7 User interaction design

7.1 Introduction

The expression "user interaction" is preferable to "user interface". The latter is too technically oriented, but nevertheless difficult to avoid in some places.[1]

The process of *interactive* system design is aptly defined by Ramsey and Atwood as follows:

> system whose basic function is communication with a user, and whose basic purpose often is to assist the user with tasks which are cognitive or informational in nature, human factors issues pervade the entire design process. Furthermore, the important design decisions may not be explicitly recognized. It appears that the primary problem-solving method employed in design tasks is an 'analytic/synthetic' approach. This approach involves an analysis of the design problem, in which the components of the problem are identified, followed by the synthesis of a solution. It does not involve an explicit search for alternative solutions; instead, a solution is 'synthesized' based on pattern recognition and the use of components from past solutions to other problems. While this characterization of designer behavior is somewhat speculative, it appears quite compatible with a variety of observations of an empirical, theoretical or pragmatic nature ... Such an approach appears justified ..." [1, p580-581]

This chapter does not pretend to cover the whole issue of the problem of designing a user interface for an IR system. As the quotation points out, this is a very complex and multi-faceted procedure. This chapter only gives general reasons for the particular approach which was chosen, and what resulted from it.

[1] The word "transparent" (which is frequently misused) is also avoided; "invisible" is preferable: it makes it clear that the user neither knows nor sees (and does not need to see) what is going on. Transparent should be used to mean self-evident to the user.

"User interaction" means communication or interactive dialogue taking place between a human being and a computer. Therefore, to design a user interface one has to take into account firstly the users and what is known so far about them (see Section 7.2), and secondly how user interaction in existing OPACs has been tackled (Section 7.3). The main approach for Okapi is explained in Section 7.4, and Section 7.5 describes what was actually done. But, before that, it is worth looking very briefly at research on interactive systems in general.

Interactive systems in general

Much research has been done on interactive computer systems for dedicated users, i.e. users who are expected to learn and remember a specific system. Much of this work has been done on the use of text editing systems, file management systems and data base management systems [2]. Many theoretical contributions and interesting results can be found in various disciplines such as cognitive psychology [3], artificial intelligence, man-machine interaction (MMI) [4], human factors or ergonomics [5,6].

Only a small proportion of this research was looked at, and not in an exhaustive manner. The aspects which are most relevant and most immediately useful to the design of OPACs are: task analysis, the study of query languages [7,8], display formats [9], keyboard layouts [10], and response times; some of these will be mentioned below. The design of interactive systems for general users, such as "public" computerised systems, cash machines, teletext systems [11], especially their "look" and their "aesthetic" or "communicational" quality (what will be referred to later on as "dialogue aspects") is relevant and useful to the design of an OPAC user interface. But interaction with a computer system is also intrinsically related to what type of task is being performed. Using a text editor or a personnel file management system does not involve the same cognitive processes as searching a catalogue. Some time has also to be given to the understanding of information search behaviour and tactics, and research about this in information science [12,13], is of great value. Furthermore, there is a strong interdependence between a task and the particular tool with which that task is carried out: the way a tool such as an OPAC is designed depends on how the IR task is viewed and understood; conversely, a task is tailored by the tool it is performed with. This also is part of the problem of designing a user interface for an interactive IR system.

7.2 Users' reactions to previous online catalogues

It was essential to look at users' reactions to, and use of, online catalogues already in operation, before starting the design of the Okapi user interface. The following results of user studies were considered important factors to take into account. Many of them are taken from the CLR evaluation study [14].

User attitudes

OPAC users are not dedicated users. The average library user generally lacks the time, the knowledge or the willingness to learn to use complicated systems. Nevertheless, users like OPACs and the time spent at terminals is higher than the time spent at conventional catalogues. This might be because OPACs seem more fun to use: they are like computer games. This is surprising since a large majority of the population has rather poor typing ability. Users have high expectations and tend to trust OPACs (these are not incompatible); this is probably due to the interactive nature of OPACs, which also explains their resemblance to computer games.

However, online catalogues have obvious advantages over conventional catalogues; for example, users like having all entry points in the same place rather than having to walk along the drawers or to select different microfiches.

It seems that many library users are not motivated enough to do very much in the way of learning: a majority of them learn on their own; they are reluctant to spend time at it. Indeed, if they have to undergo a substantial learning process, both human and catalogue resources are being wasted. When it comes to the actual usage of OPACs, the following results suggest that users do not seem to learn very much: they tend to repeat mistakes and get into loops; they remain in the same search "state" for a long time; most users don't know much about bibliographic records; they usually want short record displays, rather than "full" records. Users do not show much perseverence either: "it is evident that what is not found in the first place a user looks, is often not found or used at all" [15].

Importance of ease of use

The very first objective is therefore to design a system which does not require the user to spend time learning how to use it, i.e. one which is easy to use or, more precisely, usable at sight. Users normally do not know much about the tools used for catalogue searching (why should they?), while at the same time they are becoming more demanding of computer systems in general.

Ease of use is not a simple concept. The designer's task therefore is to improve subject searching and provide help; but two factors have to be taken into account: (a) users cannot be expected to know much about tools such as subject headings and thesauri, and (b) users would also like subject aids to be visible online (or transparent, see footnote (1) above) — probably because they would like to feel in control, but it might also improve effectiveness (recall/precision). These two factors are apparently contradictory, and to achieve ease of use in this context is not trivial.

Variety of users

Another important property of OPAC users is the fact that they form a very heterogeneous population. If one looks at this population in a more detailed manner, users can be categorised in many ways: by experience of the OPAC, by experience with catalogues in general, by experience of, or attitude to, computers. The first of these categories can be broken down into (1) first time users, (2) experienced users and (3) others; (3) contains those people who have used the system only a few times and also those who may have been familiar with it at one time but have forgotten it in detail.

> "I use the computer catalog every six months. Every time I come to use the system, I have to start over... I have never been successful remembering (how to use the system). Make it menu-driven. Takes me an hour or two to remember" (one frequent Library of Congress Science Reading Room user reported in [16]).

It is not difficult to see that these three categories of users have very different requirements. There are systems which really only cater properly for type (1), e.g. the CLSI touch-screen system or Geac circulation systems, while others are rather good for type (2), e.g. the Library of Congress system. Some systems (e.g. MELVYL at the University of California or the BRS/SEARCH-derived Dartmouth College system) offer two levels of user interaction, aimed at inexperienced and experienced users respectively.

The issue of catering for a varied population, from inexpert to sophisticated users, will be developed in more detail in Section 7.4.1.

A myriad of adjectives can be found in the literature to identify the sort of users OPACs have to deal with. Here is a selection: naive, novice, casual, in/expert, in/frequent, un/trained, in/experienced, non/dedicated, un/sophisticated, un/familiar, general, end-user. The reader will have come across some of them already in this report. The trouble is that most of them express only the two extremes (e.g. experienced/inexperienced), and not the fact that users form a multi-dimensional continuum.

The next section looks at how existing OPACs have coped with these important human factors issues and how they have tackled user interaction, especially learning and ease of use.

7.3 Previous OPAC designs and their user interfaces

It is impossible to envisage a particular OPAC user interface without relating it to the more fundamental design features of the whole system.(1) This section attempts to analyse the effects of the tool design on the actual interaction taking place for different types of OPACs.

The three "generations" of OPACs were introduced in Section 2.4. The main characteristics of the first two generations of OPACs in terms of their design, IR functions and dialogue are summarised in Table 7.1.(2) Charles Hildreth gives a more detailed table of these three generations in [17, p41].

(1) Moreover, the user interaction is the most essential aspect — if not the purpose — of an interactive system such as an online catalogue. Therefore it has to be considered from the very start in the overall design, rather than as an afterthought (so that the system is flexible enough and has the potential for quite "clever" access) (see Section 2.7.1).

(2) The third generation of OPACs (such as CITE or Paperchase) is not included in this table. CITE has been described briefly in Section 2.4.4 from the point of view of its design. In terms of its user interaction, it is probably not worth looking at it in great detail, as it is not suitable for general users: CITE is implemented on medical files, and its users have to be fairly knowledgeable in this particular field in order to understand the structure of MeSH controlled vocabulary, and be dedicated enough.

As we have seen, pre-coordinate OPACs are modelled on card/microform catalogues and early OPACs adopted the same file structure (often two ordered sequences of entries — author and title). The phrase-match model is directly related to the fact that conventional catalogues are primarily used, if not designed, for known-item searches: catalogues are (were) seen as a "representation" of the library, itself seen as a store of uniquely defined items. Catalogues have been designed in the past primarily as tools for librarians rather than for library users. Searching by subject in conventional catalogues is via a classification scheme (in the UK anyway, see Section 2.5.2) corresponding to the task of book shelving. Not surprisingly, many library users (in public libraries for example) never use the catalogue for subject searching, finding it easier to browse through the shelves. Nevertheless, library users do try to do subject searches in conventional author/title catalogues (see Section 2.5 and [18]).

Table 7.1. Summary of the first two generations of OPACs

OPAC DESIGNS	
First generation	**Second generation**
– Pre-coordinate	– Post-coordinate
– Phrase matching	– Keyword-searching
– Early OPACs	– More recent OPACs
– Derived from circulation or cataloguing systems	– Derived from traditional reference retrieval systems
– Known-item oriented	– Subject oriented
– Controlled vocabulary and/or Classification scheme	– the same plus free-text
IR function	
– Few access points	– All words are access points
– Intolerant of mistakes in word order	– False drops due to "false coordination"
Dialogue	
– Menu-driven, form-filling, or hardware dependent (touch-screen panels)	– Command language, often simplified. Sometimes two levels: naive level menu-driven, experienced level command language
– "computerese" and "Biblish"	– "computerese" and "IRish"

As explained in Section 2.5, the second generation of OPACs is derived from Boolean IR systems. With these OPACs and their free text searching capabilities, an even bigger proportion of searches are subject searches, not surprisingly. It is also true that conventional IR systems such as DIALOG are used (and designed) mainly for subject searches because of the content of their files (journal articles) and their purpose (IR for research purposes mainly), and also because these systems do not lead to a shelf position, and so are less immediately useful for finding specific items. Keyword searching is appropriate (and was designed) for free text searching of uncontrolled vocabulary fields, especially abstracts, in document citations in specialised domains.

These two types of OPAC design (for pre-coordinate and post-coordinate searching) are now examined from the user interaction viewpoint, that is:

the success of the IR function or activity, e.g. how the system helps the user with retrieval failures (see Section 7.3.1)

the dialogue taking place between the user and the machine, i.e. the clarity of screens, messages, instructions, input, general navigation (going back and forth), understanding (see Section 7.3.2).

7.3.1 IR aspects

The task an OPAC is supposed to help with is an IR task; the way the system helps in performing this task depends on its IR features as well as on how these features are presented to the user.

Phrase indexed OPACs

IR using the phrase matching function often creates problems for the user: the search argument has to match the author, title or subject entry word by word, in the order specified, beginning with the left most significant characters; the user has to know at least the exact beginning of the title, subject heading (e.g. "WITCHCRAFT—FRANCE—AUVERGNE") or class number. There are not enough access points.

Similarly to the way one uses a microform or book catalogue[1] when there is no match, phrase indexed OPACs usually display the "nearest" match

[1] Except that scanning through screens of records on a computer is slower: more tiring for the eye, and very few records can be shown on one screen of a standard VDU (see Section 2.6.3).

surrounded by the alphabetically close entries in the indexes (see Section 2.4.1), following some filing rules, which the user also has to know.

Alphabetical order is obviously a convenient sort of order, particularly because people are used to it (card catalogues, dictionaries, telephone directories).[1] But it may or may not be helpful in retrieving information (and users see it as a matter of luck, or blame themselves): this alphabetical principle can be quite successful for finding morphologically related items [19]; a search for a misspelt author may succeed by looking around the nearest match; but it is much less useful — and users do complain about the difficulty of subject searching — for semantically related items such as subject headings: see Fig. 7.1 for an example of a subject search with a pre-coordinate OPAC.

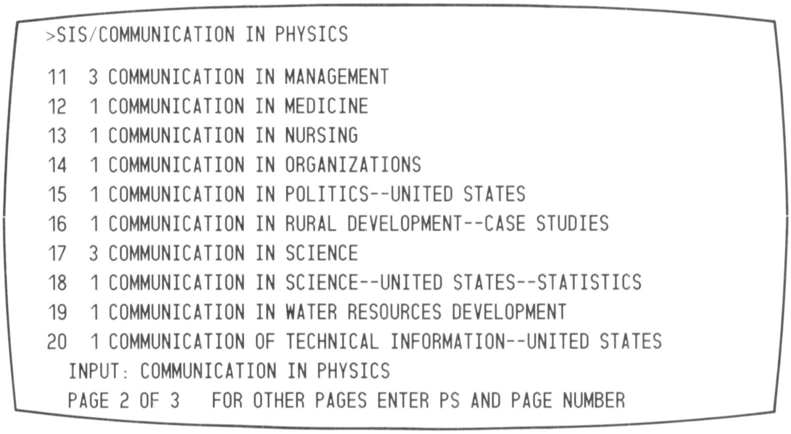
Browsing and phrase indexed OPACs

Browsing is also mentioned in Chapters 2, 5 and 9. It is an important IR activity taking place in catalogue searching, which is connected in some way with retrieval failure. It is important to look at how retrieval failures are dealt with in terms of user interaction i.e. how, if at all, browsing facilities are presented to the user.

