sub-system is displayed.

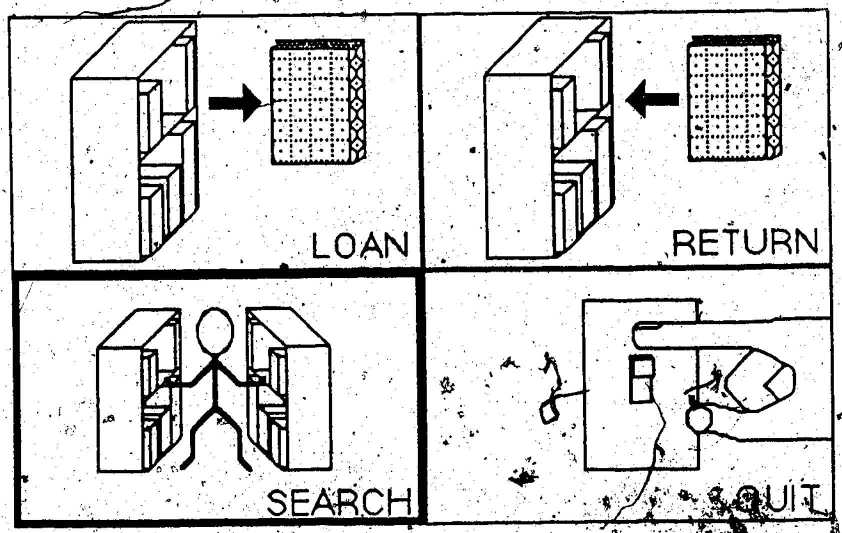


Figure 8.2 Main Menu of Loan/Return and Search Sub-system.

A user can select any one of the four panes of the menu using the cursor control keys (i.e., the up, down, right and left arrows keys). To initiate the selected function, the user presses the return key.

8.2.1 Loan (Check Out).

The procedure for borrowing an item requires two pieces of information: an item ID and a user ID. After the LOAN function is initiated from the main menu, icons of a bar code scanner and a book (Figure 8.3) are displayed.

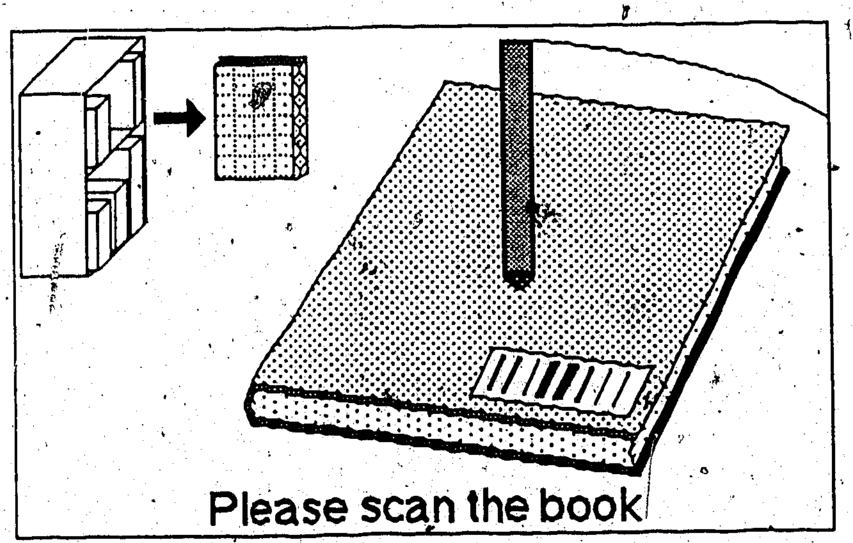


Figure 8.3 Input item ID icon in Loan.

The small icon in the top left hand corner indicates that the user is performing a LOAN rather than a RETURN. This eliminates a problem which some children had with the first version of CHILDS: sometimes a child would forget whether they were borrowing an item or returning one. This leads to the first principle in designing interfaces for children: **clarity**.

After scanning the bar code scanner across the bar code label on the item, four possible situations could happen:

1. Item is OK;
2. Item is not found;
3. Missing item is found;
4. Item is already on loan.

8.2.1.1 Item Is OK.

In the case where the item is ok, icons of a bar code scanner and a library card (Figure 8.4) are displayed.

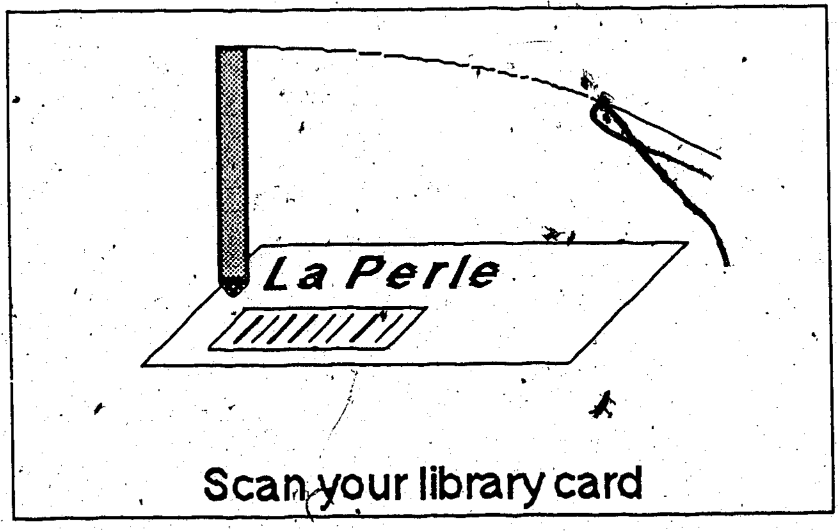


Figure 8.4 Input userID icon in Loan.

There is no need to indicate that the child is performing a LOAN because there is no similar icon for RETURN (only the item ID is needed for RETURN). After sliding the bar code scanner across the bar code label on the user's library card, one of three possible situations could happen:

1. Successful loan;
2. User has too many books;
3. User is not found.

8.2.1.1.1 Successful Loan.

A successful loan means that the item has been successfully loaned to the user. An icon of a happy face (Figure 8.5) is displayed briefly and the system returns to the input item ID icon (Figure 8.3) after about 3 seconds.

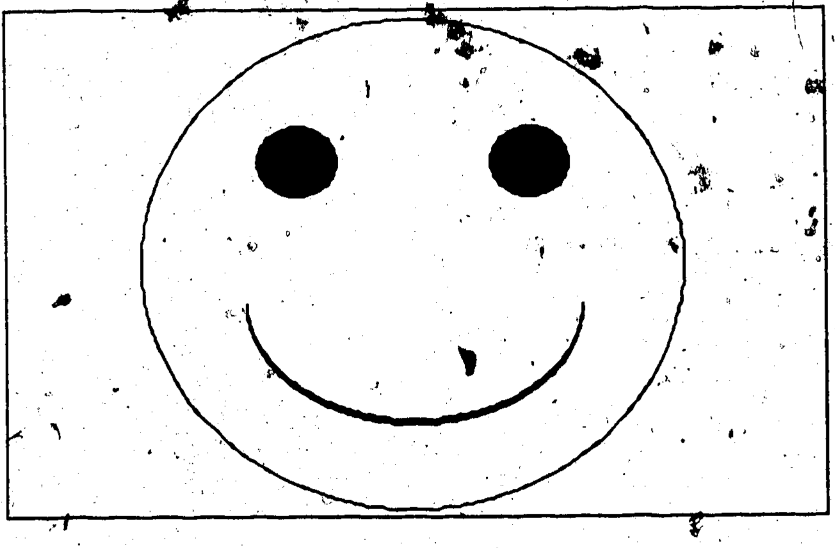


Figure 8.5 Success icon.

8.2.1.1.2 User Has Too Many Books.

If the user has reached the loan limit, an icon of three books with a cross on them (Figure 8.6) is displayed and a beep is sounded.

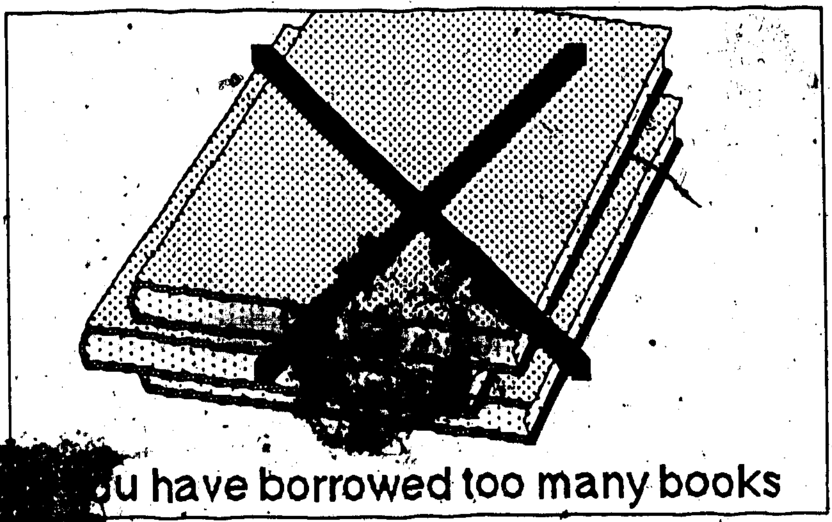


Figure 8.6 User has too many books icon.

The beep is used to attract the child's attention to the error. This feature is needed to solve the problems encountered by children who ignore error icons. This leads to another principle in designing interfaces for children: **forcefulness**. The user acknowledges and clears the error by pressing any key.

The original icon for this situation consisted of six books with a cross. However, several children interpreted the icon too literally and complained that they had not already borrowed five books. This **literalism** problem was solved for the LaPerle library, whose loan limit is two, by switching to three books. This is not a general solution however since the icon is fixed and the loan limit is not.

The icon literalism problem was reduced by displaying the ID numbers of the items already on loan and the item to be loaned, in addition to the icon.

8.2.1.1.3 User Is Not Found.

If there is no user in the library with the ID number scanned, an icon of a stick person and a question mark (Figure 8.7) is displayed and a beep is sound. In addition, the errant ID number is displayed in the stick person's head.

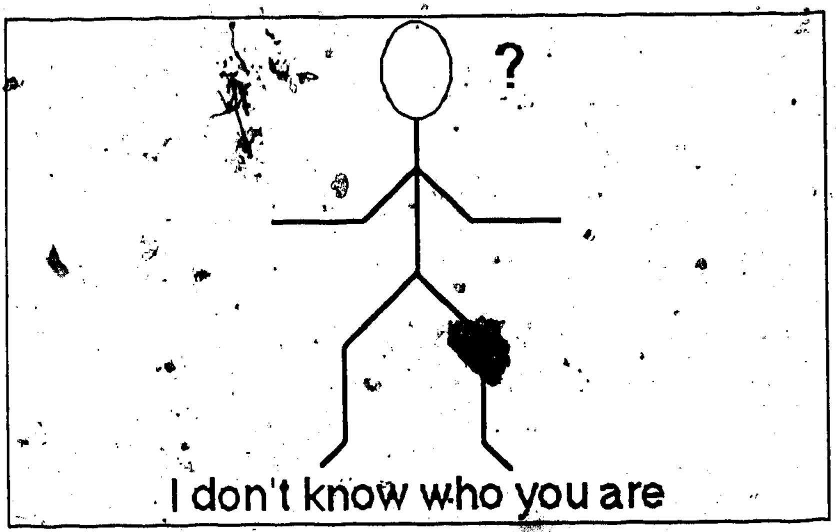
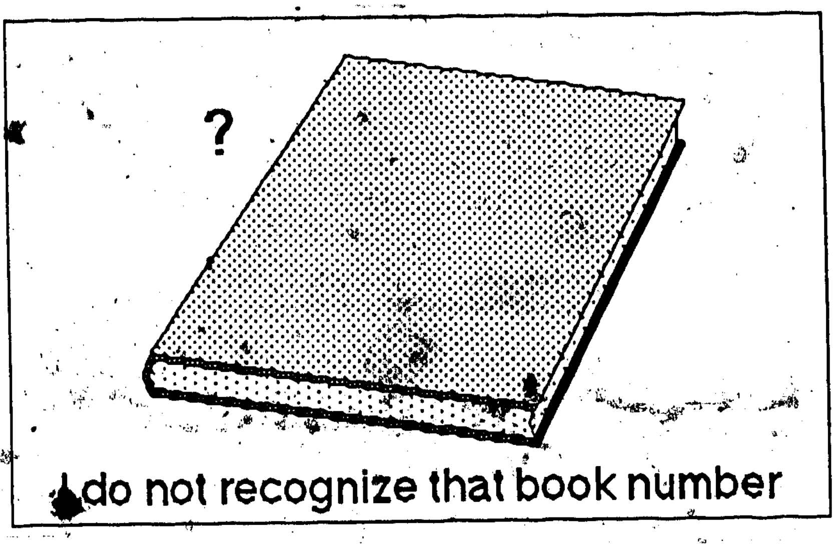


Figure 8.7 User is not found icon.

The user should ask the librarian for help in this case. This error can be cleared by pressing any key.

8.2.1.2 Item Is Not Found.

If there is no item in the library with the ID number scanned, an icon of a book and a question mark (Figure 8.8) is displayed and a beep is sound. The ID number of the item is also displayed.



I do not recognize that book number

Figure 8.8 Item is not found icon.

Again, the user should ask the librarian for help, and the error can be cleared by pressing any key.

8.2.1.3 Missing Item Is Found.

If the item which was scanned has previously been marked as missing, an icon of a stick person (the user) giving a book (the missing item) to a stick person (the librarian) is displayed (Figure 8.9) and a beep is sound. The ID number of the item is also displayed.

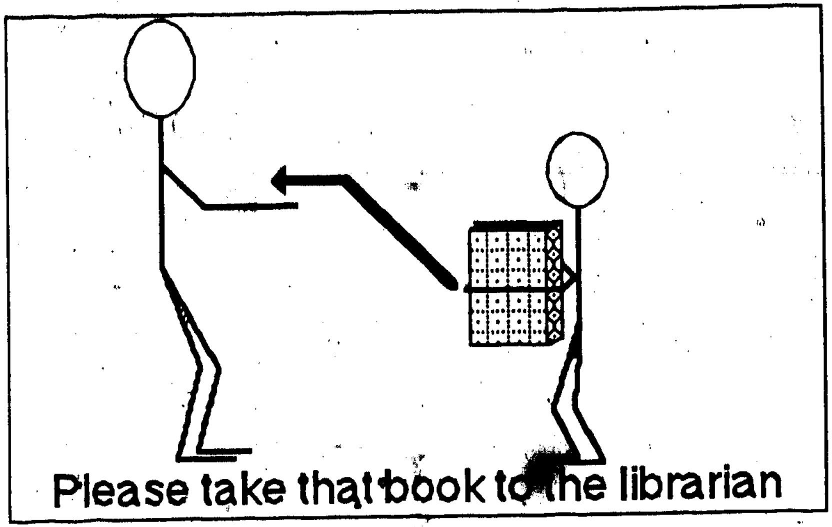


Figure 8.9 Missing item is found icon.

The user should report the missing item to the librarian and the error can be cleared by pressing any key.

8.2.1.4 Item Is Already On Loan.

If the item is already on loan, an icon of two stick persons with one of them holding a book (Figure 8.10) is displayed and a beep is sound. The ID number of the item and the user who has borrowed the item are also displayed.

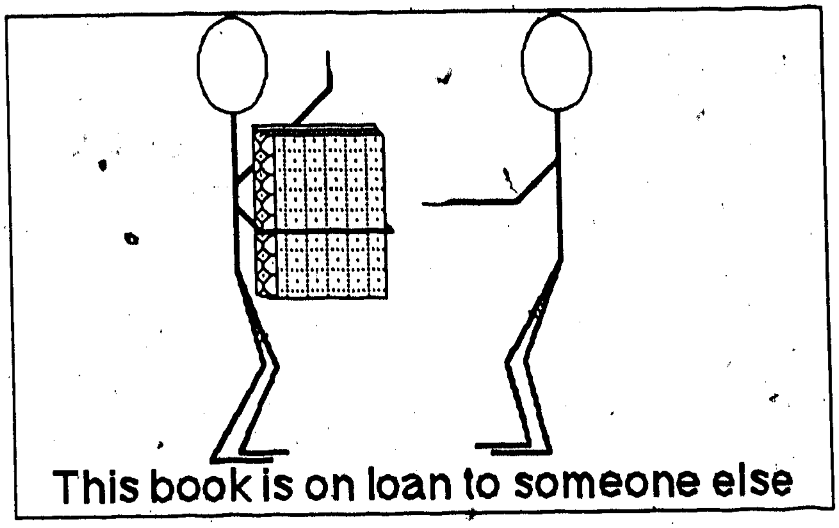


Figure 8.10 Item is already checked out icon.

The stick person holding the book represents the previous user and the other stick person represents the user who wants to borrow the item. This icon is used instead of the failure icon (a sad face) of the first version because this icon gives a better description of the situation. In this case, the user should inform the librarian about the situation and clear the error by pressing any key.

8.2.2 Return (Check In).

To return an item, only one piece of information is needed: an item ID. After selecting the RETURN function form the main menu, icons of a bar code scanner and a book (Figure 8.11) are displayed.

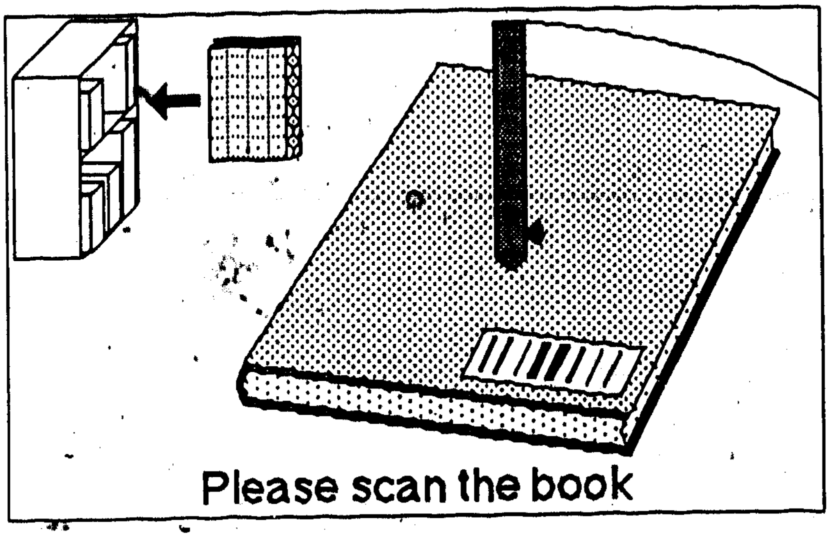


Figure 8.11 Input item ID icon in Return.

The small icon in the top left hand corner is used to indicate that the user is performing a RETURN rather than a LOAN. After sliding the bar code scanner across the bar code label on the item, four possible situations can occur:

1. A successful return and the success icon (Figure 8.5) is displayed.
2. The item is not found and the item is not found icon (Figure 8.8) is displayed; the user should ask the librarian for help.
3. A missing item is found and the missing item found icon (Figure 8.9) is displayed; the user should report the missing item to the librarian.
4. The item is not on loan and the success icon (Figure 8.5) is displayed.

In the first three cases, the purpose of using the same icons in both LOAN and RETURN is to reinforce the meaning of the icons. For example, in the case of a successful return, the success icon (Figure 8.5) is displayed to reinforce the idea that the icon means success. In a general interface it might be more important to emphasize the differences between the operations. But in the case of children's interfaces, **reinforcement** is more important.

Displaying the success icon when the item is not on loan leads to another principle in designing interfaces for children: **forgiveness.** An interface must be willing to forgive previous errors if they will not adversely affect the present situation. This error occurs when the user made an error using LOAN or forgot to use the LOAN system. This is a very common situation in elementary schools. Since the item is being returned, the error can be forgiven without affecting the system. Although this would not work very well in public libraries, it can be an acceptable solution in elementary school libraries. The librarian at LaPerle prefers this approach so the latest version of CHILDS behaves in this manner.

8.2.3 Search.

After the SEARCH function is initiated from the main menu (Figure 8.2), the search menu (Figure 8.12) is displayed.

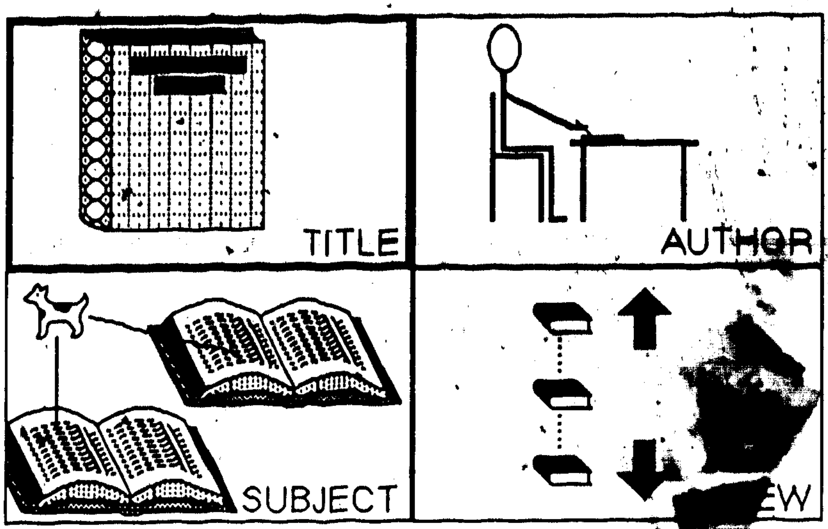


Figure 8.12 Search menu.

A user can search for items by title keywords, authors' surname, or subjects. To select a search mode, the user simply presses the return key when the desired mode is highlighted. The VIEW function allow the user to view the item list of a previous search if one was performed.

In the original design, a book with a question mark on top was used to represent search by title. Although it is quite logical to use a question mark to represent a search function, it is unclear and misleading in this case since the question mark is used in LOAN to represent an error. This leads to another principle in designing interfaces for children: **consistency**. So, the present system uses a book with a blank title (Figure 8.12) to represent search by title.

The procedures for all three (title, author, and subject) searches are quite similar. Therefore only the search by title is discussed here. For a better understanding of the search sub-system, please refer to the Tutorial Guide to Search in Appendix B.

After the TITLE function is selected from the search menu (Figure 8.12), the item kind menu (Figure 8.13) is displayed.

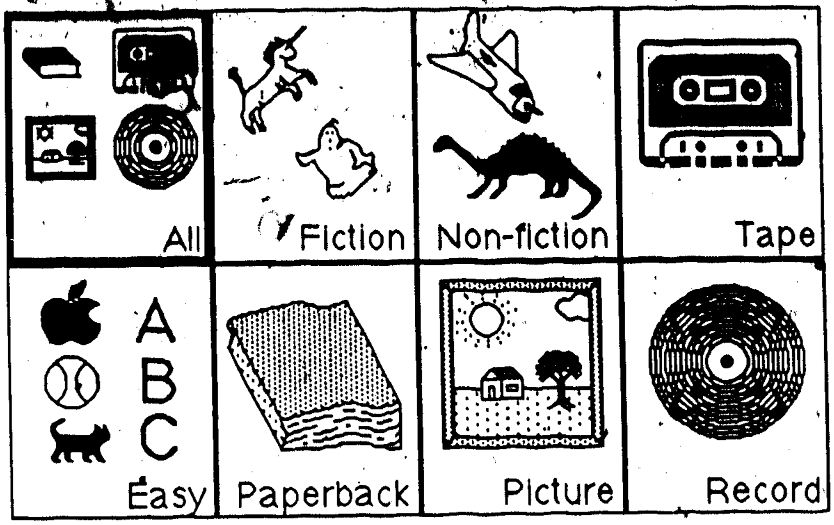


Figure 8.13 Item kind menu.

After the user has selected the desired item kind, the input title keywords for item search icon (Figure 8.14) is displayed.

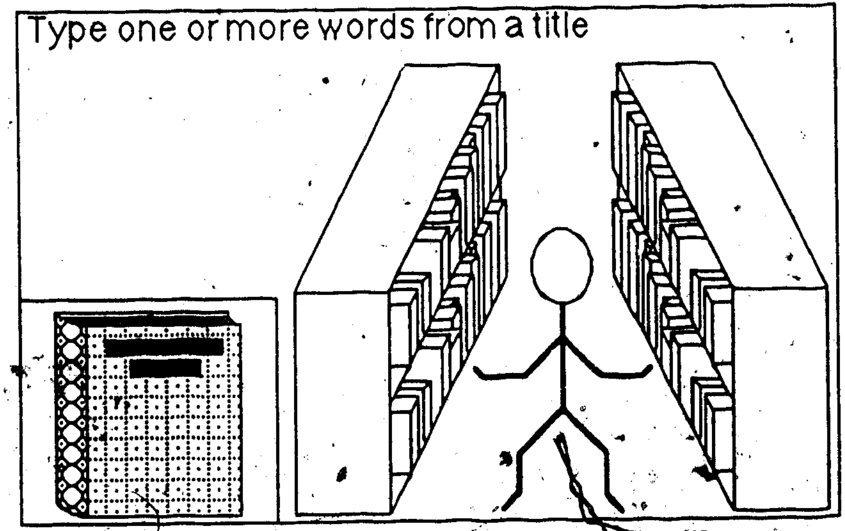


Figure 8.14 Input title keywords for item search icon.