Users often have "browsing" requirements, especially in subject searching. They often complain of not being able to see many records in OPACs, unlike in a COM catalogue, where they have the feeling they have seen "everything"; students at the Polytechnic of the South Bank in London have been observed looking through screens and screens of brief records on Geac [20]. Obviously, users like and need to see an extensive range of records.

Phrase indexed OPACs seem to provide this: the browsing is provided on an alphabetical basis, with the problems already mentioned for semantically related items. Nevertheless, these OPACs have the great advantage of showing something when no match is found (see Section 2.4.1), which is equivalent to providing some browsing automatically (the user might not know that s/he wants to browse).

[1] However, some members of the public do not know the alphabet, and in some libraries it has had to be pasted onto touch-screen OPACs (where knowledge of the alphabet is crucial, see Section 7.3.2).



```
>SIS/COMMUNICATION IN PHYSICS

11  3 COMMUNICATION IN MANAGEMENT
12  1 COMMUNICATION IN MEDICINE
13  1 COMMUNICATION IN NURSING
14  1 COMMUNICATION IN ORGANIZATIONS
15  1 COMMUNICATION IN POLITICS--UNITED STATES
16  1 COMMUNICATION IN RURAL DEVELOPMENT--CASE STUDIES
17  3 COMMUNICATION IN SCIENCE
18  1 COMMUNICATION IN SCIENCE--UNITED STATES--STATISTICS
19  1 COMMUNICATION IN WATER RESOURCES DEVELOPMENT
20  1 COMMUNICATION OF TECHNICAL INFORMATION--UNITED STATES
  INPUT: COMMUNICATION IN PHYSICS
  PAGE 2 OF 3   FOR OTHER PAGES ENTER PS AND PAGE NUMBER
```

Figure 7.1. Subject search in LCS (Ohio State University).
The system provides an alphabetical display of available subject headings [29].

Keyword OPACs

With OPACs, such as MELVYL, which offer keyword searching, subject searches are on the whole more successful than they are with phrase-matching OPACs, mainly because the user does not need to have any knowledge of subject headings to have some chance of success. There are more access points (see Section 2.4). On the other hand, keyword searching can be less effective than phrase-matching in retrieving specific items (e.g. author "JOHN FRANCIS" or title "WAR AND PEACE" retrieve false drops when the two words are ANDed).

Nevertheless, free text searching is not ideal either and has its problems for subject searching, particularly in OPACs compared with conventional reference retrieval systems, because of the nature of library catalogues:

(a) the nature of MARC records (see Section 2.4.2):

> users want subject enrichment of the catalogue, such as table of contents, book indexes, etc. — see [21] and Section 9.4.5 for ways of enriching catalogue records — "Keyword/Boolean searching alone on the vocabulary of title and subject fields of the MARC record may not improve recall because works assigned the same or related subject headings may be missed" [22].

(b) the wide coverage of library files (see Section 2.2.1):

> keyword searching is not precise searching, false term combinations are common, and this is increased if the coverage is wide (see also [23] for research on the use of classification schemes in OPACs).

Browsing and keyword OPACs

Unfortunately, most keyword OPACs do not provide much opportunity for browsing: when there is no match, the search fails and the burden is on ıhe user to reformulate it or to give up. Charles Hildreth indicated in 1982 that "the ability to view displays of subject headings or titles grouped by the common occurrence of one or more keywords entered by the searcher is a rare browsing feature in today's online catalogs" [24]. More recently, a few OPACs have incorporated some browsing facilities, such as browsing through subject authority files and cross-references, but they have to be asked for explicitly by the user and, as was said earlier, users might not think of doing so.

There is another rather important case of retrieval failure which occurs much more often with keyword OPACs than phrase OPACs: this is the "too many hits" case, which can occur quite frequently with a large library file. In most systems the only possible action for the user is to scan through the records, hoping to see something relevant, and to give up if no relevant item appears after some time. Some OPACs offer "limit" facilities (by date, by language, by site), which ought to be suggested to the user rather than having to be requested (since s/he might not think of doing so).

There are also more technical problems about browsing, such as whether it is better to browse through the indexes rather than through brief records.

Phrase versus keyword

To summarise, phrase-matching is more appropriate for retrieving specific items because of its "morphological" bias [19]; keyword access is better than phrase-matching for subject searching, because it gives more access points. Okapi incorporates both. As stated in Section 2.5 and in [25] both contribute towards the success of retrieval; but this is only half the answer: even with sophisticated IR systems there will always be retrieval failures; there remains the problem of helping users with these failures, by providing such facilities as browsing, in a sensible, useful and active manner.

7.3.2 Dialogue aspects

Touch-screen OPACs

Regarding the dialogue aspects of user interaction, mention should be made first of touch-screen type interfaces. The touch-screen OPACs (CLSI, ALS Browser) have been popular in public libraries and with children [26]. Their dialogue is very hardware-specific, and the IR function almost identical to other pre-coordinate systems. Instead of having to type her/his search terms the user selects, by touching, the wanted term (or, if it is not there, the one preceding it) from an initial alphabetical display of headings; then the system presents progressively narrower lists until the sought item is found. The input procedure looks like this:

```
Anorexia
Cat    Touching ''cat'' will give the following list:
Feminism
.........                    Cat
Nuclear                      Dinosaur
Unemployment                 Espionage
Zimbabwe                     Feminism      etc.
```

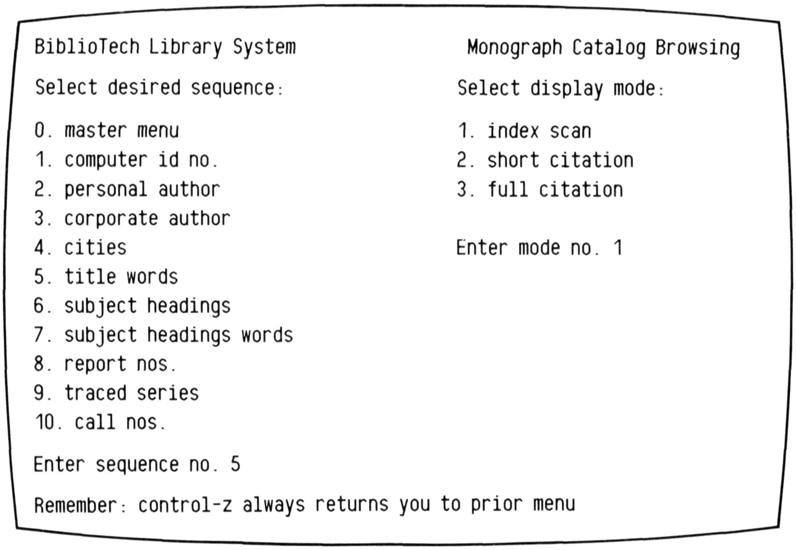
With CLSI/PAC 1 it takes between six and nine touches to get to a record. This type of procedure can become quite tedious with large files; some of these systems also make the reader start the selection again from the beginning if a mistake is made.

Other OPACs

OPAC dialogues on standard terminals (VDU and keyboard) either use menus, sometimes with form-filling type of input, or a command language (see Table 7.1).

Vocabulary

Many OPACs make use of codes, abbreviations, jargon and syntax that users have first to understand (they might not reach that point) and then learn. Some of the jargon comes from the cataloguing world; for example in the first menu, some systems ask users to choose to search one of the following fields: “uniform title”, “main entry”, “added entries”, “LCCN”, “ISBN” and other library jargon or “biblish” (see Fig. 7.2). Most users do not know what corporate authors are, sometimes cannot distinguish between author, title and subject, or do not know that an author is sometimes an author, sometimes a title, sometimes a subject, and that the computer needs to know which is wanted [27]. People may have to understand instructions and commands in computing jargon or “computerese” e.g. “control-z returns you to prior menu”, “Type PS and RETURN”, “Enter code letter then press <SEND>”, “ENTER , [CR], [ESC], + ‘word’:” or messages in IR jargon or “IRish” such as “ANDs were assumed for missing Boolean operators” or “10 records matches after term cookery”.



```
BiblioTech Library System                  Monograph Catalog Browsing

Select desired sequence:                  Select display mode:

0. master menu                            1. index scan
1. computer id no.                        2. short citation
2. personal author                        3. full citation
3. corporate author
4. cities                                 Enter mode no. 1
5. title words
6. subject headings
7. subject headings words
8. report nos.
9. traced series
10. call nos.

Enter sequence no. 5

Remember: control-z always returns you to prior menu
```

Figure 7.2. Main menu in Bibliotech (Advanced Management System) [29]

Record displays

Similarly, displays often include tags, mysterious numbers, peculiar labels and can be quite bewildering for the general public.

Modes of input

There may be input problems: to input an author's name, users have to follow some sort of syntax ("A=KROPOTKIN, PETR", "AU SKINNER B F") and this can create problems, especially since some systems are rather intolerant of syntax mistakes (if a space or a comma is missed out, the system comes up with error messages such as "Input error 122"). Names like "LLOYD GEORGE" or "HENRY JAMES" may be difficult to input.

In some cases, the user has to build a derived or acronym key (similar to catalogu*ing* systems), like an "author/title" key: for example the first four characters of the author and the first five characters of the title ("ATS/TWAIHUCKL" in the LCS system). Sometimes, users have to know what leading articles are so that they can avoid typing them in.

Help

When help screens are provided, people quite often don't use them:

> "HELP is useless. If anyone ever touches HELP, they immediately get out of it. (Technical services staff)." [28].

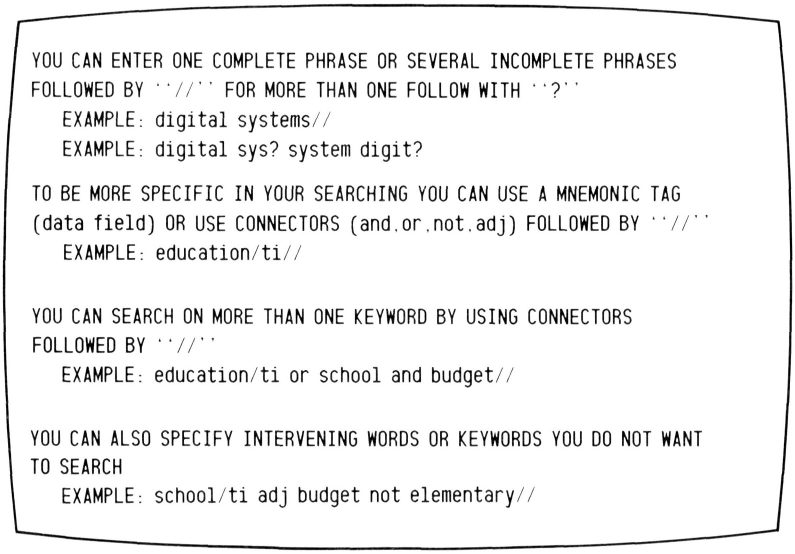
This is hardly surprising if they are like Fig. 7.3, which is an example of a help screen in a keyword/Boolean OPAC. Help is often designed almost as a documentation of the system, rather than as advice appropriate to a specific stage of a search.

Menu

The menu approach can be cumbersome and tedious to use; it is sometimes difficult to get out of a particular option, and on the other hand, if many options are available all the time, it becomes tedious to have to answer the same questions over and over again. If a mistake has been made it may be necessary to start again almost from the beginning.

Command languages

Keyword OPACs sometimes provide a complicated command language, with explicit use of Boolean operators, positional operators, limit functions, truncation, etc.



```
YOU CAN ENTER ONE COMPLETE PHRASE OR SEVERAL INCOMPLETE PHRASES
FOLLOWED BY ``//'' FOR MORE THAN ONE FOLLOW WITH ``?''
   EXAMPLE: digital systems//
   EXAMPLE: digital sys? system digit?

TO BE MORE SPECIFIC IN YOUR SEARCHING YOU CAN USE A MNEMONIC TAG
(data field) OR USE CONNECTORS (and,or,not,adj) FOLLOWED BY ``//''
   EXAMPLE: education/ti//

YOU CAN SEARCH ON MORE THAN ONE KEYWORD BY USING CONNECTORS
FOLLOWED BY ``//''
   EXAMPLE: education/ti or school and budget//

YOU CAN ALSO SPECIFY INTERVENING WORDS OR KEYWORDS YOU DO NOT WANT
TO SEARCH
   EXAMPLE: school/ti adj budget not elementary//
```

Figure 7.3. User asks for help in Datalib [29]

Here are some examples of possible commands from several OPACs (most of them as reported by Matthews in [29]):

```
COMBINE SU MATHEMATICS STUDY AND TEACHING
FIND TI MEASUREMENT AND PA LANCASTER
SEARCH 3 AND (ENERGY OR POWER) AND CANADA
FIND SU ENERG? AND CONSERVATION AND SU HOUSES AND DATE 1980-
COMBINE S012 AND S015
F (ROBERT WITH PENN WITH WARREN). ME,AE.
DISPLAY 1-14 REVIEW
DIS 4-8 CAL SUB
A=OHARA J OR O'HARA J
A=ELIOT OR ELIOT T
BROWSE AU HOFFMAN M
SCHOOL/TI ADJ BUDGET NOT ELEMENTARY//
(CORRUPTION)/SU AND (GOVERNMENT)
R6 AND CALCUL$
```

Obviously, people have to be taught what Boolean searching is in order to use these command languages. Inexperienced users do not understand the logic of Boolean searching (AND, OR, NOT), and cannot perform it effectively without extensive training and subsequent regular use. Apart from the learning problem, formulating search strategies using Boolean operators is not a natural activity for the naive user [22].

Not surprisingly, when the use of OPACs incorporating sophisticated retrieval functions is investigated it has been found that most of these functions are not used. In MELVYL, it was found that the majority of the searchers used the easy "LOOKUP" mode (menu-driven) which does not make available most of the more sophisticated features of the system [30].

7.4 General approach to Okapi user interaction

Although the discussion of user interaction is near the end of this report, and it was among the last parts of Okapi actually to be implemented, user interaction was under discussion during the whole design and development of Okapi. User interaction was the first design consideration (Chapter 2 and Section 7.3), and, as will be obvious to the conscientious reader, it influenced the way the constraints of the hardware were overcome (Chapter 3), the content and structure of the source file (Chapter 4 and Section 7.5.3), the design and structure of the indexes (Chapter 5 and Section 7.4.3), and the choice and implementation of the search functions (Chapter 6 and Section 7.4.3). Okapi is an OPAC; an OPAC is an interactive system for general users; its whole *raison d'être* lies in its effective interaction with the user.

7.4.1 How many interaction levels?

As mentioned in Section 7.2, OPAC users vary from totally inexperienced to sophisticated. So far, the only way of trying to cope with this diversity of users is that found in keyword-derived OPACs such as MELVYL and Dartmouth College BRS, where two levels of interaction are offered: the first type (menu-driven, similar to what phrase indexed OPACs usually offer) which copes with naive users at one end of the spectrum, and the second type (command language) which copes with sophisticated users at the other extreme.

These two levels of interaction are broadly summarised in Table 7.2.

The philosophy behind the provision of these two levels is that the inexperienced level ought to teach the user and lead her/him to the expert level. Ideally, an *adaptive* interface would make a link between the two levels more or less automatically (i.e. by the system doing some sort of translation), or with the possibility of switching from one to the other easily (anywhere and any time during a session); but nothing of the sort has been achieved yet (see Section 9.4.4).

Table 7.2. Two levels of interaction

UNFAMILIAR USER	*FAMILIAR USER*
Menu-driven	Command language
Fill-in format, prompting	
"Window" (paging)	Scrolling
Touch-screen (could be mouse or light-pen as well)	Keyboard
Single search input: phrase-searching or automatic Booleans (AND only)	Boolean operators (sometimes simplified or in a guided form) Truncation
TYPES OF SEARCHES (see Section 7.4.2)	
Prompted choice: user has to choose among suggested field(s) to be searched, e.g. author/title title phrase, title word, class mark, subject heading.	User-specified "free" choice: user has to ask for field to be searched, e.g. "SMITH/ME/AE" or "SU=ECONOMICS", or "multi-meaning" message

For example, MELVYL users are expected to use the menu-driven "LOOKUP" mode *to start with*; it is seen as a first stage, before getting into the "COMMAND" mode which is more flexible and more efficient. MELVYL user interface specifications [31] included a "tutorial" mode, which, it was hoped, would establish a link between the two types of dialogue by explaining and teaching the basics of Boolean logic and introducing the expert level command language. Unfortunately, this tutorial mode has not been implemented. Most people still use the simple "LOOKUP" mode [30]. At the University of California, instead of having an online tutorial, they provide training on the grounds that, the "LOOKUP" mode being a "friendly" interface, "users do not exploit it to the fullest, because with a minimum of effort they get results and are lulled into believing they can use the system quite well" [32].

Linda Arret [33] and Anne Lipow [27] both argue that easy to use menu-driven interfaces, as well as making users underestimate the capabilities of a system, do not give enough power and flexibility to the user; most of all, they do not give the user the possibility of developing mastery or even competence in using retrieval tools. Their answer seems to be that, in order to be efficient, a user has to learn to use a command language. One could argue that by trying to train and force people to learn something like a command language, many users will be put off and stop using it altogether. Some of the training also includes such trivialities as teaching users to type a "1" rather than an "l" (see [32]), which computers can recognise and signal to the user rather easily.

Okapi interaction philosophy is that it ought to be possible to do without a command language altogether (so that users need not be taught Boolean logic, nor remember mnemonics like "FI AU="), without diminishing retrieval power. It is doubtful whether the use of a command language is appropriate even for experienced users of an OPAC (this may also apply to other reference retrieval systems). An argument against it is that the kind of mental approach used for subject searching is much more subtle than that which can be expressed using a command language. The use of Boolean operators may interfere with the formulation and refinement of a search query. Command languages are used because they are the easiest way of implementing access to post-coordinate inverted file systems. Most OPAC users need help in formulating and refining their search; it is questionable whether being taught a command language helps them in doing this.

Our fundamental approach is almost the opposite of MELVYL's: our goal is not that OPAC users should become expert in the use of Boolean logic, nor to try to impose it on users. Therefore, the emphasis was put on ease of use for novice users. One of the reasons is that this is a good situation for designers to learn not to impose techniques on the user — not only IR techniques such as Boolean searching, but also library and computer-oriented techniques. It was more challenging and interesting than designing a straightforward command language, since it involved making the system more clever "behind the screen" (see Section 7.4.3) as well as making it clear on the screen.

To sum up, ease of use means that users are given as much retrieval power (see Section 7.4.3) as possible without having to know about Boolean techniques, cataloguing rules or computers.

Okapi's single level of interaction is a combination of selection from options (choosing records, types of search), prompted form-filling (entry of specific item search terms) and free language input (entry of subject search topic). Instructions are kept simple and there are as few of them as possible, because naive users like to get a feeling of relief and achievement quite quickly while performing a task with a computer. This is what Shneiderman [34] refers to as the "closure" factor: users prefer to go through multiple small steps rather than a single complex one (e.g. they prefer three different screens rather than three menus on the same screen).

7.4.2 How to offer search types

There are two alternatives to be found in the way existing OPACs offer types of searches to the user (see Table 7.2).

FIRST ALTERNATIVE: A LIST OF OPTIONS

The first alternative is that of prompting the user to choose among a list of fields to be searched (see Fig. 7.1). This originated in pre-coordinate OPACs and is convenient for phrase-matching. Some pre-combination of fields is often included in the options given, for example with an author/title option (like Geac), or an author option searching both personal author and corporate author indexes. Users might not understand what these options mean (see Section 7.3.2).

If keyword searching is used without a command language (see Section 7.4.1), which would give the possibility of combining any field with any other, the system has to prompt the user for an AND (allowing the entry of only one term at a time) so that different fields can be combined together. Dartmouth College User Cordial Interface [35] uses this method by prompting "do you want to add more to your search statement?" and if the answer is YES, asking the user to choose the search type (again) of the element to add. This may be cumbersome and tedious.

Second alternative: "unrestricted" search

The second alternative is to provide an "unrestricted" or "unconditional" search so there is no need for a rigid choice of searchable fields to select from, nor any need to specify in advance which field(s) is/are going to be searched. This method is usually a feature of the command languages of online reference retrieval systems and gives a feeling of flexibility and freedom. But this could be puzzling for the general user: looking for "BROWN" for instance could bring in all sorts of different objects and look silly. It is difficult to explain to the user what is happening. Systems' response is often a "multi-meaning" message: for example in a "keyword" search of the OCLC LS/2000 at Newcastle University [36], after typing in a word the user is presented with a list of corresponding index entries and asked to choose among them. This then becomes the same as the first alternative.

One of the biggest problems in free text searching is the input and processing of personal names. There are good arguments, from the point of view of system efficiency, for making users say that s/he is looking for a name. A completely unrestricted search facility does not seem to be the solution, even if it is a tempting one.

The Okapi solution

The Okapi solution is a compromise between these two alternatives: a very small number of options, but less freedom than in the second alternative.

Okapi's first screen simply gives a choice between two types of searches described as a "specific book search" (if author and/or title are known) and a "book about something search" (any topic you have in mind) — see Fig. 7.5.

The traditional distinction between "known-item" and "subject" searches in catalogue use has been described in Section 2.5.

> "The distinction, although not entirely valueless, is not well defined. A search which starts as known-item often becomes a search for books which are in some way related to the item first sought. Should such a search be defined as a subject or a known-item search?" [37].

Nevertheless, it seems likely that one or the other of these two definitions is a reasonable description of most catalogue searches. This distinction certainly has the virtues of clarity and simplicity.

The "specific book search" allows searching on title, author or both. The "book about something search" combines title and subject term search. It is also technically possible to search for class numbers, but there was not enough time to work out a way of explaining to the users that it is available.

7.4.3 *Automatic features: behind the screen...*

AUTOMATIC FIELD SEARCHING

Okapi's two types of searches are a reduction from the first alternative mentioned above (Section 7.4.2): instead of making people choose among a list of fields which correspond to the nature of the entries kept in the index, Okapi bridges the gap between the two types of searches and the index by automatically assuming which field(s) are going to be searched. Different data types or "beasts" (Section 5.2.3) are appropriate for searching by the system depending on the type of search chosen. The structure and choice of beasts describing the nature of each index term (which field it comes from) is derived from what it was assumed the user would know about what s/he is looking for, and the best use which the system could make of this information. (There is a list of the available beasts in Section 5.2.3.)

Specific-item search

The decision not to distinguish in the index between a personal name "main entry" and a personal name "added entry", was based on the assumption that users would not know (nor need to know) the difference; when the user inputs a personal name, Okapi searches all personal names — no information is retained in the index about their main or added entry source (record displays, however, do convey this information).

Similarly, users are not expected fully to understand the difference between a personal and corporate author (most people do not think of institutions as authors). It was decided not to make users choose between personal and corporate when doing an author search. The index, therefore, does not differentiate between personal and corporate author; they are merged into a unique beast (Sections 5.2.3 and 6.8).

Users are not asked to build a derived author/title key, but Okapi builds it automatically from the user's input. This leads to problems which were not resolved properly in the prototype: (a) how to make people realise that only the first four characters are needed, so that they type as little as possible, and (b) there are a number of author/title "false drops": the system ought to be cautious when an author/title key retrieves more than, say, three items. In such cases the system should use more of the information in the search keys, or seek elucidation from the user, but this has not been implemented because of lack of time.

Leading articles and stop words are automatically removed from title and subject entries so far as possible (see Section 5.5.1) to avoid the user having to be aware of them.

Subject search

Subject searching uses the following: subject words, name subject headings (phrase and word), title words, corporate author words (Section 5.2.3). Title and corporate author fields are used because they contain subject information (many of the Okapi records have no subject headings — see Section 2.5.2). Title phrases and subject phrases are not included; title phrases can be misleading in subject searching, but subject phrases ought to have been included (see *automatic displays* below).

Phrase and word searching

Because of the nature of catalogues, phrase and keyword approaches are both necessary and are complementary (see Sections 2.5 and 7.3.1). The index keeps both. For example, there is an "author phrase" beast which includes personal name "phrases" (surname + initials) and corporate phrases, and an "author word" beast which include surnames and corporate words (see Section 5.2.3). Surnames (or "words") are kept because users may not know initials. They are also useful, theoretically, for truncation purposes (see below). This also affected the design of the input "form": Okapi prompts for initials only if less than three words were typed as "surname" (see Fig. 7.12), otherwise it assumes a corporate author was entered.

Search trees

Although users know the difference between a word and a phrase, they cannot be expected to understand how this affects retrieval. Okapi uses both phrases and words, making the greatest possible use of what is entered without further intervention from the user. It follows the "search decision trees" described in Section 6.8 and illustrated in Section 7.5.2.

The search decision trees use these word and phrase beasts in a helpful way, as well as the other beasts. According to the type of search chosen and what was entered by the user, Okapi will try a phrase search (for example an author phrase) before trying a word search (author "word" or surname). See Section 7.5.2 for more examples. The principle underlying the search trees and their use of the beasts is that they do not leave the user to have to cope on her/his own in case of retrieval failure, and they try all possibilities before reporting failure to the user.