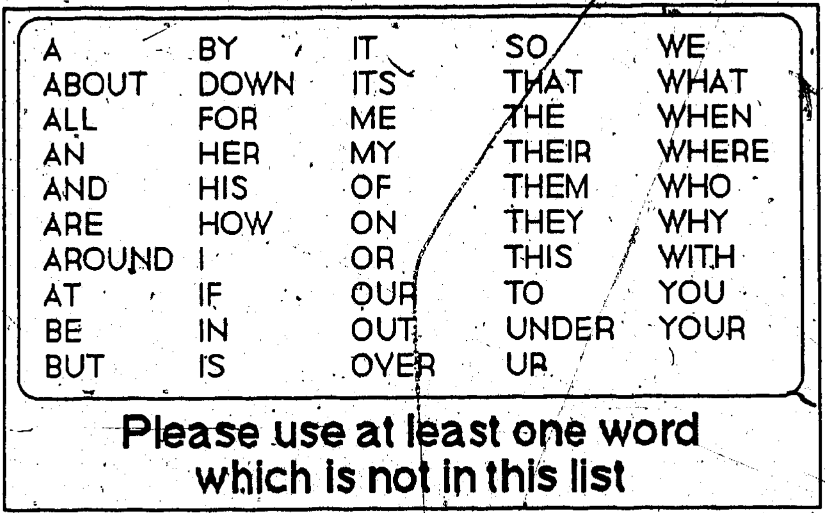
The user can either type in title keywords or an item ID. If title keywords are entered, the system will search on the keywords.

If an item ID is entered, the system will use the subjects of the item with that item ID to perform the search. This feature lowers the abstraction level (discussed earlier in Chapter 2) from recognition to conceptualization and therefore even kindergarten students can perform item search. The subjects of the item are used instead of the title because it is much simpler. If the title is used, the system must "intelligently" decide which word in the title is significant and should be searched on and which ones are not significant. Furthermore, children who use this feature are usually interested in items of the same subject. However, intelligent title search would be a good subject for further research.

In the case where an item ID is entered in a subject search, the subject of the item is used. In the case where an item ID is entered in an author search, the author(s) of that item are used. This is done because there are cases where children want to find other items by the same author(s).

During the time when the system is performing the search, the interface displays one small happy face for every item found. This non-numerical fashion of showing quantities (i.e. more happy faces mean more items found) is very suitable in children's interfaces because of their limited reading ability.

In the title search mode, if the user have chosen a keyword which is too common (discussed in section 6.3.6), the illegal keyword icon (Figure 8.15) is displayed with the keyword highlighted.



```
A        BY     IT     SO      WE
ABOUT    DOWN   ITS    THAT    WHAT
ALL      FOR    ME     THE     WHEN
AN       HER    MY     THEIR   WHERE
AND      HIS    OF     THEM    WHO
ARE      HOW    ON     THEY    WHY
AROUND   I      OR     THIS    WITH
AT       IF     OUR    TO      YOU
BE       IN     OUT    UNDER   YOUR
BUT      IS     OVER   UP
```

Please use at least one word which is not in this list

Figure 8.15 Illegal keyword icon.

This icon informs the user to use at least one keyword not given in the list displayed on the screen when using the title search mode.

8.2.4 Quit.

After the QUIT function is selected, the system will ask the user to enter a password. The system returns to the main menu (Figure 8.2) if the password is incorrect and quits if the password is correct. This feature is added for security since the librarian does not want the children to be able to get out of the loan/return and search sub-system and have access to the computer's operating system commands.

Chapter 9.

Conclusions.

CHILDS is an interactive display system for loan/return, searching and inventory control in an elementary school library. It was originally developed as a laboratory for studying computer interfaces for children.

The design of the second version of CHILDS was divided into two parts: the file system operations and the user interface. The file system was fairly stable since system capabilities were essentially fixed. The user interfaee, on the other hand, changed with user feedback.

The file system was constructed based on the data abstraction principle and the concept of an Encoded String. As a result, the file system of the second version is more efficient and reliable as compared to the first version.

From children using both version of CHILDS, six fundamental principles for children-computer interface design are postulated: clarity, consistency, literalism, forgiveness, forcefulness, and reinforcement. However, experiments have to be conducted to verify these principles. In addition, we have shown how the abstraction level required to perform the three library operations (loan, return, and search) can be reduced to a level at which kindergarten children can perform.

Although CHILDS was originally developed to study children-computer interface, there has not been any formal experiments conducted on the students at LaPerle. Instead, informal observations and feedback were obtained from the school librarian.

As far as future improvements of CHILDS is concerned, there is currently an effort to redesign the interface to run on a Macintosh^{TM} to take advantage of the high revolution graphics, the mouse, and speech synthesis.

Finally, it is this author's hope that the second version of CHILDS could be used as a tool to study children-computer interface and icon design for childen.

References.

[Chan85] Chan, H. C., "Children's Interactive Library Display System": project report, University of Alberta,1985

[Fair85] Fairley, R. E., *Software Engineering Concepts*, McGraw-Hill, Inc., New York, 1985

[Ford85] Ford, G. and Wiener R., *Modula-2, A Software Development Approach*, John Wiley & Sons, Inc., 1985.

[Gold81] Goldberg, A. and Ross, J., "Is SmallTalk-80 System for Children?", BYTE, 6(8), pp. 348-368, 1981.

[Pape80] Papert, S.,*Children, Computers, and Powerful Ideas*, Basic Books, New York, 1980.

[Szaf87] Szafron, D. and Chan, H. C., "CHILDS: An Interactive Graphical Display System For Elementary School Libraries.", to appear in CIPS/ACI Congress 87, University of Alberta, 1987.

[Szaf82] Szafron, M., "Library Internship Report", Portland State University, 1982.

Appendix A.

CHILDS: Tutorial Guide to Loan and Return.

CHILDS Version 2.0

Manual Revision 1.0 (September 1987) by:

Duane Szafron, Bill Kaluzniak, Raymond Kim Ho and Brian Wilkerson

Programming Languages Group

Department of Computing Science

University of Alberta

The circulation system supports three functions: loan, return and search. This tutorial describes loan and return. A separate tutorial, Tutorial Guide To Search, describes the search capabilities. This tutorial describes the six situations which may arise when borrowing items and the one situation which arises when returning items, as well as how to exit the circulation system. The guide is divided into twelve sections:

1. Starting Loan
2. Stopping Loan
3. Successfully Borrowing an Item
4. Trying to Borrow an Item Which is on Loan
5. Trying to Borrow an Invalid Item
6. Trying to Use an Invalid User Card
7. Trying to Borrow too Many Items
8. Trying to Borrow a Missing Item
9. Starting Return
10. Stopping Return
11. Sucessfully Returning an Item
12. Exiting the Circulation System

When the circulation system is started the circulation menu of figure 1 is displayed.

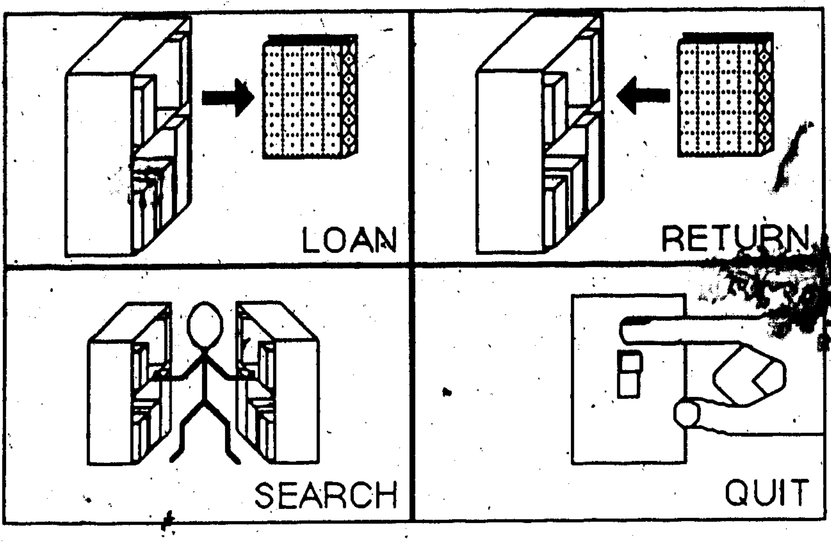
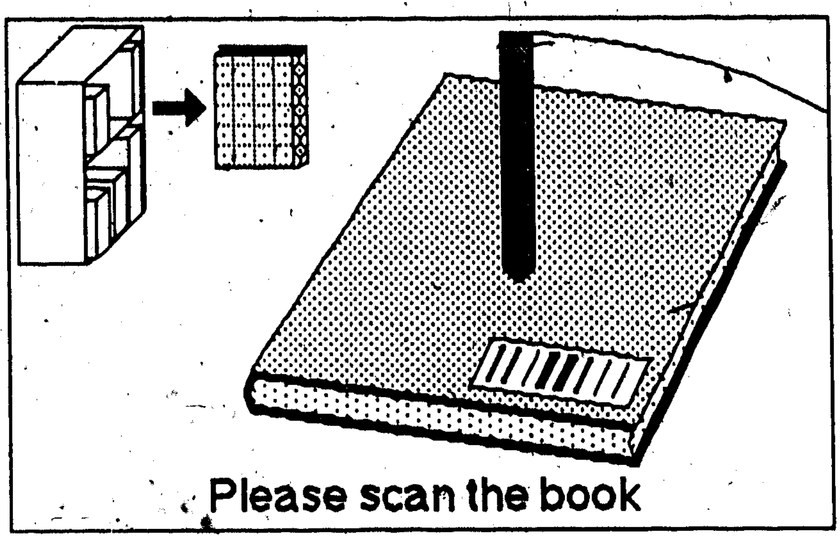


Figure 1. The circulation menu

1. Starting Loan

STEP 1: Press the ☝ key until the circulation menu of figure 1 is displayed.

STEP 2: By pressing the four arrow keys 8↑ 2↓ 6→ 4← it is possible to choose one of the four panes in the circulation menu. As the arrow keys are pressed, the highlighting square will move from pane to pane. When the ↵ key is pressed, the action pictured in the highlighted pane will begin. Use the arrow keys to move the highlighting square to the loan pane and press the ↵ key. The cue of figure 2 will be displayed. If you do not see the cue of figure 2, start over from step 1, and this time, make sure that the loan pane is highlighted before pressing the ↵ key.



Please scan the book

Figure 2. The item scan cue for loan

Notice the book coming out of the shelf in the top left-hand corner of this cue. It illustrates that you are about to borrow an item. This icon differentiates this cue from a similar cue which is displayed during item return.

2. Stopping Loan

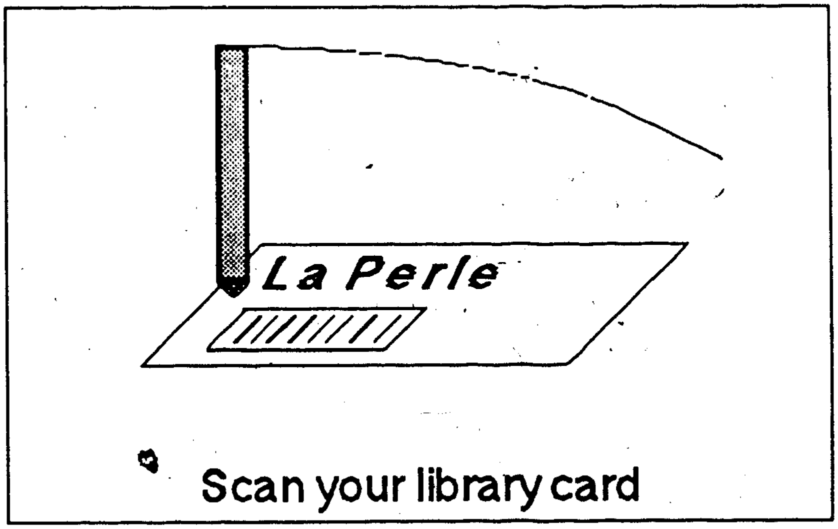
STEP 1: To exit loan and go back to the circulation menu of figure 1 press the

☜ key until the circulation menu of figure 1 is displayed.

3. Successfully Borrowing an Item

STEP 1: Ensure that the cue of figure 2 is displayed. If it is not, then follow the procedure described in section 1: Starting Loan.