Again because of lack of time, Okapi search trees were not developed as fully as the team had hoped: the beasts could have been used even further in the search trees (for example, users sometimes cannot see that an author can be a subject as well); in some cases the number of postings retrieved could be taken into account, for example in surname search (e.g. "SMITH" and "CZERNIEWSKA" should not be treated in the same way); it was also hoped to try to use subject *phrases* before combining subject words. It would be extremely valuable to test and evaluate these trees thoroughly in "live" use, in order to modify them or suggest new branches and alternative structures.

Implicit booleans

The search trees enable Okapi to perform normal Boolean searching automatically: for example when no phrase match is found and more than one word is entered, the words are ANDed. This overcomes the necessity of teaching the use of Boolean operators.

To overcome the "no match" situation when several words are ANDed, Okapi uses the "hyper-OR", described in Section 6.5. This gives users the opportunity to *browse* through ranked records. See Section 9.4.1 for suggestions to improve this by involving the user and asking her/him to judge the relevance of the records retrieved. OPACs in general have not implemented Boolean techniques in a helpful way in the case of IR problems such as no match, too many matches, browsing requirements.

> "If there is no exact match *and* there is more than one word in the search key, it is sensible for the system to report this and offer to look

for items *containing* all (or even 2 or more of) the individual words in the key (i.e. AND them automatically)" [22].

AUTOMATIC DISPLAYS

As well as performing various IR functions automatically, Okapi also tries to behave in a sensible way with the presentation of the results of a search to the user, and with providing browsing or suggesting other actions, in case of complete failure.

The system always tells the user the number of postings retrieved; then it offers different kinds of displays according to the result of the search, i.e. whether something was retrieved, and the number of postings. Table 7.3 gives a brief summary of this principle.

Table 7.3. Automatic displays

No. of Postings	Default Display /Suggested action (after informing user of number of postings)
IF EXACT MATCH	
100 or more postings	Display short records and warning, after having asked if user would like to add another word or phrase
5 - 99 postings	Short records displayed (first 5-10 records)
fewer than 5 postings	Full records displayed
PARTIAL MATCH	Index display
NO EXACT MATCH	Index display

In case of retrieval failures, one solution is to present the user with an index display. But there is no general rule. For example, it is questionable whether, having typed a surname because the initials were not known, getting an author display is of any use when there are many authors with that surname. Brief records would be preferable. Do title phrases have a browsing usage when not even a partial match was found on a one word title?

Some testing of and experimenting with possible procedures would have been very valuable; again, time did not permit this.

Truncation

As mentioned in Section 6.9, an explicit truncation option was not provided. Although easy to implement (see Section 6.4), it is difficult to explain to the user (see Section 6.9). In any case, an OPAC should provide some degree of automatic synonym generation. This is one of the subjects of a proposal for further work on Okapi [38].

Messages

Although all these things happen behind the screen, automatically, the general approach is to explain what is going on as much as possible, in simple terms, so that Okapi is not a completely "invisible" system (see for example the message **searching for title "drea.." & author "freu.."** in Fig 7.21). This was found rather difficult to do and it is not certain that it was achieved in a very satisfactory way. One reason is the small size of the screen, but a more important factor is the necessity, and difficulty, of making a distinction between *information* (which may be useful but is not essential), and *options* which must be read so that the user knows how to proceed. (Users rarely read everything on the screen, and tend to pay attention only to the more obvious messages.) Display devices offering a choice of fonts and/or colours would be a great advantage here. "A well-designed page of printed material has a density loading of about 40%: CRT display screens that are subjectively preferred have a density loading of only 15%" [29] — see also Section 9.4.3.

7.5 Structure and description of Okapi dialogue

7.5.1 User input

The user is not assumed to be good at typing nor to be familiar with the Apple IIe's keyboard: a QWERTY layout plus a few of the extra keys found on most computer terminals. Okapi attempts to alleviate this situation by minimising typing, using coloured keys, and being indifferent to case and punctuation.

Input of choices and commands, use of coloured keys

All commands and choices are executed by single keystrokes. Commands are made by the user pressing a coloured key; choices are usually made by pressing one of the numerical keys located along the top row of the keyboard starting with "1". Okapi responds to the single keystroke, since the use of SEND or RETURN is redundant for fixed length input. (For variable length input, i.e. input of search terms, RETURN key is not used as such, but a surrogate painted key, GREEN, see below).

Six of the keys are brightly painted. This makes them easy to find and provides congenial nomenclature. Each of these keys has the same general function in whatever context it is used. These functions are explained on a HELP screen (Fig. 7.7), but the user is not expected to memorise anything. Where possible the layout of instructions about the coloured keys on the screen reflects the relative positions of these keys on the keyboard. They are situated in obvious positions on the circumference of the keyboard. The functions of the coloured keys are intended to be subliminally mnemonic:

GREEN = go (continue, proceed, browse forwards)
BLUE = go back (reenter search terms, browse backwards)
RED = stop (interrupt, enable choice of a different option)
YELLOW = help
WHITE = erase one character (like white paint on white paper)
BLACK = finish session

The first three are used for commands and have less rigid functions than the second three. We considered labelling the three keys which have a (relatively) constant function, but only the BLACK key is labelled ("PRESS WHEN FINISHED").

The use of the BLACK key is quite optional. It serves the tidy user who wishes to leave the screen as s/he found it, and was also provided in an attempt to demarcate user sessions during logging for evaluation purposes (see Chapter 8). Okapi also employs a timing out function: after a period of "silence" from the user s/he is assumed to have finished and the display reverts to its introductory state (Fig. 7.4) ready for the next user. The length of non-interaction which is interpreted as end-of-session by Okapi depends on the current state of the search. It is shorter during search term input than during display of results (1.25 and 3.75

minutes respectively), since users may be expected to pause for longer during the latter than the former.

INPUT OF SEARCH TERMS: "SPECIFIC ITEM" SEARCH

Okapi uses a form-filling mode to prompt the user to input a specific item (see Fig. 7.8). It evolved after a long process of discussion and trial and error. Title comes first because users tend to remember titles better than authors. Effective prompting for author was the most difficult part of the user input design. The screen actually never appears quite like Fig. 7.8 because the dotted line for the author, and the initials prompt, appear only after the user has pressed GREEN to move on from the title.

The user may have a title or an author in mind or both. Okapi asks for the title first by highlighting the label TITLE and leaving the cursor at the start of a dotted line. The messages explain that the title may be entered incompletely or skipped altogether. The help available at this stage is shown in Figs. 7.9 and 7.10.

After the title has been entered, or skipped, Okapi removes the dotted line for the title, alters the GREEN message, highlights the label AUTHOR and leaves the cursor at the start of that dotted line (Fig. 7.11). Authors may be personal or corporate. Personal names are virtually impossible to cope with effectively if entered in a totally unstructured way. Okapi deals with these problems by not displaying the INITIALS label at first and by asking for "Surname ONLY, if a person". (Even the comma in this message is vital to its meaning.) This short message does two things: it tells the user the effective way to enter a personal name, and it reminds the user that the author is not always a person.

Initials are useful, if known, for personal names, but there are no foolproof rules which will always distinguish between corporate and personal names. As a compromise, Okapi only prompts for initials if the user has entered a one or two word name (Fig. 7.12).

The GREEN message changes during the input process (compare Figs. 7.8, 7.11, 7.12).

INPUT OF SEARCH TERMS: SUBJECT SEARCH

In contrast to the specific item option, subject search input is free and unstructured in the Okapi prototype. Users are simply asked to "enter word(s) or a short phrase" describing their subject (Fig. 7.13). Effectiveness is reduced by unnoticed spelling or typing mistakes, or by failure to separate words with spaces (see Chapter 8). The help available at this stage is shown in Figs. 7.14 and 7.15.

7.5.2 Action by Okapi

Once the user input is complete, the action taken by Okapi depends on a number of factors:

the broad type of search (specific or subject)
the number of fields input for specific item
    (author only, title only or both)
the number of words input
the results of searching the index in each case
    (match, partial match, no match)
the number of postings in each case
the user's choice of action

Okapi has to make various decisions:

the appropriate beasts to search for in each case
how to interpret partial matches
when to consult the user
what sort of display is appropriate
when an implied AND might be useful
when a hyper-OR might be useful

It does this by following search decision trees as described in Chapter 6. There are a prohibitive number of branches for an exhaustive description, but the following examples will illustrate the process involved.

SPECIFIC ITEM SEARCH: AUTHOR ONLY

In Fig. 7.16 the user has input the author "Freud S". Okapi searches for this as a complete author. There is an exact match with 26 postings. If the user presses GREEN Okapi will display short records, six per screen.

In Fig. 7.17 the user has input the author "Friendlich". Okapi searches for this as a complete author, and, because it is one word and there was no match, also as an author surname or word. There is no match. The user is given the option of browsing the author index around "Friendlich..." (see Fig. 7.24), which might help if the author was misspelt.

In Fig. 7.18 the user has input the author "Department of health & social security". Okapi searches for this as a complete author. There is no match, so, because it is more than one word, Okapi searches for the individual words, ignoring stop words, and performs an implied AND. This succeeds.

Specific item search: title only

In Fig. 7.19 the user has input the title "A tale of two cities". Okapi searches for this as a title phrase (without the leading article). There is an exact match with two postings. If the user presses GREEN Okapi will display the records in full, one per screen.

In Fig. 7.20 the user has input a one word title "Women". Okapi searches for this as a title phrase. There is no match. The user is offered the choice of browsing the title index around "Women.." (see Fig. 7.25), or asking Okapi to search for "women" as a title word.

Specific item search: title and author

In Fig. 7.21 the user has input the title "Dreams" and the author "Freud". Okapi searches for the combined title/author key "dreafreu". If the user has input the first four letters of the title and author correctly Okapi can give a very quick response. In this case there is no match so Okapi automatically "AND"s "dreams" as a title word with "Freud" as an author word, and finds three postings. If the user presses GREEN Okapi will display the records in full, one per screen.

Subject search

In Fig. 7.22 the user has input the subject "nuclear energy". Okapi searches for the separate words, as subject words, and then automatically performs an "AND" which results in 16 postings. If the user presses GREEN Okapi will display short records, six per screen.

In Fig. 7.23 the user has input the subject "the psychology of motivation at work in industry". Okapi searches for the separate words, as subject words, ignoring stop words, and then automatically performs an "AND". This fails, so, as there are more than two words, Okapi performs the hyper-OR described in Chapter 6. This can be effective in finding relevant material and presenting it earlier to the user than irrelevant material. When the user presses GREEN the records are immediately displayed in full, one per screen. The short display is by-passed since the concept of a set of records in this situation is fairly meaningless. Towards the end records may be completely irrelevant, containing only one of the terms.

7.5.3 Okapi displays

GENERAL COMMENTS

Okapi's displays are designed within the limitations of the standard VDU screen: 24 lines of 80 characters, one font, simple highlighting, etc. The general layout adopted for displaying results splits the screen into three bands. At the top of the screen there is relatively static information about the current state of the search, below this is a window in which the actual results are displayed, and at the bottom of the screen the user is presented with the appropriate choices of action.

There are two levels of record display, short and full. These are discussed below together with index displays and help. Both index entries (Figs. 7.24 and 7.25) and short records (Fig. 7.26) are displayed on single lines, double spaced, and identified by a number from 1 to 6 at the left-hand side. If the user is browsing backwards the lines are displayed and numbered from the bottom of the window to the top. This makes it self evident what the last operation was.

In the record displays personal authors and added names appear in capitals. Added names appear in parentheses but are otherwise treated as authors (see Figs. 7.26 and 7.27). The records are displayed in the same order as they are held in the source file, i.e. in control number order, as mentioned in Section 4.7 (see also Section 9.4.5), except when they are displayed in ranked order after a hyper-OR (see Section 6.5).

INDEX DISPLAY

Where Okapi is not able to satisfy the user, or where the user input is ambiguous, the index may be displayed. Examples of author index display and title index display are shown respectively in Figs. 7.24 and 7.25. They follow, respectively, from earlier examples, namely: the author only search for "Friendlich" (Fig. 7.17), where the index may be useful because there is no match, and the title only search in Fig. 7.20, where "Women" partially matches several titles. These two index displays are headed "author index" and "title index" respectively. As explained in Section 5.7, Okapi does not maintain separate author and title indexes, but by inspecting beasts it is able to display only appropriate index entries in each case. In these two cases the user was asked if s/he would like to see the index. There are also occasions when Okapi goes straight into the index display, for example if the user inputs only an author's surname (e.g. "Freud"), which partially matches more than one author in the index.

The user may browse forwards or backwards through the index, or opt to look at the records for any entry. Okapi will display short records, six per screen, or records in full, depending on the number of postings, and the user can return to the index at any time.