STEP 2: The cue of figure 2 requests you to scan the bar code on the item to be borrowed. Alternatively you can type in the number on the bar code and then press the ↵ key. In fact, you can always type a bar code number and press the ↵ key instead of scanning the barcode. After scanning the item's bar code, the cue shown in figure 3 will be displayed. If a different picture appears instead, then press any key and start over at step 2 using a different item (a valid item that is not currently on loan or missing).



Scan your library card

Figure 3. The user card scan cue

STEP 3: The cue shown in figure 3 requests you to scan the bar code on your user card. After scanning it, the success icon shown in figure 4 will briefly flash on the screen to indicate a successful loan and then the loan item scan cue of figure 2.will be displayed so that another item can be borrowed. If a different picture appears instead of the success icon, press any key and begin again at step 1 using a different user card (a valid user card whose owner has not borrowed the maximum allowable number of items).

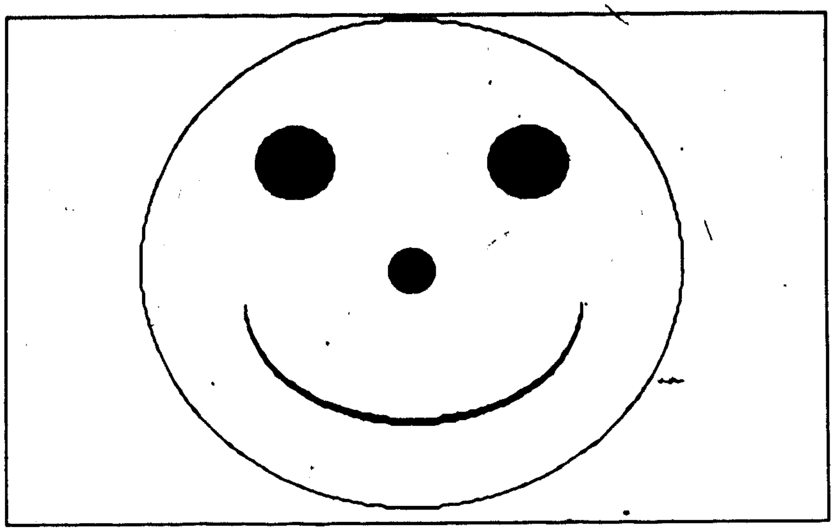


Figure 4. The success icon

STEP 4: If you are finished borrowing items, follow the procedure described in section 2, Stopping Loan. Otherwise you may repeat steps 2 and 3 to borrow more items.

4. Trying to Borrow an Item Which is on Loan

STEP 1: Follow steps 1, 2 and 3 of section 3 to successfully borrow an item. The item scan cue of figure 2 will then be displayed.

STEP 2: Now scan the bar code on the same item again. The error icon of figure 5 will be displayed except that the number in the head of the figure, and the number in the book will be different. This icon indicates that the item with the item number shown has been borrowed by the user with the user number shown and that you cannot borrow it. Press any key to acknowledge the error and the item scan cue of figure 2 will be displayed.

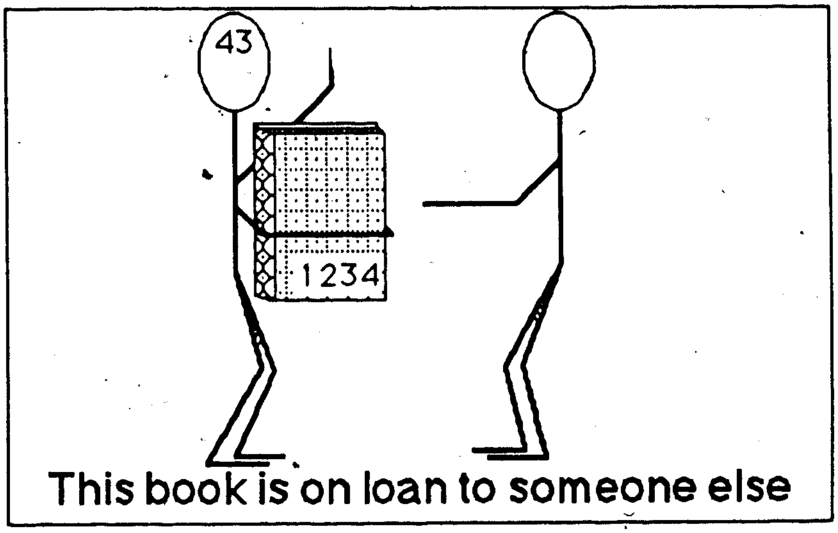
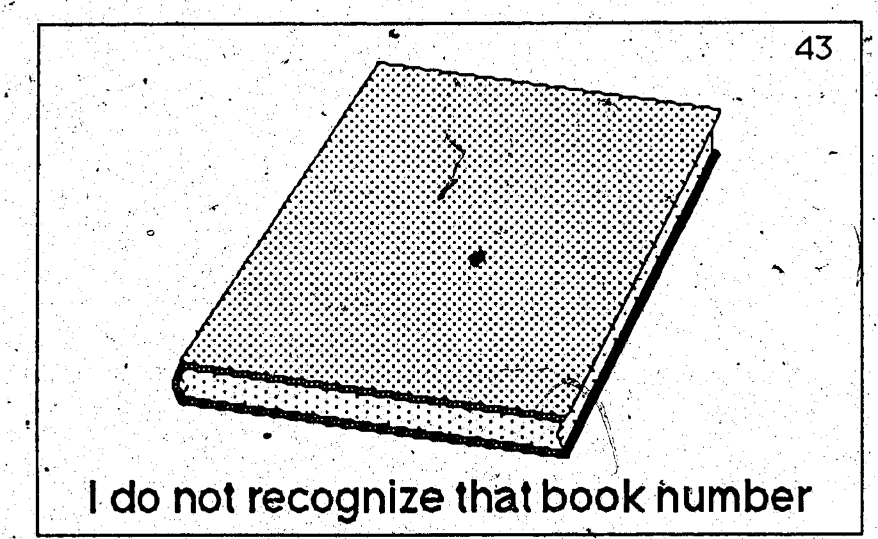


Figure 5: The already on loan error icon

5. Trying to Borrow an Invalid Item

STEP 1: Ensure that the cue of figure 2 is displayed. If it is not, then follow the procedure described in section 1: Starting Loan.

STEP 2: Now scan the bar code on a new item which has not yet been entered into CHILDS or scan a user card instead of an item. The error icon of figure 6 will be displayed. The bar code number of the item or user card which you scanned will appear in the top right hand corner. This icon indicates that CHILDS does not have an item with the number which was entered. Press any key to acknowledge the error and the item scan cue of figure 2 will be re-displayed.



43

I do not recognize that book number

Figure 6. The unknown item error icon

6. Trying to Use an Invalid User Card

STEP 1: Follow steps 1 and 2 of section 3 to start borrowing an item. The user card scan cue of figure 3 will then be displayed.

STEP 2: Now scan the bar code on a new user card which has not yet been entered into CHILDS or scan an item instead of a user card. The error icon of figure 7 will be displayed. The bar code number of the user card or item which you scanned will appear in the top right hand corner. This icon indicates that CHILDS does not have a user with the number which was entered. Press any key to acknowledge the error and the item scan cue of figure 2 will be re-displayed.

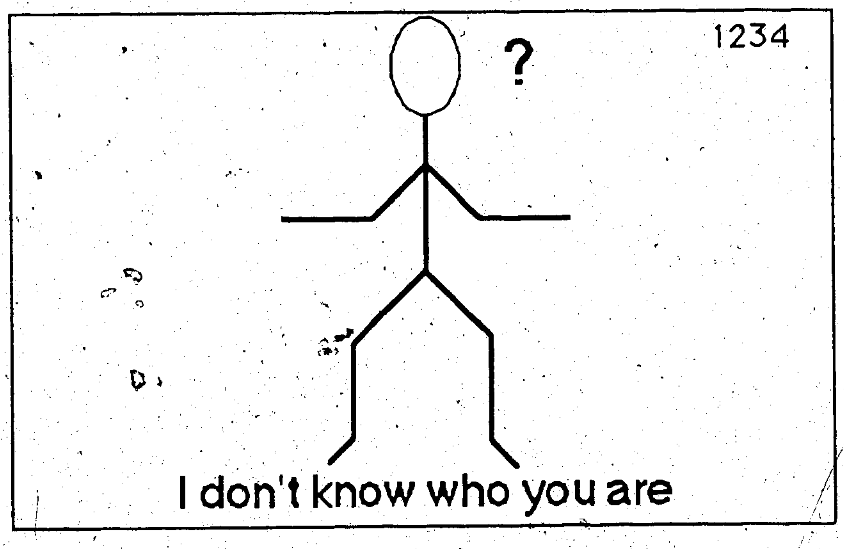


Figure 7. The unknown user icon

7. Trying to Borrow too Many Item

STEP 1: Repeatedly follow steps 1, 2 and 3 of section 3 to successfully borrow items until the error icon of figure 8 appears. It will have the ID number of the item you scanned visable in the top right hand corner. This icon indicates that the item cannot be borrowed because you have already borrowed your limit of items. The ID numbers of the items you have already borrowed will be displayed under the ID number of the item which you tried to borrow. Press any key to acknowledge the error and the item scan cue of figure 2 will be displayed.

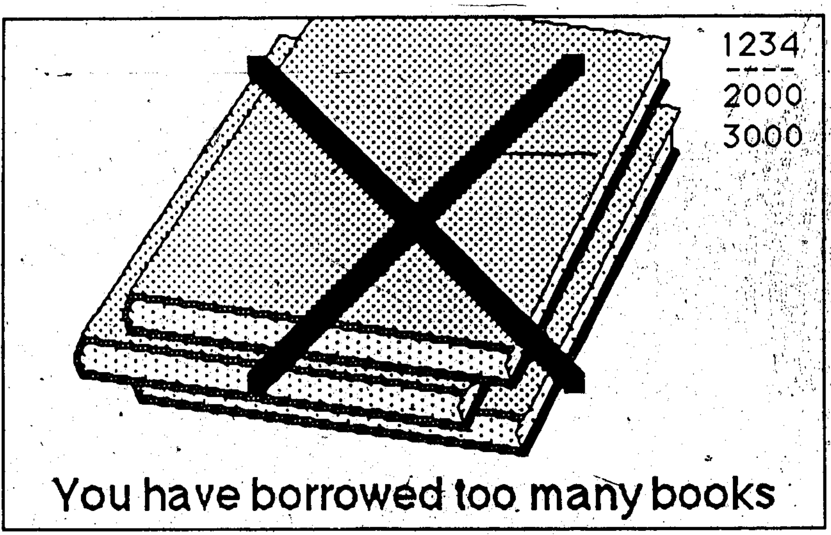


Figure 8. The over limit error icon

8. Trying to Borrow a Missing Item

This situation will only arise if you try to borrow an item which has been marked missing. You will not be able to try this part of the tutorial unless you have such a missing item. Items can be marked missing by using the Item Edit system described in the Item Editor Tutorial.

STEP 1: Ensure that the cue of figure 2 is displayed. If it is not then follow the procedure described in section 1: Starting Loan.

STEP 2: Now scan the bar code on a missing item. The error icon of figure 9 will be displayed with the ID number of the scanned item in the top right hand corner. This icon indicates that the item should be taken to the librarian and cannot be borrowed at this time. Press any key to acknowledge the error and the item scan cue of figure 2 will be displayed.

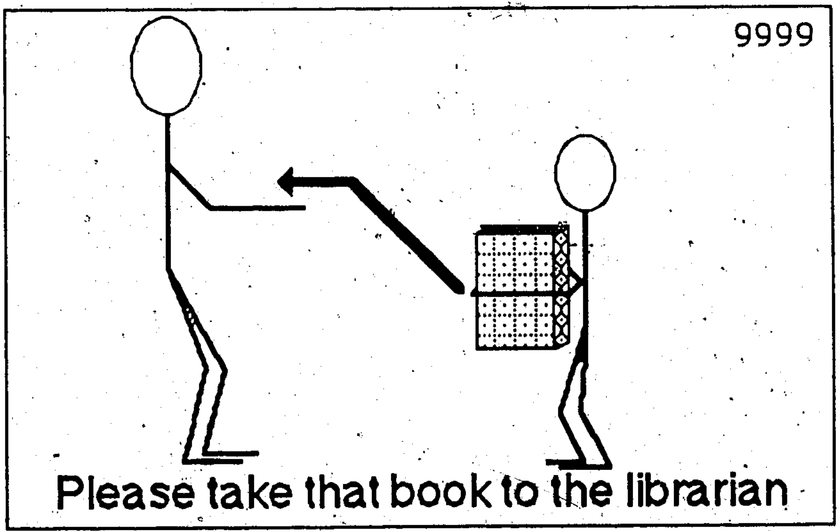


Figure 9. The missing item error icon

9. Starting Return

STEP 1: Press the ☞ key until the circulation menu of figure 1 is displayed.

STEP 2: By pressing the four arrow keys 8↑ 2↓ 6→ 4← it is possible to choose one of the four panes in the circulation menu. As the arrow keys are pressed, the highlighting square will move from pane to pane. When the ↵ key is pressed, the action pictured in the highlighted pane will begin. Use the arrow keys to move the highlighting square to the return pane and press the ↵ key. The cue of figure 10 will be displayed. If you do not see the cue of figure 10, start over from step 1, and this time, make sure that the return pane is highlighted before pressing the ↵ key.

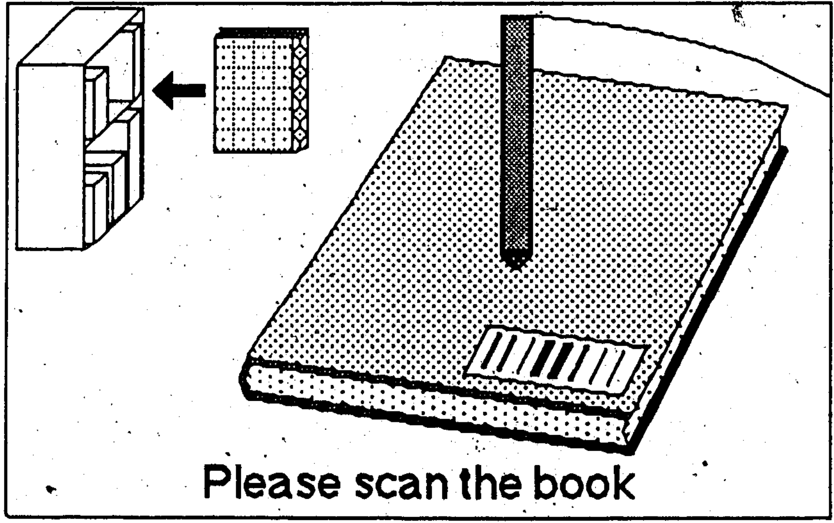


Figure 10. The item scan cue for return

Notice the book going into the shelf in the top left hand corner of this cue. It illustrates that you are about to return an item. This icon differentiates this cue from a similar cue which is displayed during item loan.

10. Stopping Return

STEP 1: To exit return and go back to the circulation menu of figure 1 press the 👍 key until the circulation menu of figure 1 is displayed.

11. Successfully Returning an Item

STEP 1: Ensure that the cue of figure 2 is displayed. If it is not then follow the procedure described in section 9: Starting Return.

STEP 2: The cue of figure 10 requests you to scan the bar code on the item to be returned. Alternatively you can type in the number on the bar code and then press the ↵ key. In fact, you can always type a bar code number and press the ↵ key instead of scanning the barcode. After scanning the item's bar code, the success icon shown in figure 4 will be displayed. If a different picture appears instead, then press any key and start over at step 2 using a different item (a valid item that is not missing).

STEP 3: If you are finished returning items, follow the procedure described in section 10 to leave return. Otherwise you may repeat step 2 to return more items.

12. Exiting the Circulation System

STEP 1: Press the ☞ key until the circulation menu of figure 1 is displayed.

STEP 2: By pressing the four arrow keys 8↑ 2↓ 6→ 4← it is possible to choose one of the four panes in the circulation menu. As the arrow keys are pressed, the highlighting square will move from pane to pane. When the ↵ key is pressed, the action pictured in the highlighted pane will begin. Use the arrow keys to move the highlighting square to the quit pane and press the ↵ key. The cue of figure 11 will be displayed. If you do not see the cue of figure 11, start over from step 1, and this time, make sure that the quit pane is highlighted before pressing the ↵ key.

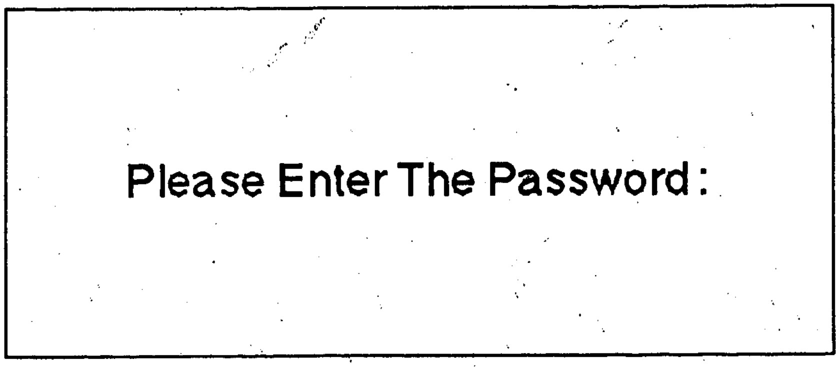

Please Enter The Password:

Figure 11. The password cue

STEP 3: Enter the password by typing it on the keyboard and pressing the ↵ key. An incorrect password will cause the system to return to the circulation menu of figure 1. If the password is correct, the circulation system will be exited.

Appendix B.

CHILDS: Tutorial Guide to Search.

CHILDS Version 2.0

Manual Revision 1.0 (September 1987) by:

Duane Szafron, Bill Kaluzniak, Raymond Kim Ho and Brian Wilkerson

Programming Languages Group

Department of Computing Science

University of Albertá

The circulation system supports three functions: loan, return, and search. This tutorial describes search. A separate tutorial, Tutorial Guide to Loan and Return, describes the loan and return capabilities. Since library collections are different, the search results described in this tutorial will not exactly match your search results. However, your results will be similar. This tutorial is divided into twelve sections:

1. Starting Search
2. Stopping Search
3. Searching for Items By Title
4. Viewing the Previous Search
5. Searching for Items By Author
6. Searching for Items By Subject
7. Limiting the Search by Category
8. Using More than One Title (Subject or Author) Keyword
9. Searching for Items by a Favorite Author (or on a Favorite Subject)
10. An Unsuccessful Subject (Author or Title) Search
11. An Unsuccessful Title Search due to Common Words
12. Exiting the Circulation System

When the circulation system is started the circulation menu of figure 1 is displayed.

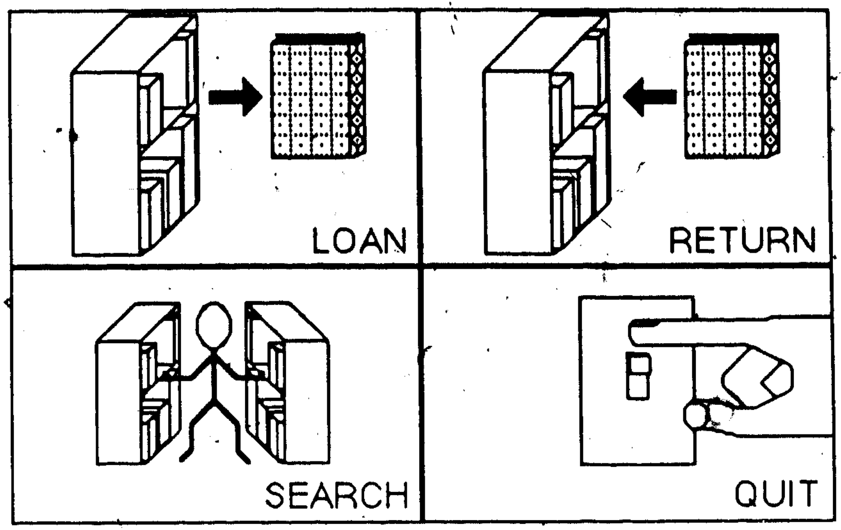


Figure 1. The circulation menu

1. Starting Search

STEP 1: Press the ☞ key until the circulation menu of figure 1 is displayed.

STEP 2: By pressing combinations of the four arrow keys 8↑ 2↓ 6→ 4← it is possible to choose one of the four panes in the circulation menu. As the arrow keys are pressed, the highlighting square will move from pane to pane. When the ↵ key is pressed, the action pictured in the highlighted pane will begin. Use the arrow keys to move the highlighting square to the search pane and press the ↵ key. The search menu of figure 2 will be displayed. If you do not see the search menu of figure 2, start over from step 1, and this time, make sure that the search pane is highlighted before pressing the ↵ key.

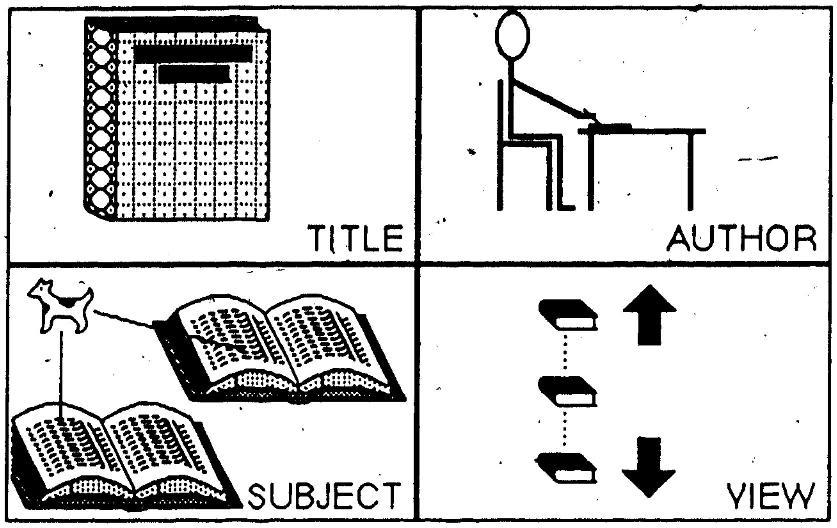


Figure 2. The search menu

The search menu has four panes:

1. Search by title.
2. Search by author.
3. Search by subject.
4. View the previous search results.

2. Stopping Search

STEP 1: To exit search and go back to the circulation menu of figure 1 press the 👍 key until the circulation menu of figure 1 is displayed.

3. Searching for Items By Title

STEP 1: Ensure that the search menu shown in figure 2 is displayed. If it is not, then follow the procedure described in section 1: Starting Search.

STEP 2: By pressing combinations of the four arrow keys 8↑ 2↓ 6→ 4← it is possible to choose one of the four panes in the search menu. As the arrow keys are pressed, the highlighting square will move from pane to pane. When the ↵ key is pressed, the action pictured in the highlighted pane will begin. Use the arrow keys to move the highlighting square to the title pane and press the ↵ key. The cue of figure 3 will be displayed.

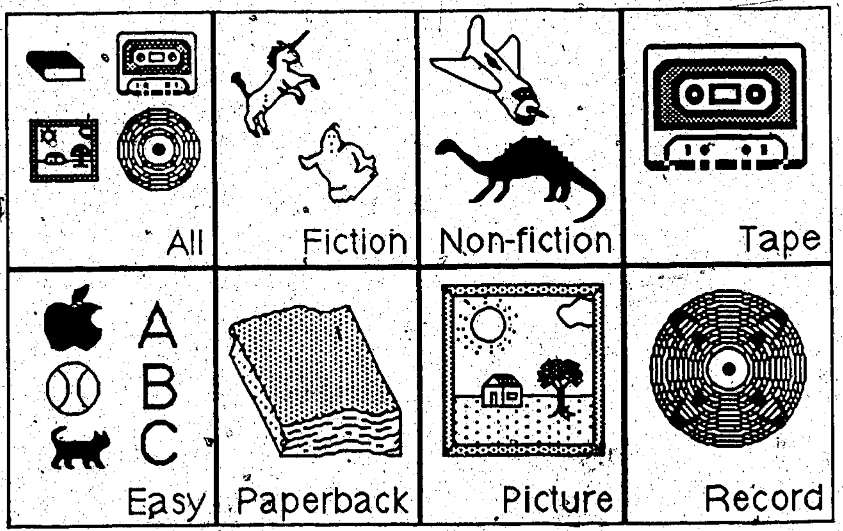
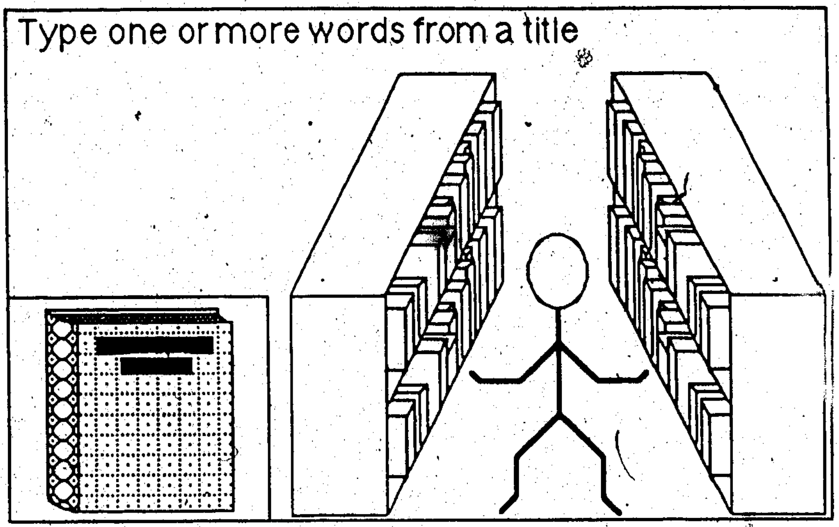


Figure 3. The category menu

If you do not see the category menu of figure 3, start again from step 1 and this time, make sure that the title pane is highlighted before pressing the ↵ key. The category menu contains seven panes representing seven categories of items as well as a pane representing all categories of items. These categories are used to restrict the results of a search.