Short record display

Okapi uses the short record display format when there are more than four postings. There is a slight difference between the short display used for subject and specific item searches, see below. The example in Fig. 7.26 results from a subject search for "Libraries". There are 356 postings. The records are displayed in a single line format, six per screen. The fields are identified by column headings: "Author", "Title", etc. (The short display for a specific item search does not include the "Shelved at" column, so has more room for the author and title.) The first added name is displayed if there is no author. Two dots, "..", are used to indicate that names or titles have had to be truncated. If the last word displayed is complete, the dots are preceded by one blank. These niceties are intended to be clear and helpful to the user without the need for any explanation.

Full record display

Okapi uses the full record display format in three circumstances: when there are not more than four postings, after a hyper-OR (since the number of postings is ill-defined), and when the user selects a book from the short display. The example in Fig. 7.27 results from the user selecting record 2 from the previous example (Fig. 7.26). The first appearance of the search term is highlighted in the record. Fields are labelled on the left. Empty fields are not labelled. Local data always appear at the bottom of the window. Having got into the full display the user can choose to see "previous" or "the next" books in the set without returning to the short display.

Help

Okapi provides various levels and degrees of help. All messages are clear and jargon-free to minimise the need for help. The YELLOW key is reserved for requesting help. Sometimes it can be used successively to display layers of help, for example Figs. 7.5, 7.6, 7.7 or Figs. 7.8, 7.9, 7.10, 7.7 or Figs. 7.13, 7.14, 7.15, 7.7.

References

1 **Ramsey H R** and **Atwood M E.** Man-machine interaction design guidance: state of the art. In: *Proceedings of the international conference on cybernetics and society*, Boston, Mass, October 8-10 1980. IEEE, 1980.

2 **Ledgard H**, **Singer A** and **Whiteside J.** *Directions in human factors for interactive systems.* Springer Verlag, 1981. (Lecture notes in computer science, 103.)

3 **Borgman C L.** Psychological research in human-computer interaction. In: *Annual Review of Information Science and Technology 19.* Edited by Martha E Williams. Knowledge Industry Publications, 1984.

4 **Card S K**, **Moran T P** and **Newell A.** *The psychology of human-computer interaction.* Hillsdale, NJ: Lawrence Erlbaum Associates, 1983.

5 **Shackel B.** Dialogues and language, can computer ergonomics help? *Ergonomics 23* (9), 1980, p857-880.

6 **Shneiderman B.** *Software psychology: human factors in computer and information systems.* Little, Brown & Co., 1980.

7 **Ehrenreich S L.** Query languages: design recommendations derived from the human factors literature. *Human Factors 23* (6), 1981, p709-725.

8 **Reisner P.** Human factors studies of database query languages: a survey and assessment. *ACM computing surveys 13* (1), 1981, p13-31.

9 **Galitz W O.** *Handbook of screen format design.* Wellesley, Mass: QED Information Sciences, 1981.

10 **Norman D A** and **Fisher D.** Why alphabetic keyboards are not easy to use: keyboard layout doesn't much matter. *Human Factors 24* (5), 1982, p509-519.

11 **Reynolds L.** *The presentation of bibliographic information on Prestel.* Royal College of Art, 1980. British Library Research and Develop ment Report No 5536.

12 **Bates M J.** Information search tactics. *Journal of the American Society for Information Science 30* (4), 1979, p205-214.

13 **Rouse W B** and **Rouse S H.** Human information seeking and design of information systems. *Information Processing and Management 20* (1-2), 1984, p129-138.

14 **Kaske N K** and **Sanders N P.** *A comprehensive study of online public access catalogs: an overview and application of findings.* Final report to the Council on Library Resources. Vol III. OCLC, 1983.

15 **Moore C W.** User reactions to online catalogs: an exploratory study. *College and Research Libraries 42* (4), 1981, p295-302.

16 **Markey K.** Offline and online user assistance for online catalog searchers. *Online 8* (3), 1984, p54-66.

17 **Hildreth C R.** Pursuing the ideal: generations of online catalogs. In: *Online catalogs, online reference - converging trends.* Proceedings of a LITA Preconference, June 23-24 1983, Los Angeles. American Library Association, 1984, p31-56.

18 **Tagliacozzo R** and **Kochen M.** Information-seeking behavior of catalog users. *Information Storage and Retrieval 6* (5), 1970, p363-381.

19 **Mitev N N** and **Walker S.** Information retrieval aids in an online public access catalogue: automatic intelligent search sequencing. In: *Informatics 8: Advances in intelligent retrieval.* Proceedings of an Aslib/BCS conference. Oxford, 16-17 April 1985. To be published 1985.

20 **Akeroyd J** (Polytechnic of the South Bank). Verbal communication, 1983.

21 **Cochrane P A** and **Settel B.** Augmenting subject descriptions for books in online catalogs. *Database 5* (4), 1982, p29-37.

22 **Hildreth C R.** To boolean or not to boolean. *Information Technology and Libraries 2* (3), 1983, p235-237.

23 **Markey K.** Subject-searching experiences and needs of online catalog users: implications for library classification. *Library Resources and Technical Services*, January-March 1985, p34-51.

24 **Hildreth C R.** Online browsing support capabilities. In: *Information interaction.* Proceedings of the 45th ASIS annual meeting, Columbus, Ohio, 17-21 October 1982. Knowledge Industry Publications, 1982.

25 **Mitev N N.** The user interface in an online public access catalogue. In: *Computer assisted information retrieval.* RIAO 85, International

symposium organised by the Centre des Hautes Etudes d'Informatique Documentaire, 18-20 March 1985, Grenoble. Paris: CNRS, 1985.

26 **Cherry S S.** The moving finger "accesses". *American Libraries 12* (1), 1981, p14-16.

27 **Lipow A G.** Practical considerations of the current capabilities of subject access in online public catalogs. *Library Resources and Technical Services 27* (1), 1983, p81-87.

28 **Markey K.** *Subject searching in library catalogs: before and after the introduction of online catalogs.* OCLC, 1984.

29 **Matthews J R.** *Public access to online catalogs: a planning guide for managers.* Online Inc, 1982.

30 **Larson R R** and **Graham V.** Monitoring and evaluating MELVYL. *Information Technology and Libraries 2* (1), 1983, p93-104.

31 University of California prototype online catalog. *Preliminary specifications for the patron interface.* [Prepared by] Katharina Klemperer and Mike Berger. University of California, Berkeley, July 1980.

32 [Lipow A] in: *Training users of online public access catalogs.* Report by Marsha Hamilton McClintock of a conference sponsored by Trinity University and the Council for Library Resources, San Antonio, Texas, January 12-14 1983. CLR, July 1983.

33 **Arret L.** Can online catalogs be too easy? User-easy is not user-friendly if progressive learning and system mastery are sacrificed. *American Libraries*, February 1985, p118-120.

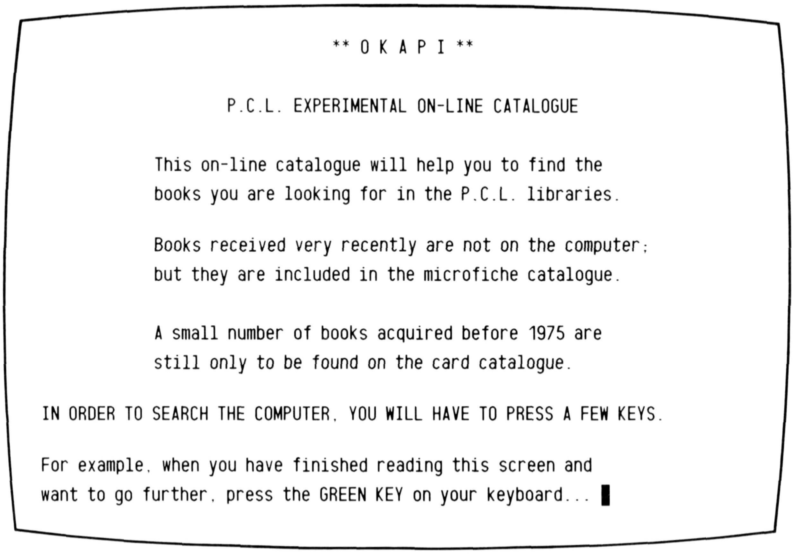
34 **Shneiderman B.** Human factors experiments in designing interactive systems. *Computer (IEEE) 12*, 1979, p9-19.

35 **Hildreth C R.** *Online public access catalogs: the user interface.* OCLC, 1982.

36 **Manson P** (Polytechnic of Central London). Verbal communication, April 1985.

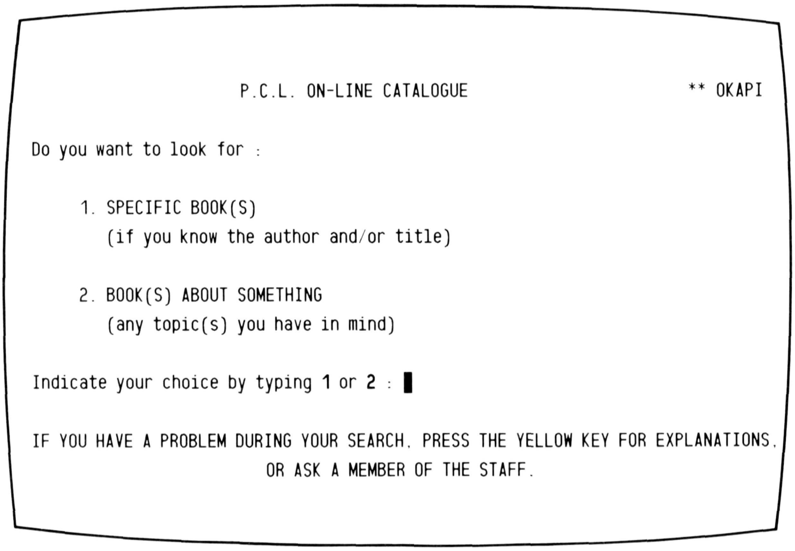
37 **Walker S.** The free language approach to online catalogues. In: *Keyword catalogues and the free language approach.* Edited by Philip Bryant. Centre for Catalogue Research. Bath University Library. 1985.

38 [Walker S]. *Improving access in an online public access catalogue by automatic word stemming, spelling correction, and limited synonym generation.* Research proposal submitted to the British Library Research and Development Department. April 1985.



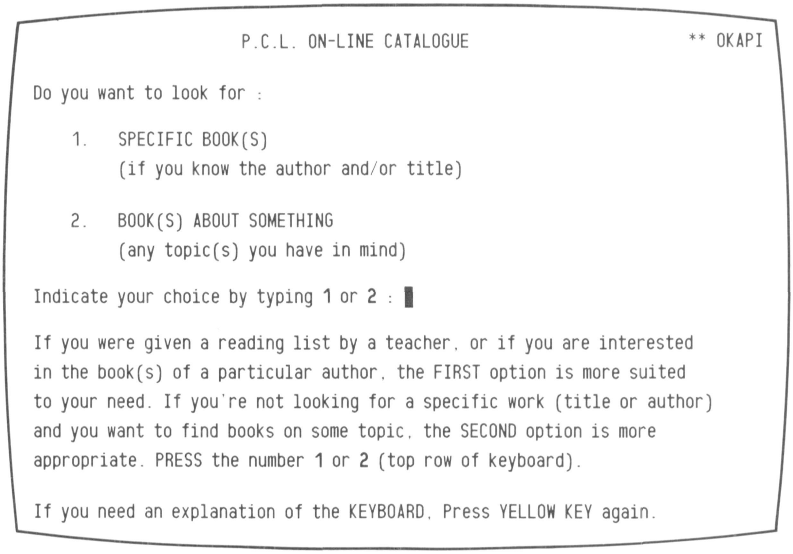
```
                         ** O K A P I **

              P.C.L. EXPERIMENTAL ON-LINE CATALOGUE

          This on-line catalogue will help you to find the
          books you are looking for in the P.C.L. libraries.

          Books received very recently are not on the computer;
          but they are included in the microfiche catalogue.

          A small number of books acquired before 1975 are
          still only to be found on the card catalogue.

IN ORDER TO SEARCH THE COMPUTER, YOU WILL HAVE TO PRESS A FEW KEYS.

For example, when you have finished reading this screen and
want to go further, press the GREEN KEY on your keyboard... ▌
```