STEP 3: Use the arrow keys to highlight the **All** pane and then press the ↵ key. The title cue shown in figure 4 will be displayed.



Type one or more words from a title

Figure 4. Title cue

Notice the title-search icon in the lower left hand corner. If it is not present then press the ☞ key to go back to the search menu and start again from step 1.

STEP 4: Use the keyboard to type a word from the title of an item. For example, type the five characters in the word **horse**. As you press a key on the keyboard, the character will be displayed on the screen. If you make a typing mistake you can erase it by pressing the ← key. After typing the word press the ↵ key; several happy face icons will be displayed on the screen as shown in figure 5.

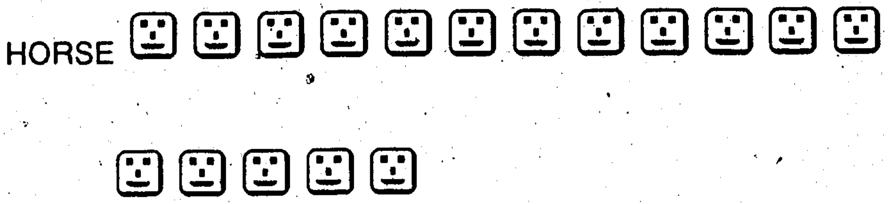
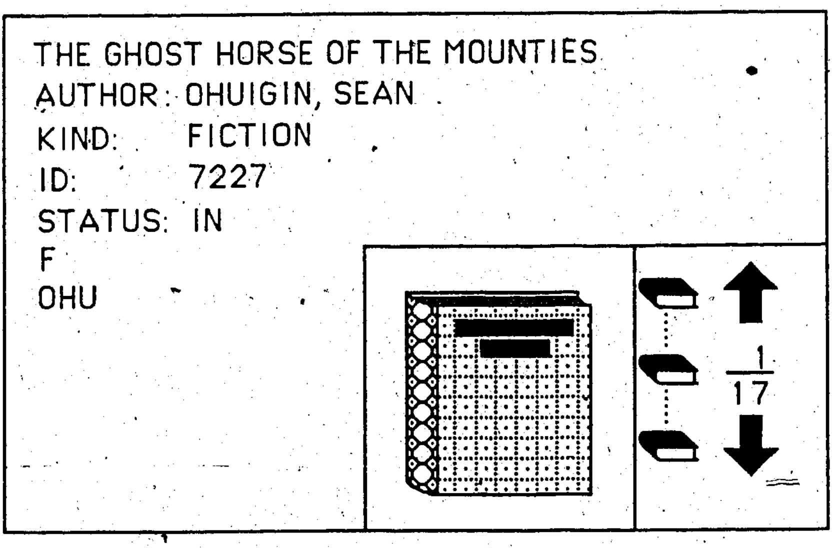


Figure 5. Searching screen

One happy face will be displayed for each item which contains the word **horse** in its title. After all the items have been found, the first one will be displayed as shown in figure 6. Since your library contains different items, a different item will probably be displayed.



THE GHOST HORSE OF THE MOUNTIES
AUTHOR: OHUIGIN, SEAN
KIND: FICTION
ID: 7227
STATUS: IN
F
OHU

Figure 6. Item information display

If no happy faces are displayed and a display unlike the one shown in figure 6 appears, then press the ☞ key so that the search menu of figure 2 is displayed and begin again at step one. This time, choose a different word like **dog** or **cat** instead of **horse**. The first line shows the title. Notice that the word horse is in the title. The second line gives the author's name. If there is more than one author, then all of the authors are listed. Next come the subjects that this item concerns. The next line shows that this item is categorized as Fiction. Fiction is an item category. Since we specified **All** in the category menu of figure 3, items from all categories may be included in the search result. The number 7227 is the number that is on the bar code of this item. The status line can be one of three words:

IN : This item may be borrowed

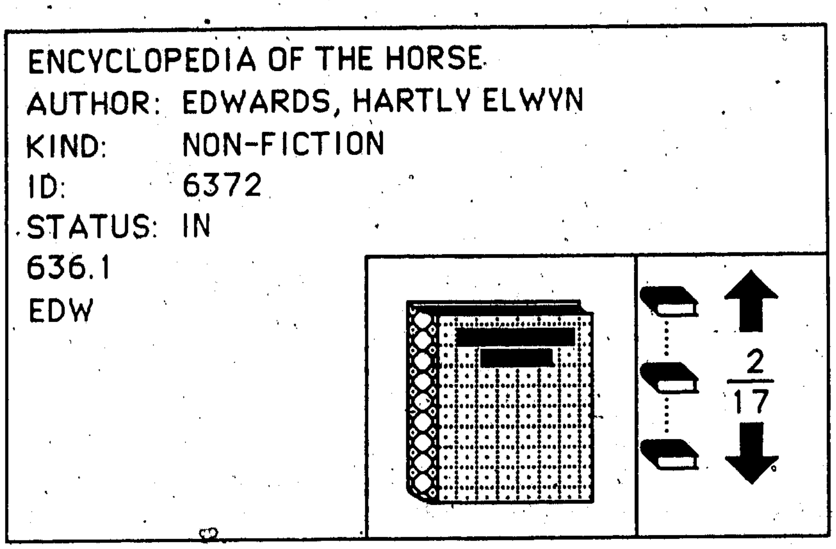
OUT : This item is on loan and may not be borrowed

MISSING : This item is lost

The next two lines contain the Dewey number of the item. The Dewey number can be used to locate the item in the library.

The picture in the lower right hand corner is called the view icon. Inside the view icon there are two numbers separated by a bar. The lower number represents the number of items found in the search. The upper number represents the item being displayed. In this case we are viewing the first item of seventeen items in the search list.

STEP 5: To view the next item in the search list press the 2↓ key on the keyboard. The second item in the search list will be displayed as shown in figure 7. Again, since your library collection is different, you will probably see a display for a different item.



ENCYCLOPEDIA OF THE HORSE
AUTHOR: EDWARDS, HARTLY ELWYN
KIND: NON-FICTION
ID: 6372
STATUS: IN
636.1
EDW

Figure 7. Item information display

This item is also in the category non-fiction. The view icon now indicates that this is the second item of seventeen. The 2↓ key and 8↑ key can be used to view all of the items in the search list.

STEP 6: When you are done viewing the items in this search list, press the ☜ key to go back to the search menu shown in figure 2.

4. Viewing the Previous Search

STEP 1: Perform a search by following steps 1 to 5 in section 3.

STEP 2: Ensure that the search menu shown in figure 2 is displayed. If it is not, then follow the procedure described in section 1: Starting Search.

STEP 3: By pressing combinations of the four arrow keys [8↑] [2↓] [6→] [4←] it is possible to choose one of the four panes in the search menu. As the arrow keys are pressed, the highlighting square will move from pane to pane. When the [↵] key is pressed, the action pictured in the highlighted pane will begin. Use the arrow keys to move the highlighting square to the view icon and press the [↵] key. The item information shown in figure 6 will again be displayed. If you wish, you may again use the [8↑] [2↓] keys to view the information on other items in the search list. Whenever the view pane of the search menu is selected, the results of the previous search will be displayed. If no previous search has been performed, then nothing will be displayed.

STEP 4: When you are done viewing the items in this search list, press the [☞] key to go back to the search menu shown in figure 2.

5. Searching for Items By Author

STEP 1: Ensure that the search menu shown in figure 2 is displayed. If it is not, then follow the procedure described in section 1: Starting Search.

STEP 2: By pressing combinations of the four arrow keys 8↑ 2↓ 6→ 4← it is possible to choose one of the four panes in the search menu. As the arrow keys are pressed, the highlighting square will move from pane to pane. When the ↵ key is pressed, the action pictured in the highlighted pane will begin. Use the arrow keys to move the highlighting square to the author pane and press the ↵ key. If you do not see the category menu of figure 3, start again from step 1 and this time, make sure that the author pane is highlighted before pressing the ↵ key.

STEP 3: Use the arrow keys to highlight the **All** pane and then press the ↵ key. The author cue shown in figure 8 will be displayed.

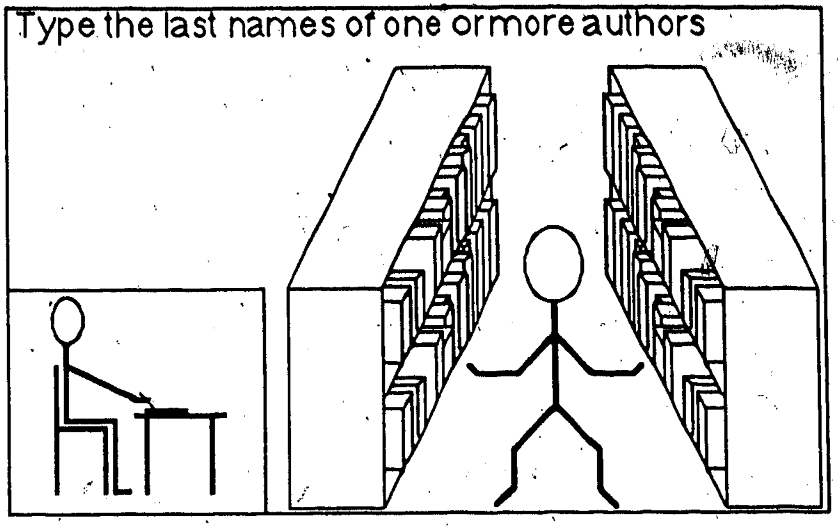

Type the last names of one or more authors

Figure 8. Author cue

Notice the author-search icon in the lower left hand corner. If it is not present then press the ☞ key to go back to the search menu and start again from step 1.

STEP 4: Use the keyboard to type one or more author names of an item. For example, type the six characters in the name **Sendak**. As you press a key on the keyboard, the character will be displayed on the screen. If you make a typing mistake you can erase it by pressing the ← key. After typing the name, press the ↵ key. Several happy face icons as shown in figure 9 will be displayed on the screen.

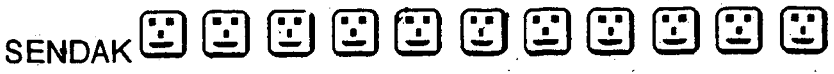
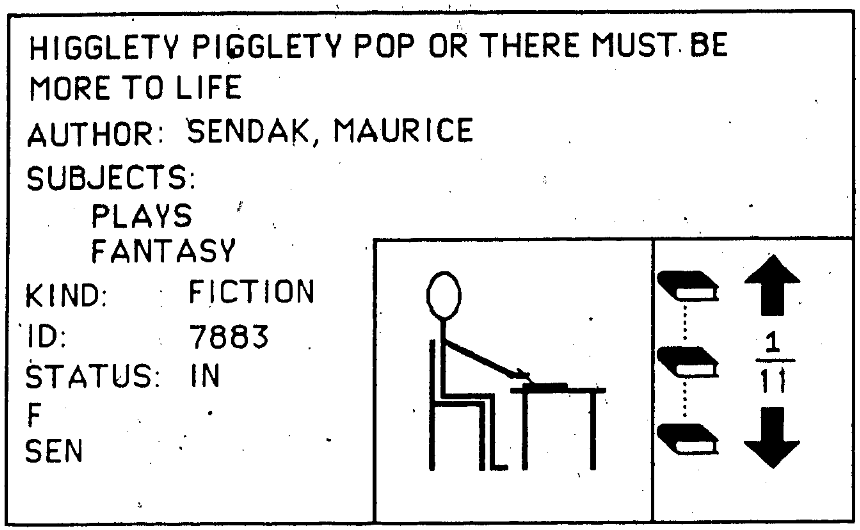

SENDAK

Figure 9. Searching screen

One happy face will be displayed for each item which contains the name **SENDAK** in its author list. After all the items have been found, the first one will be displayed as shown in figure 10. Since your library contains different items, a different item will probably be displayed.



HIGGLETY PIGGLETY POP OR THERE MUST BE
MORE TO LIFE
AUTHOR: SENDAK, MAURICE
SUBJECTS:
PLAYS
FANTASY
KIND: FICTION
ID: 7883
STATUS: IN
F
SEN

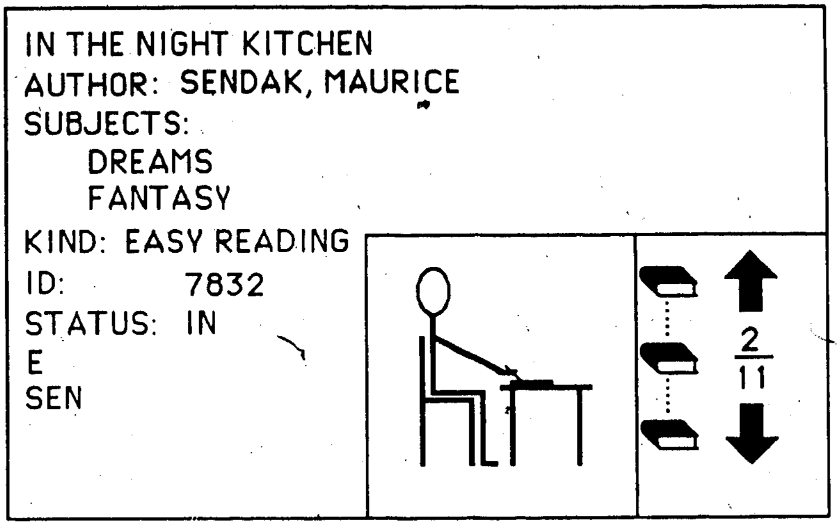
Figure 10. Item information display

The first line shows the title. The second line gives the author's names. As expected SENDAK is in the author list. The next line shows that this item is categorized as Fiction. Fiction is an item category. Since we specified **All** in the category menu of figure 3, items from all categories may be included in the search result. The ID number 7883 is the number that is on the bar code of this item. The status line can be one of three words:

IN : This item may be borrowed
OUT : This item is on loan and may not be borrowed
MISSING : This item is lost

The next two lines contain the Dewey number of the item. The Dewey number can be used to locate the item in the library. The picture in the lower right hand corner is called the view icon. Inside the view icon there are two numbers separated by a bar. The lower number represents the number of items found in the search. The upper number represents the item being displayed. In this case we are viewing the first item of eleven in the search list.

STEP 5: To view the next item in the search list press the 2↓ key on the keyboard. The second item in the search list will be displayed as shown in figure 11. Since your library contains different items, a different item will probably be displayed.



```
IN THE NIGHT KITCHEN
AUTHOR: SENDAK, MAURICE
SUBJECTS:
    DREAMS
    FANTASY
KIND: EASY READING
ID:        7832
STATUS:  IN
E
SEN
```

Figure 11. Item information display

This item is in the category Easy Reading. Notice that it has Sendak listed under the author heading. The view icon now indicates that this is the second item of eleven. The 2↓ key and 8↑ key can be used to view all of the items in the search list.

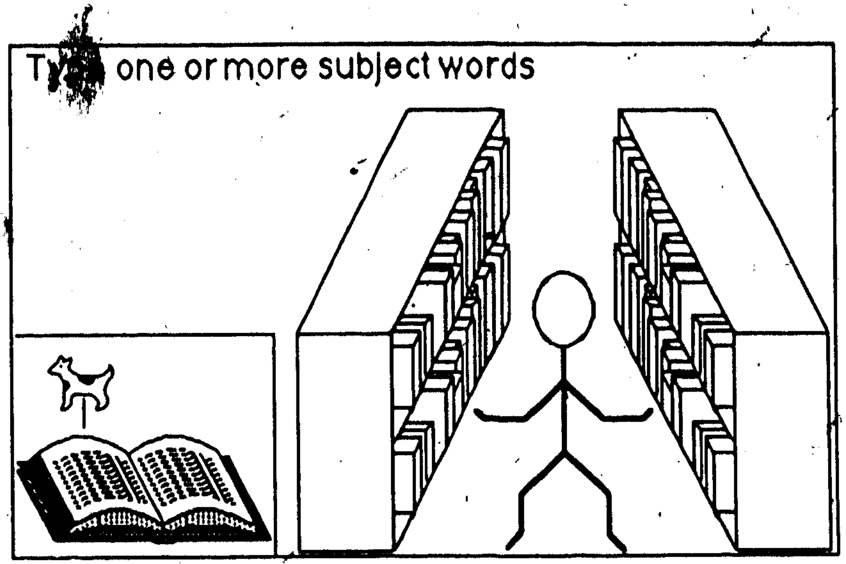
STEP 6: When you are done viewing the items in this search list, press the ☞ key to go back to the search menu shown in figure 2.

6. Searching For Items By Subject

STEP 1: Ensure that the search menu shown in figure 2 is displayed. If it is not, then follow the procedure described in section 1: Starting Search.

STEP 2: By pressing combinations of the four arrow keys [8↑] [2↓] [6→] [4←] it is possible to choose one of the four panes in the search menu. As the arrow keys are pressed, the highlighting square will move from pane to pane. When the [↵] key is pressed, the action pictured in the highlighted pane will begin. Use the arrow keys to move the highlighting square to the subject pane and press the [↵] key. If you do not see the category menu of figure 3, start again from step 1 and this time, make sure that the subject pane is highlighted before pressing the [↵] key. The category menu contains seven panes representing seven categories of items as well as a pane representing all categories of items. These categories are used to restrict the results of a search.

STEP 3: Use the arrow keys [8↑] [2↓] [6→] [4←] to highlight the All pane and then press the [↵] key. The subject cue shown in figure 12 will be displayed.

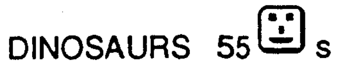


Type one or more subject words

Figure 12. Subject cue

Notice the subject-search icon in the lower left hand corner. If it is not present then press the ☞ key to go back to the search menu and start again from step 1.

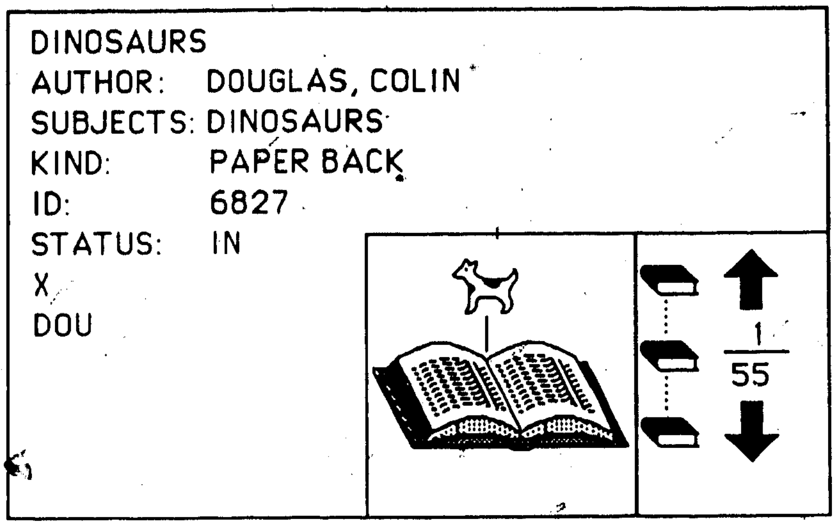
STEP 4: Use the keyboard to type a word from the title of an item. For example, type the nine characters in the word **dinosaurs**. As you press a key on the keyboard, the character will be displayed on the screen. If you make a typing mistake you can erase it by pressing the ← key. After typing the word press the ↵ key; several happy face icons will be displayed on the screen as shown in figure 13.



DINOSAURS 55☺s

Figure 13. Searching screen

One happy face will be displayed for each item which contains the word **dinosaurs** in its subject-list. After all the items have been found, the first one will be displayed as shown in figure 14. Since your library contains different items, a different item will probably be displayed.



DINOSAURS
AUTHOR: DOUGLAS, COLIN
SUBJECTS: DINOSAURS
KIND: PAPER BACK
ID: 6827
STATUS: IN
X
DOU

Figure 14. Item information display

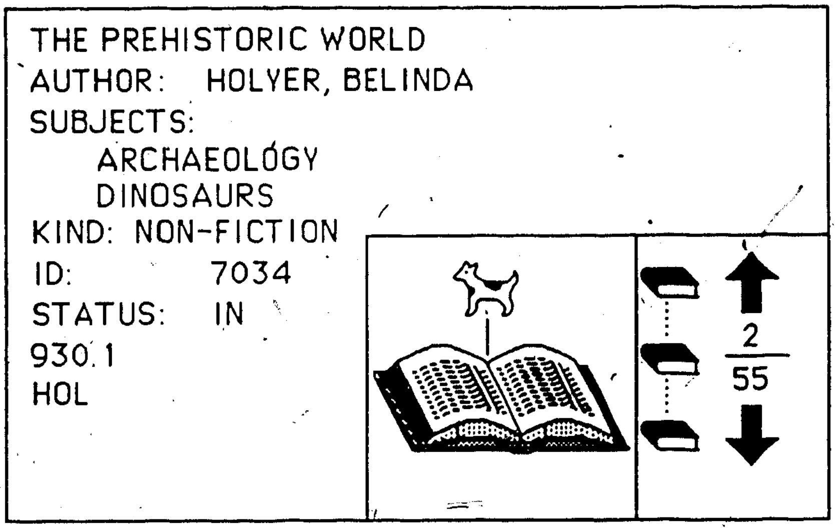
If no happy faces are displayed and a display unlike the one shown in figure 13 appears, then press the ☞ key so that the search menu of figure 2 is displayed and begin again at step one. This time, choose a different word like **dog** or **cat** instead of **dinosaurs**. The first line shows the title. The second line gives the author's name. If there is more than one author, then all of the authors are listed. Next come the subjects that this item concerns. Notice that the word **dinosaurs** is in the subject list. The next line shows that this item is categorized as Paper Back. Paper Back is an item category. Since we specified **All** in the category menu of figure 3, items from all categories may be included in the search result. The ID number 6827 is the number that is on the bar code of this item . The status line can be one of three words:

IN : This item may be borrowed
OUT : This item is on loan and may not be borrowed
MISSING : This item is lost

The next two lines contain the Dewey number of the item. The Dewey number can be used to locate the item in the library.

The picture in the lower right hand corner is called the view icon. Inside the view icon there are two numbers separated by a bar. The lower number represents the number of items found in the search. The upper number represents the item being displayed. In this case we are viewing the first item of fifty-five items in the search list.

STEP 5: To view the next item in the search list press the 2↓ key on the keyboard. The second item in the search list will be displayed as shown in figure 15. Again, since your library collection is different, you will probably see a display for a different item.



THE PREHISTORIC WORLD
AUTHOR: HOLYER, BELINDA
SUBJECTS:
ARCHAEOLOGY
DINOSAURS
KIND: NON-FICTION
ID: 7034
STATUS: IN
930.1
HOL

Figure 15. Item information display

This item is in the category Non-fiction. Notice that **dinosaurs** is listed under the heading "subject". The view icon now indicates that this is the second item of fifty-five items. The 2↓ key and 8↑ key can be used to view all of the items in the search list.

STEP 6: When you are done viewing the items in this search list, press the

👍 key to go back to the search menu shown in figure 2.

7. Limiting the Search by Category

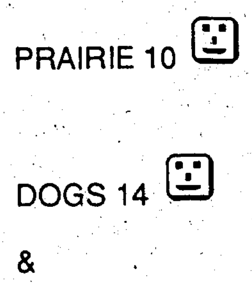
STEP 1: Repeat steps 1 and 2 of section 3, "Searching for Items by Title". You should see the category menu of figure 3 displayed.

STEP 2: Use the arrow keys 8↑ 2↓ 6→ 4← to choose the **fiction** category and then press the return key ↵. You will now see the title cue of figure 4, just as you did in section 3. Follow steps 4 and 5 of section 3. This time, only items whose KIND is fiction will be displayed. Note that you can limit the search to one category when you search by author, by title, or by subject; and that you can choose any of the seven categories.

8. Using More than One Title (Subject or Author) Keyword

STEP 1: Repeat steps 1, 2 and 3 of section 3, "Searching for Items by Title". You should see the keyword request cue of figure 4 displayed on the screen.