Figure 7.4. Okapi introductory screen



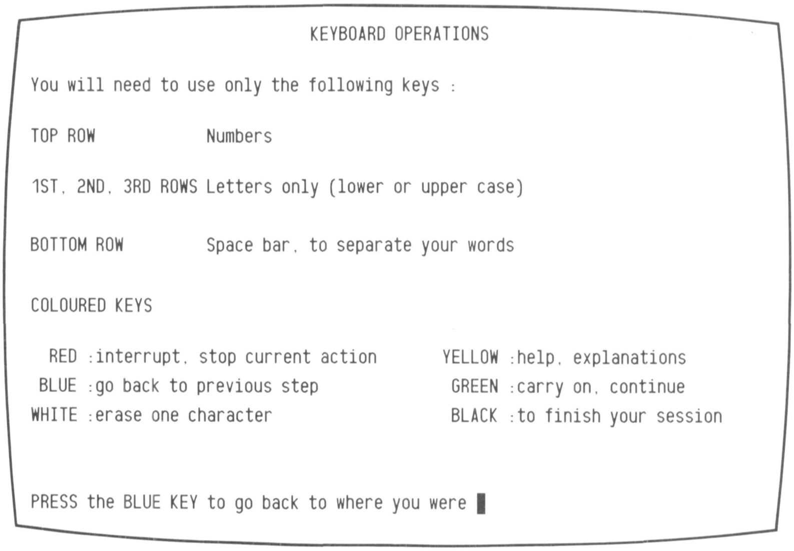
```
                     P.C.L. ON-LINE CATALOGUE                 ** OKAPI

Do you want to look for :

     1. SPECIFIC BOOK(S)
        (if you know the author and/or title)

     2. BOOK(S) ABOUT SOMETHING
        (any topic(s) you have in mind)

Indicate your choice by typing 1 or 2 : ▌

IF YOU HAVE A PROBLEM DURING YOUR SEARCH, PRESS THE YELLOW KEY FOR EXPLANATIONS,
                     OR ASK A MEMBER OF THE STAFF.
```

Figure 7.5. Okapi first menu



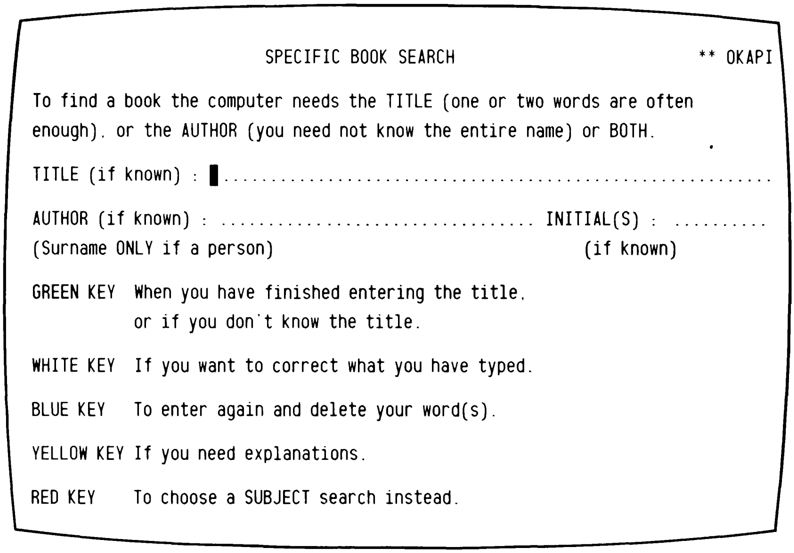
```
                    P.C.L. ON-LINE CATALOGUE                        ** OKAPI

Do you want to look for :

    1.   SPECIFIC BOOK(S)
         (if you know the author and/or title)

    2.   BOOK(S) ABOUT SOMETHING
         (any topic(s) you have in mind)

Indicate your choice by typing 1 or 2 : ▌

If you were given a reading list by a teacher, or if you are interested
in the book(s) of a particular author, the FIRST option is more suited
to your need. If you're not looking for a specific work (title or author)
and you want to find books on some topic, the SECOND option is more
appropriate. PRESS the number 1 or 2 (top row of keyboard).

If you need an explanation of the KEYBOARD, Press YELLOW KEY again.
```

Figure 7.6. Okapi first menu with help



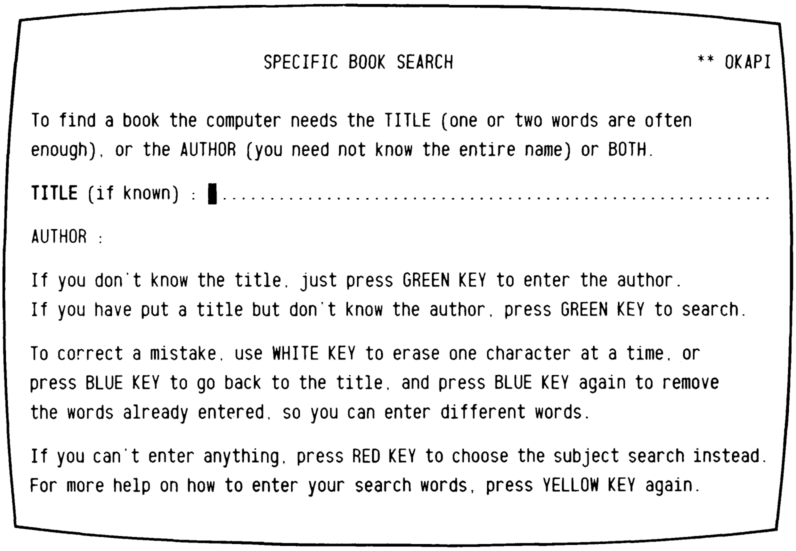
```
                          KEYBOARD OPERATIONS

You will need to use only the following keys :

TOP ROW             Numbers

1ST, 2ND, 3RD ROWS Letters only (lower or upper case)

BOTTOM ROW          Space bar, to separate your words

COLOURED KEYS

  RED :interrupt, stop current action       YELLOW :help, explanations
 BLUE :go back to previous step              GREEN :carry on, continue
WHITE :erase one character                   BLACK :to finish your session


PRESS the BLUE KEY to go back to where you were ▌
```

Figure 7.7. Okapi keyboard help



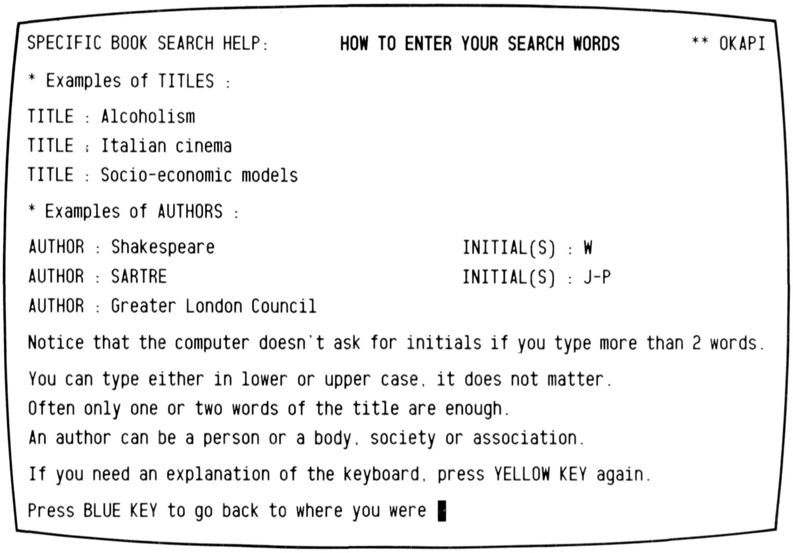
```
                      SPECIFIC BOOK SEARCH                       ** OKAPI

To find a book the computer needs the TITLE (one or two words are often
enough), or the AUTHOR (you need not know the entire name) or BOTH.

TITLE (if known) : █...................................................

AUTHOR (if known) : ................................ INITIAL(S) : ..........
(Surname ONLY if a person)                            (if known)

GREEN KEY  When you have finished entering the title,
           or if you don't know the title.

WHITE KEY  If you want to correct what you have typed.

BLUE KEY   To enter again and delete your word(s).

YELLOW KEY If you need explanations.

RED KEY    To choose a SUBJECT search instead.
```

Figure 7.8. Okapi overall view of specific item input



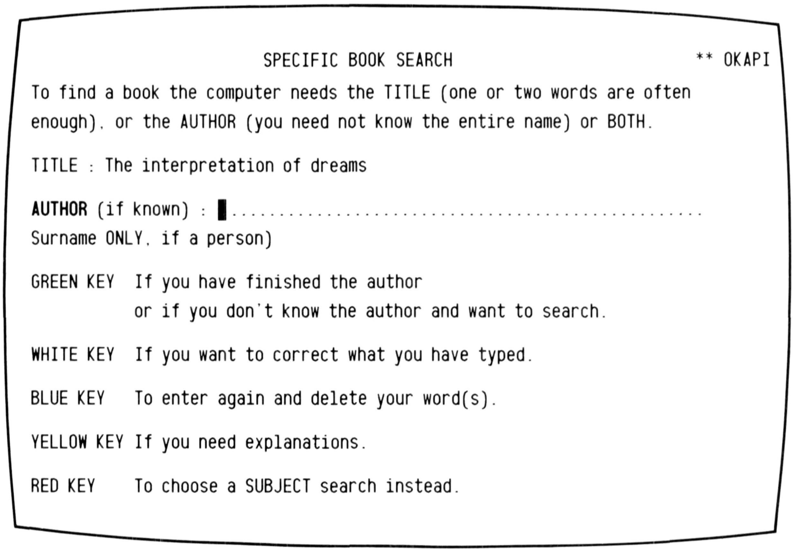
```
                      SPECIFIC BOOK SEARCH                       ** OKAPI

To find a book the computer needs the TITLE (one or two words are often
enough), or the AUTHOR (you need not know the entire name) or BOTH.

TITLE (if known) : █...................................................

AUTHOR :

If you don't know the title, just press GREEN KEY to enter the author.
If you have put a title but don't know the author, press GREEN KEY to search.

To correct a mistake, use WHITE KEY to erase one character at a time, or
press BLUE KEY to go back to the title, and press BLUE KEY again to remove
the words already entered, so you can enter different words.

If you can't enter anything, press RED KEY to choose the subject search instead.
For more help on how to enter your search words, press YELLOW KEY again.
```

Figure 7.9. Okapi specific book help (1)



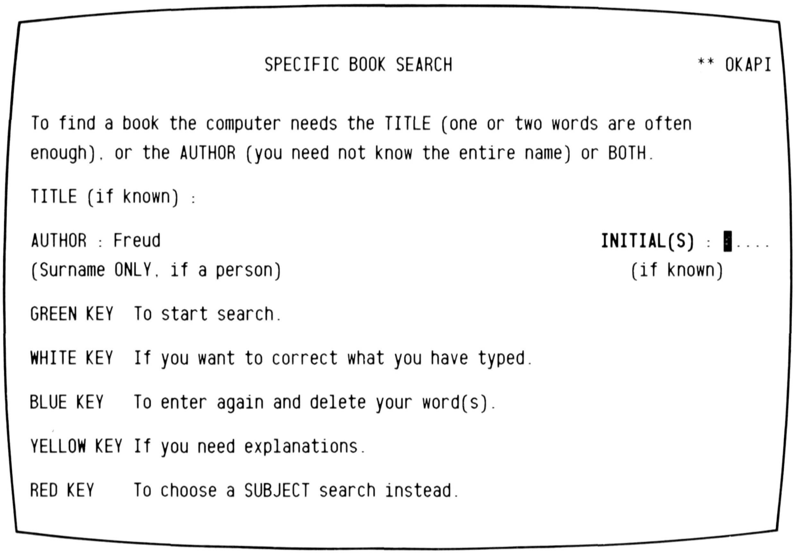
```
SPECIFIC BOOK SEARCH HELP:       HOW TO ENTER YOUR SEARCH WORDS       ** OKAPI
* Examples of TITLES :
TITLE : Alcoholism
TITLE : Italian cinema
TITLE : Socio-economic models
* Examples of AUTHORS :
AUTHOR : Shakespeare                           INITIAL(S) : W
AUTHOR : SARTRE                                INITIAL(S) : J-P
AUTHOR : Greater London Council
Notice that the computer doesn't ask for initials if you type more than 2 words.
You can type either in lower or upper case, it does not matter.
Often only one or two words of the title are enough.
An author can be a person or a body, society or association.
If you need an explanation of the keyboard, press YELLOW KEY again.
Press BLUE KEY to go back to where you were █
```

Figure 7.10. Okapi specific book help (2)