STEP 2: The text at the top of the screen shown in figure 4 asks you to type one or more TITLE words for an item. Use the keyboard to type the four characters in the word **dogs** and the seven characters in the word **prairie.** As you type a character on the keyboard, it will be displayed on the screen. If you make a typing mistake you can erase it by pressing the backspace key ← . After you have checked that the word is spelled correctly, press the return key ↵. The happy face icons shown in figure 16 will be displayed on the screen.



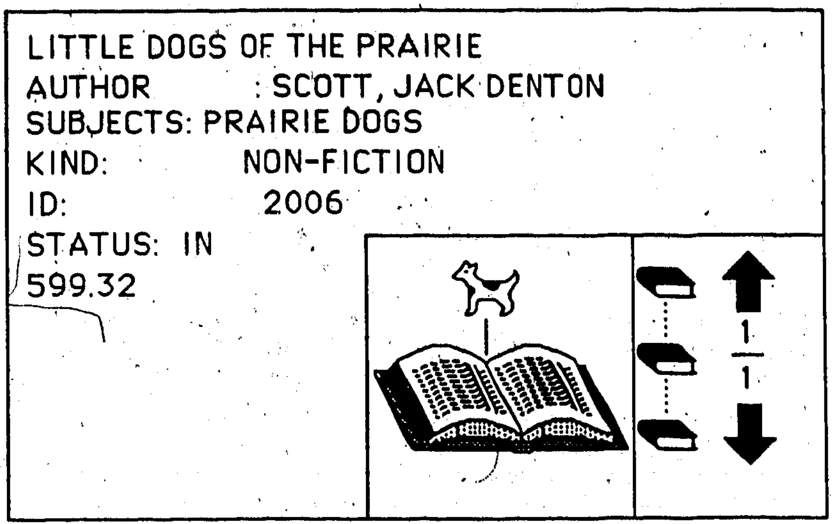
PRAIRIE 10 ☺

DOGS 14 ☺

&

Figure 16. Searching Screen for Title containing Prarie Dogs

This message means that there are 10 items in the which have the word " **prairie** " in the title and 14 items which have " **dogs** "in the title; but only 1 item which has both in the title. Your results will be different since your collection is different. After all the items have been found, the first one will be displayed as shown in figure 17.



LITTLE DOGS OF THE PRAIRIE
AUTHOR : SCOTT, JACK DENTON
SUBJECTS: PRAIRIE DOGS
KIND: NON-FICTION
ID: 2006
STATUS: IN
599.32

Figure 17. Item Information Screen

By using two keywords you can limit the search list to just the title words you are interested in. When you compare this search to that of section 3, you will notice that you did not find the books Prairie Dog and The Owl And The Prairie Dog. This is because the plural dogs is not in the title of these two books.

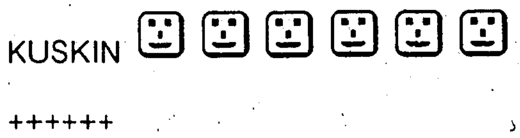
Multiple words can also be used when searching by subject or author.

STEP 3: When you are done viewing the items in this search list, press ☞ to return to the search menu shown in figure 2.

9. Searching for Items by a Favorite Author (or on a Favorite Subject)

STEP 1: To use this procedure, you must have an item by the author whose items you wish to see. Follow steps 1, 2 and 3 of section 5, "Searching for Items By Author" so that the author cue of figure 8 is displayed.

STEP 2: The text at the top of the screen of figure 8 asks you to type one or more AUTHOR names. Instead, use bar code scanner to scan the item you have or use the keyboard to type its item ID. This is equivalent to typing the name of the first author. If the item you have scanned has Kuskin as its first author, you will see the following screen display:

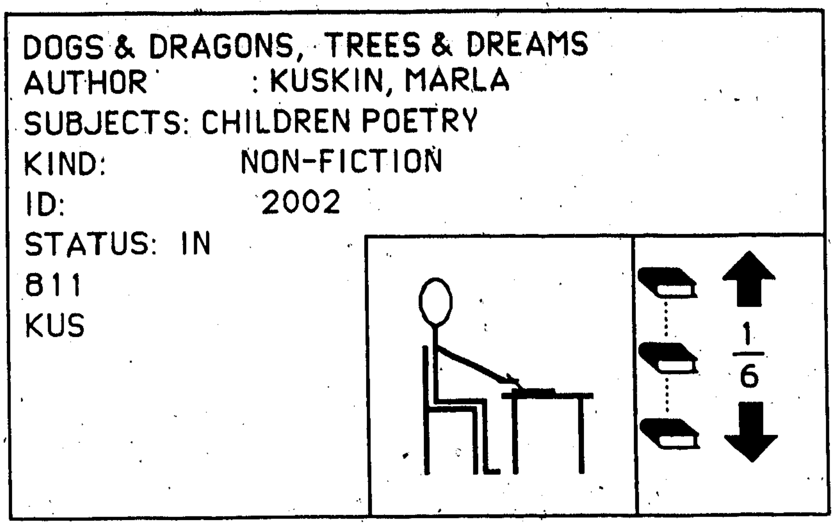


```
KUSKIN ☺☺☺☺☺☺

++++++
```

Figure.18. Searching Screen for Author Kuskin

This message means that there are 6 items who have KUSKIN as an author. After all the items have been found, the first one will be displayed as shown in figure 19. Note that this procedure will also work for subject searches, but not title searches. In the case of a title search, the subject is used instead.



DOGS & DRAGONS, TREES & DREAMS
AUTHOR : KUSKIN, MARLA
SUBJECTS: CHILDREN POETRY
KIND: NON-FICTION
ID: 2002
STATUS: IN
811
KUS

Figure 19. Item Information Screen

STEP 3: When you are done viewing the items in this search list, press ☞ to return to the search menu shown in figure 2.

10. An Unsuccessful Subject (Author or Title) Search

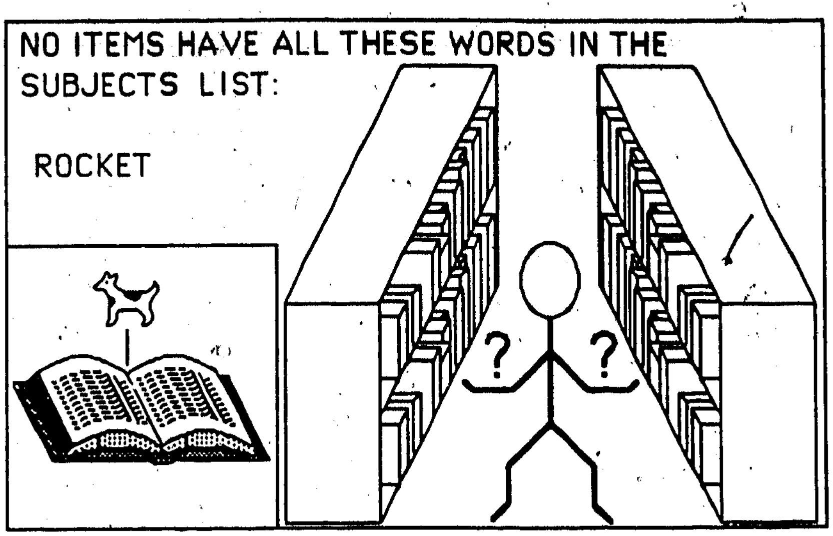
STEP 1: Repeat steps 1, 2 and 3 of section 6, "Searching for Items by Subject". You should see the subject keyword request cue of figure 12 displayed on the screen.

STEP 2: The text at the top of the screen shown in figure 12 asks you to type one or more SUBJECT words for an item. Use the keyboard to type the six characters in the word **rocket**. As you type a character on the keyboard, it will be displayed on the screen. If you make a typing mistake you can erase it by pressing the backspace key ←. After you have checked that the word is spelled correctly, press the return key ↵. The screen shown in figure 20 will be displayed.

ROCKET

Figure 20. Searching Screen

No 😐 beside ROCKET means that "rocket" is a keyword that is not in the subject list of any item. The Searching screen is automatically replaced by the screen of figure 21.



NO ITEMS HAVE ALL THESE WORDS IN THE SUBJECTS LIST:

ROCKET

Figure 21. Unknown Keyword Screen

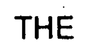
This display may also appear during author and title searches.

STEP 3: Press any key, for example ↵ to return to the search menu.

11. An Unsuccessful Title Search due to Common Words

STEP 1: Repeat steps 1, 2 and 3 of section 3, "Searching for Items by Title". You should see the keyword request cue of figure 4 displayed on the screen.

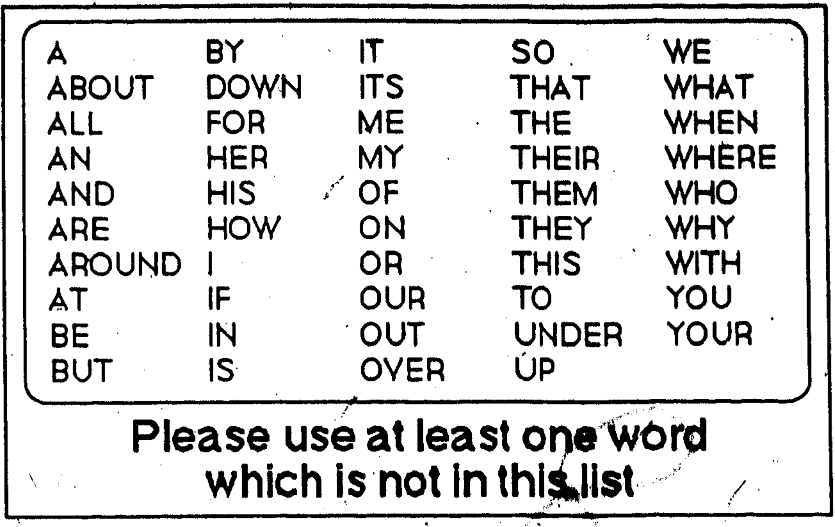
STEP 2: The text at the top of the screen shown in figure 4 asks you to type one or more TITLE words for an item. Use the keyboard to type the three characters in the word **the**. As you type a character on the keyboard, it will be displayed on the screen. If you make a typing mistake you can erase it by pressing the backspace key ←. After you have checked that the word is spelled correctly, press the return key ↵. The screen shown in figure 22 will be displayed.



THE

Figure 22. Searching Screen

and then the screen shown in figure 23 will be displayed.



A
ABOUT
ALL
AN
AND
ARE
AROUND
AT
BE
BUT
BY
DOWN
FOR
HER
HIS
HOW
I
IF
IN
IS
IT
ITS
ME
MY
OF
ON
OR
OUR
OUT
OVER
SO
THAT
THE
THEIR
THEM
THEY
THIS
TO
UNDER
UP
WE
WHAT
WHEN
WHERE
WHO
WHY
WITH
YOU
YOUR

Please use at least one word which is not in this list

Figure 23. Illegal Keyword Display

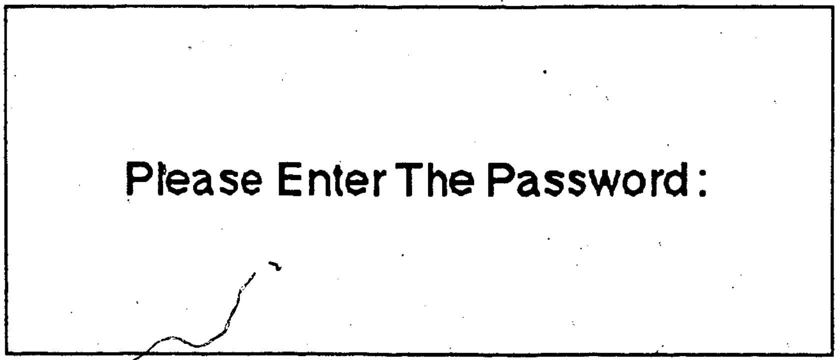
This display means that you have chosen a keyword which is too common. You must use at least one word not given in the list displayed on the screen when using the title search mode.

STEP 3: Press any key, for example ↵ to return to the search menu.

12. Exiting the Circulation System

STEP 1: Press the ☞ key until the circulation menu of figure 1 is displayed.

STEP 2: By pressing the four arrow keys 8↑ 2↓ 6→ 4← it is possible to choose one of the four panes in the circulation menu. As the arrow keys are pressed, the highlighting square will move from pane to pane. When the ↵ key is pressed, the action pictured in the highlighted pane will begin. Use the arrow keys to move the highlighting square to the quit pane and press the ↵ key. The cue of figure 24will be displayed. If you do not see the cue of figure 24, start over from step 1, and this time, make sure that the quit pane is highlighted before pressing the ↵ key.



Please Enter The Password:

Figure 24. The password cue

STEP 3: Enter the password by typing it on the keyboard and pressing the ↵ key. An incorrect password will cause the system to return to the circulation menu of figure 1. If the password is correct, the circulation system will be exited.