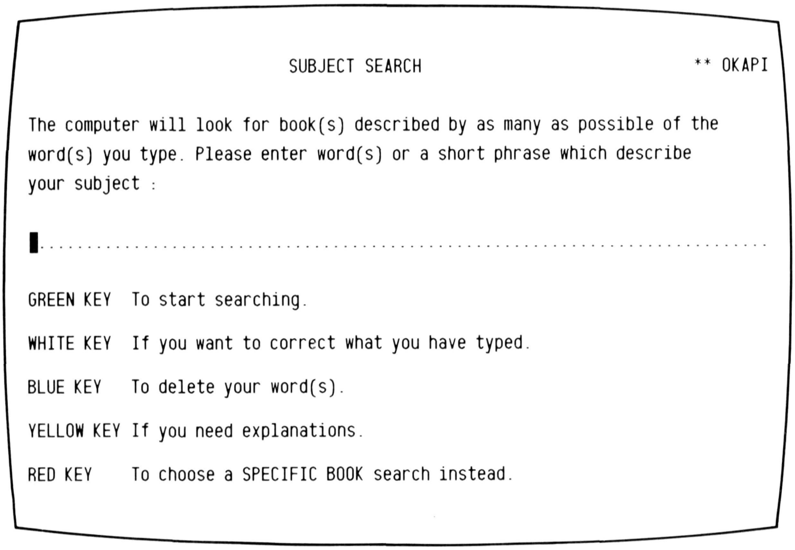
```
                         SPECIFIC BOOK SEARCH                          ** OKAPI
To find a book the computer needs the TITLE (one or two words are often
enough), or the AUTHOR (you need not know the entire name) or BOTH.
TITLE : The interpretation of dreams
AUTHOR (if known) : █..................................................
Surname ONLY, if a person)
GREEN KEY  If you have finished the author
           or if you don't know the author and want to search.
WHITE KEY  If you want to correct what you have typed.
BLUE KEY   To enter again and delete your word(s).
YELLOW KEY If you need explanations.
RED KEY    To choose a SUBJECT search instead.
```

Figure 7.11. Okapi title input



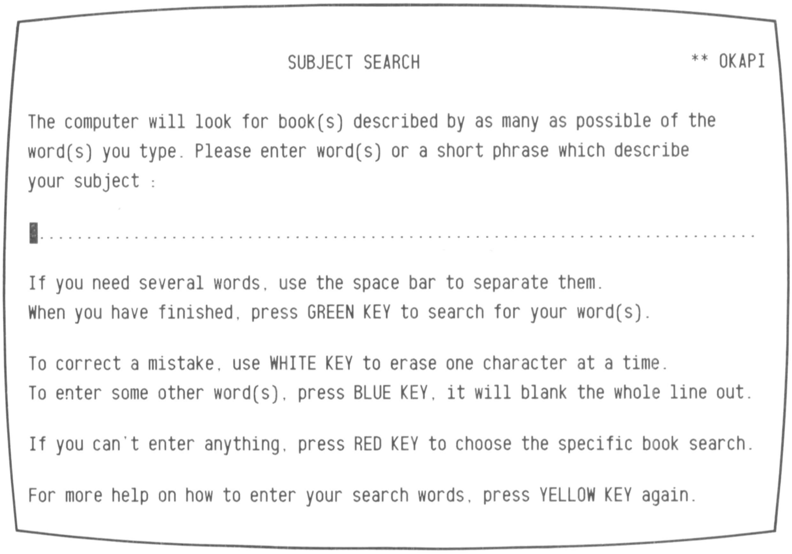
```
                         SPECIFIC BOOK SEARCH                     ** OKAPI

To find a book the computer needs the TITLE (one or two words are often
enough). or the AUTHOR (you need not know the entire name) or BOTH.

TITLE (if known) :

AUTHOR : Freud                                          INITIAL(S) : █....
(Surname ONLY, if a person)                               (if known)

GREEN KEY  To start search.

WHITE KEY  If you want to correct what you have typed.

BLUE KEY   To enter again and delete your word(s).

YELLOW KEY If you need explanations.

RED KEY    To choose a SUBJECT search instead.
```

Figure 7.12. Okapi author input



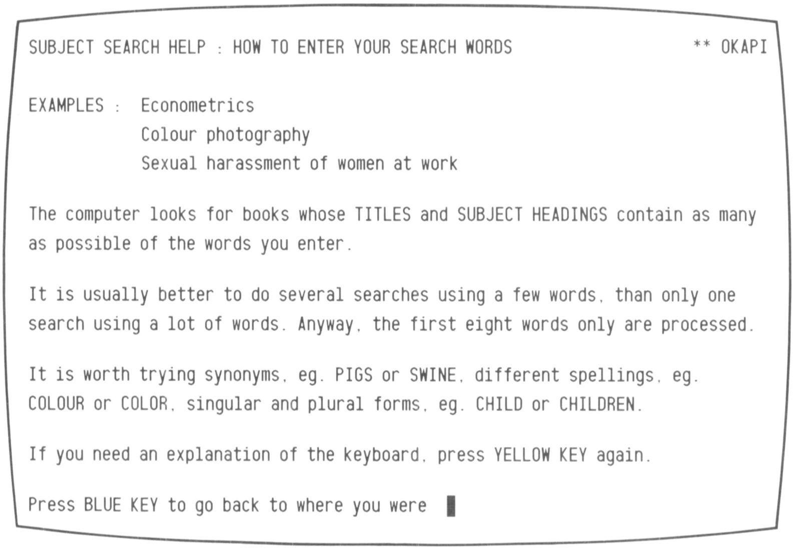
```
                            SUBJECT SEARCH                        ** OKAPI

The computer will look for book(s) described by as many as possible of the
word(s) you type. Please enter word(s) or a short phrase which describe
your subject :

█.........................................................................

GREEN KEY  To start searching.

WHITE KEY  If you want to correct what you have typed.

BLUE KEY   To delete your word(s).

YELLOW KEY If you need explanations.

RED KEY    To choose a SPECIFIC BOOK search instead.
```

Figure 7.13. Okapi subject search input



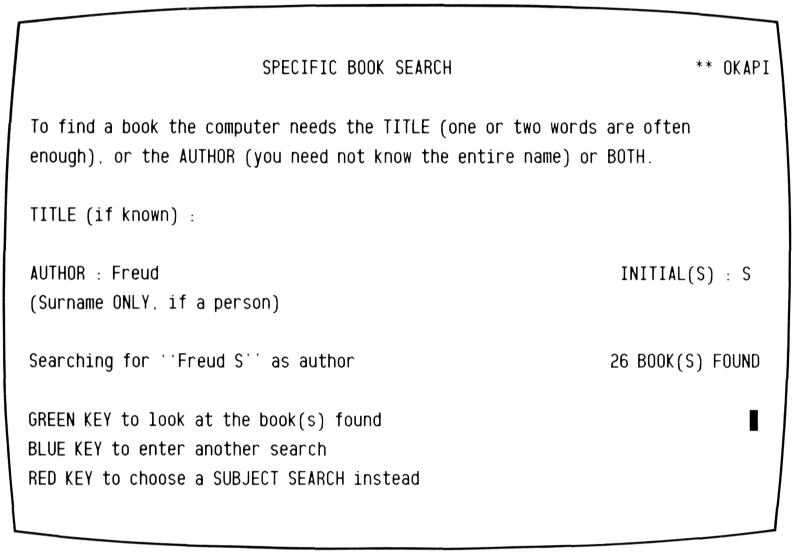
```
                         SUBJECT SEARCH                          ** OKAPI

The computer will look for book(s) described by as many as possible of the
word(s) you type. Please enter word(s) or a short phrase which describe
your subject :

█..............................................................................

If you need several words, use the space bar to separate them.
When you have finished, press GREEN KEY to search for your word(s).

To correct a mistake, use WHITE KEY to erase one character at a time.
To enter some other word(s), press BLUE KEY, it will blank the whole line out.

If you can't enter anything, press RED KEY to choose the specific book search.

For more help on how to enter your search words, press YELLOW KEY again.
```

Figure 7.14. Okapi subject search help (1)



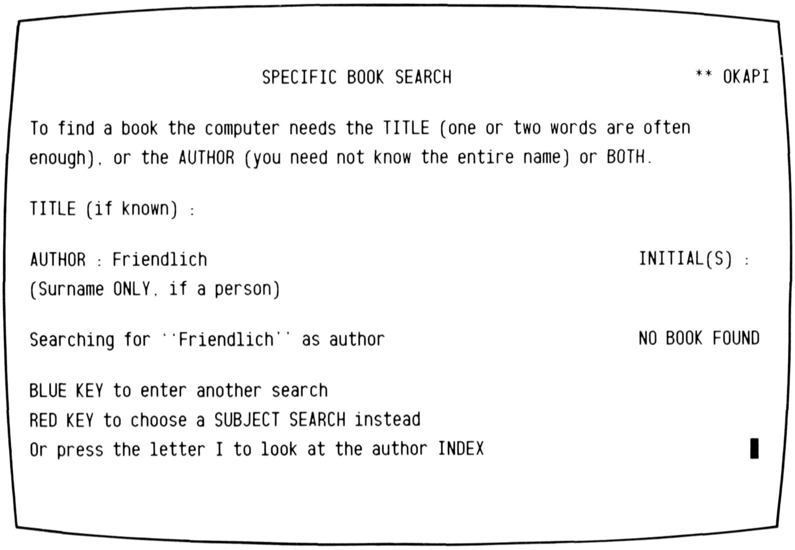
```
SUBJECT SEARCH HELP : HOW TO ENTER YOUR SEARCH WORDS             ** OKAPI

EXAMPLES :  Econometrics
            Colour photography
            Sexual harassment of women at work

The computer looks for books whose TITLES and SUBJECT HEADINGS contain as many
as possible of the words you enter.

It is usually better to do several searches using a few words, than only one
search using a lot of words. Anyway, the first eight words only are processed.

It is worth trying synonyms, eg. PIGS or SWINE, different spellings, eg.
COLOUR or COLOR, singular and plural forms, eg. CHILD or CHILDREN.

If you need an explanation of the keyboard, press YELLOW KEY again.

Press BLUE KEY to go back to where you were  █
```

Figure 7.15. Okapi subject search help (2)



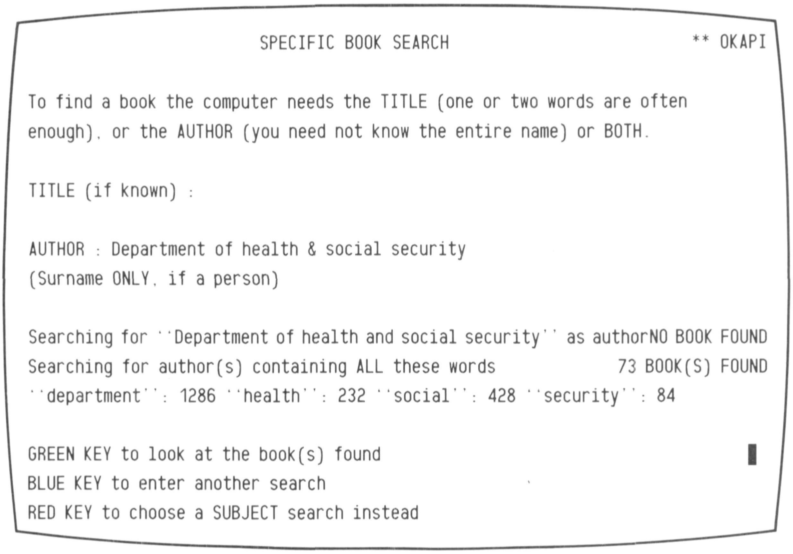
```
                         SPECIFIC BOOK SEARCH                         ** OKAPI

To find a book the computer needs the TITLE (one or two words are often
enough), or the AUTHOR (you need not know the entire name) or BOTH.

TITLE (if known) :

AUTHOR : Freud                                               INITIAL(S) : S
(Surname ONLY, if a person)

Searching for ``Freud S'' as author                         26 BOOK(S) FOUND

GREEN KEY to look at the book(s) found                                     █
BLUE KEY to enter another search
RED KEY to choose a SUBJECT SEARCH instead
```

Figure 7.16. Okapi search: author only (match)



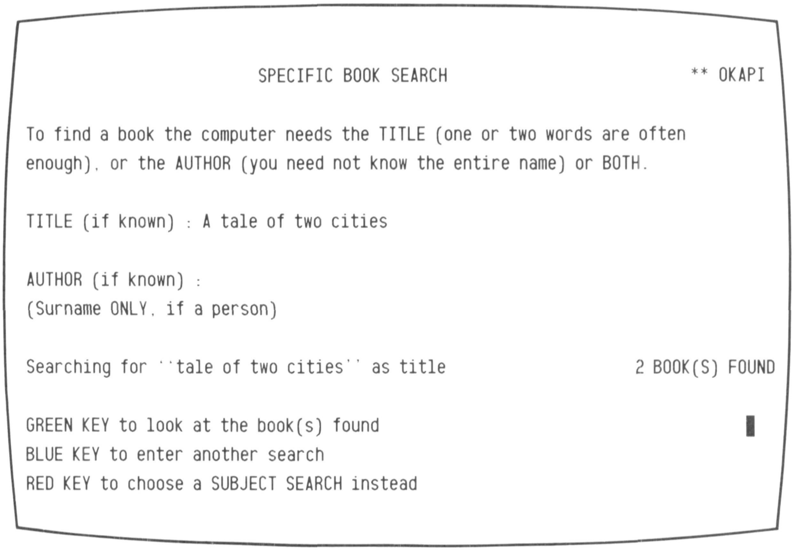
```
                         SPECIFIC BOOK SEARCH                         ** OKAPI

To find a book the computer needs the TITLE (one or two words are often
enough), or the AUTHOR (you need not know the entire name) or BOTH.

TITLE (if known) :

AUTHOR : Friendlich                                            INITIAL(S) :
(Surname ONLY, if a person)

Searching for ``Friendlich'' as author                         NO BOOK FOUND

BLUE KEY to enter another search
RED KEY to choose a SUBJECT SEARCH instead
Or press the letter I to look at the author INDEX                          █
```

Figure 7.17. Okapi search: author only (no match)



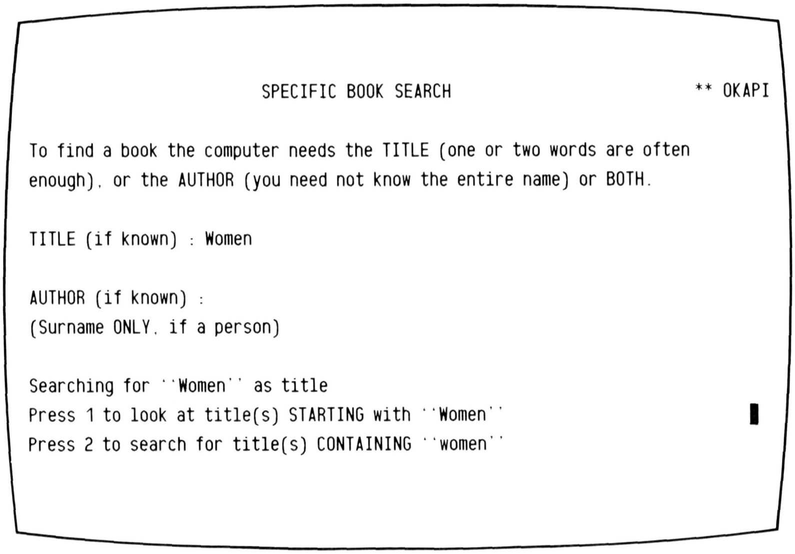
```
                         SPECIFIC BOOK SEARCH                        ** OKAPI

To find a book the computer needs the TITLE (one or two words are often
enough), or the AUTHOR (you need not know the entire name) or BOTH.

TITLE (if known) :

AUTHOR : Department of health & social security
(Surname ONLY, if a person)

Searching for ``Department of health and social security'' as authorNO BOOK FOUND
Searching for author(s) containing ALL these words            73 BOOK(S) FOUND
``department'': 1286 ``health'': 232 ``social'': 428 ``security'': 84

GREEN KEY to look at the book(s) found                                       ▮
BLUE KEY to enter another search
RED KEY to choose a SUBJECT search instead
```

Figure 7.18. Okapi search: author only (AND)



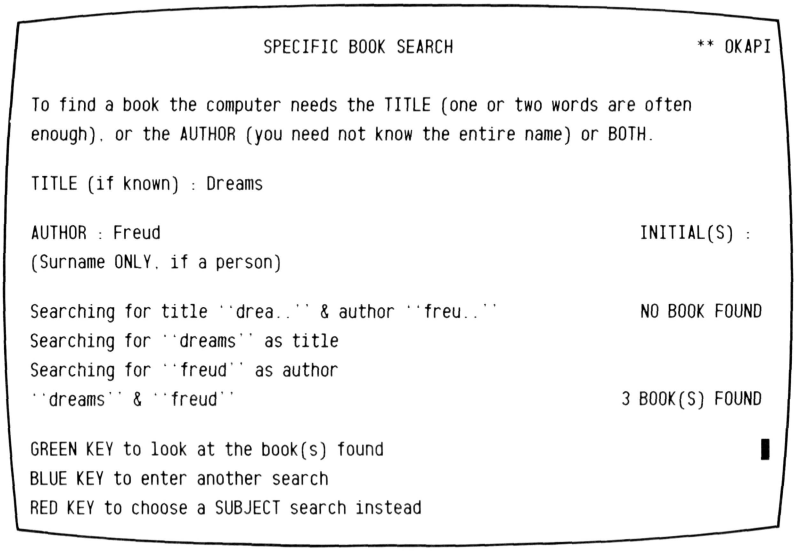
```
                         SPECIFIC BOOK SEARCH                        ** OKAPI

To find a book the computer needs the TITLE (one or two words are often
enough), or the AUTHOR (you need not know the entire name) or BOTH.

TITLE (if known) : A tale of two cities

AUTHOR (if known) :
(Surname ONLY, if a person)

Searching for ``tale of two cities'' as title                  2 BOOK(S) FOUND

GREEN KEY to look at the book(s) found                                       ▮
BLUE KEY to enter another search
RED KEY to choose a SUBJECT SEARCH instead
```

Figure 7.19. Okapi search: title only (match)



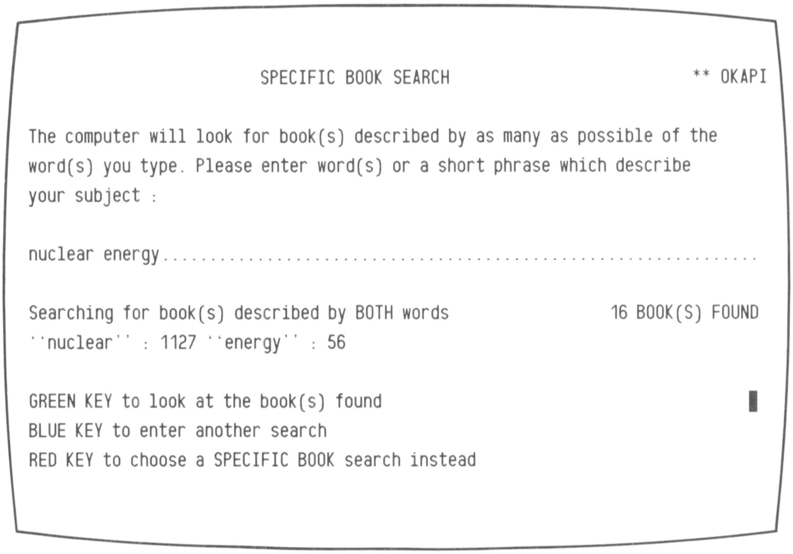
```
                         SPECIFIC BOOK SEARCH                    ** OKAPI

To find a book the computer needs the TITLE (one or two words are often
enough), or the AUTHOR (you need not know the entire name) or BOTH.

TITLE (if known) : Women


AUTHOR (if known) :
(Surname ONLY, if a person)


Searching for ``Women'' as title
Press 1 to look at title(s) STARTING with ``Women''
Press 2 to search for title(s) CONTAINING ``women''
```

Figure 7.20. Okapi search: title only (partial match)



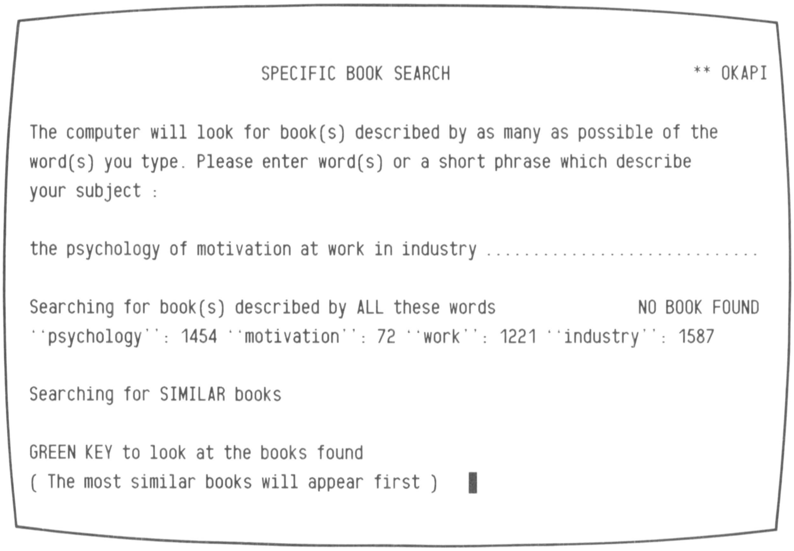
```
                         SPECIFIC BOOK SEARCH                    ** OKAPI

To find a book the computer needs the TITLE (one or two words are often
enough), or the AUTHOR (you need not know the entire name) or BOTH.

TITLE (if known) : Dreams

AUTHOR : Freud                                          INITIAL(S) :
(Surname ONLY, if a person)

Searching for title ``drea..'' & author ``freu..''      NO BOOK FOUND
Searching for ``dreams'' as title
Searching for ``freud'' as author
``dreams'' & ``freud''                                 3 BOOK(S) FOUND

GREEN KEY to look at the book(s) found
BLUE KEY to enter another search
RED KEY to choose a SUBJECT search instead
```

Figure 7.21. Okapi search: title and author



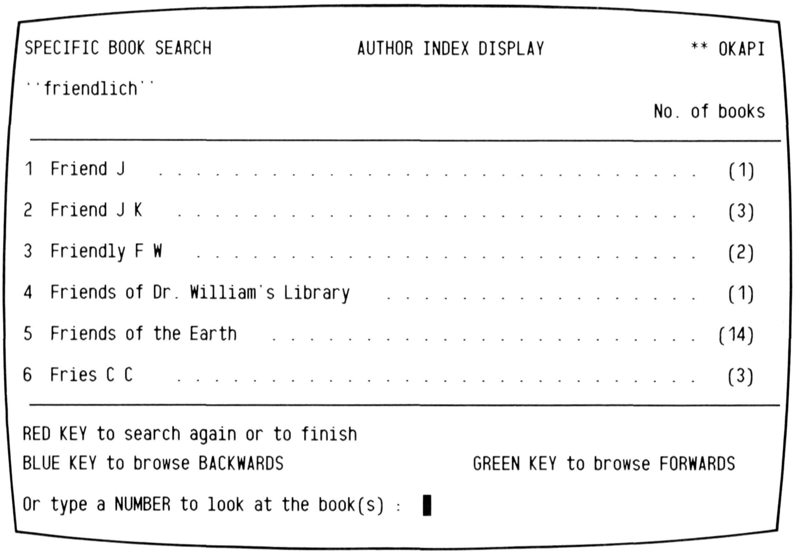
```
                         SPECIFIC BOOK SEARCH                        ** OKAPI

The computer will look for book(s) described by as many as possible of the
word(s) you type. Please enter word(s) or a short phrase which describe
your subject :

nuclear energy..................................................................

Searching for book(s) described by BOTH words                  16 BOOK(S) FOUND
``nuclear'' : 1127 ``energy'' : 56

GREEN KEY to look at the book(s) found
BLUE KEY to enter another search
RED KEY to choose a SPECIFIC BOOK search instead
```

Figure 7.22. Okapi search: subject (AND)



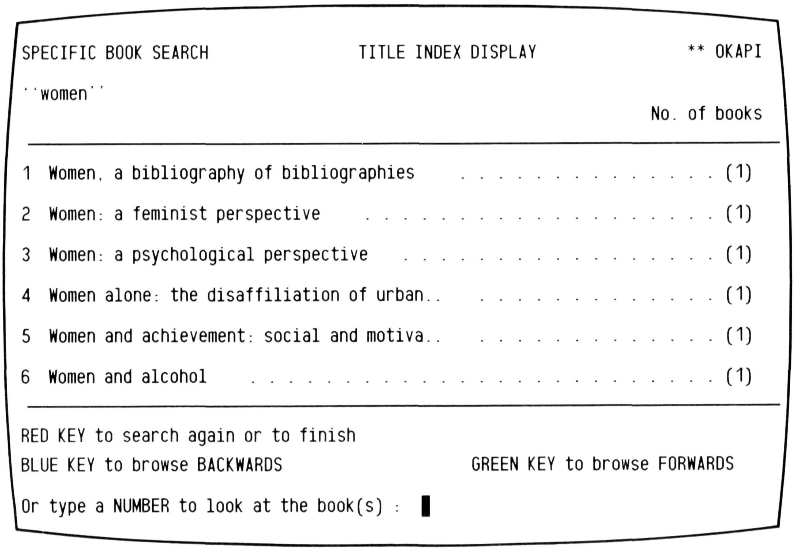
```
                         SPECIFIC BOOK SEARCH                        ** OKAPI

The computer will look for book(s) described by as many as possible of the
word(s) you type. Please enter word(s) or a short phrase which describe
your subject :

the psychology of motivation at work in industry .............................

Searching for book(s) described by ALL these words               NO BOOK FOUND
``psychology'': 1454 ``motivation'': 72 ``work'': 1221 ``industry'': 1587

Searching for SIMILAR books

GREEN KEY to look at the books found
( The most similar books will appear first )
```

Figure 7.23. Okapi search: subject (hyper-OR)



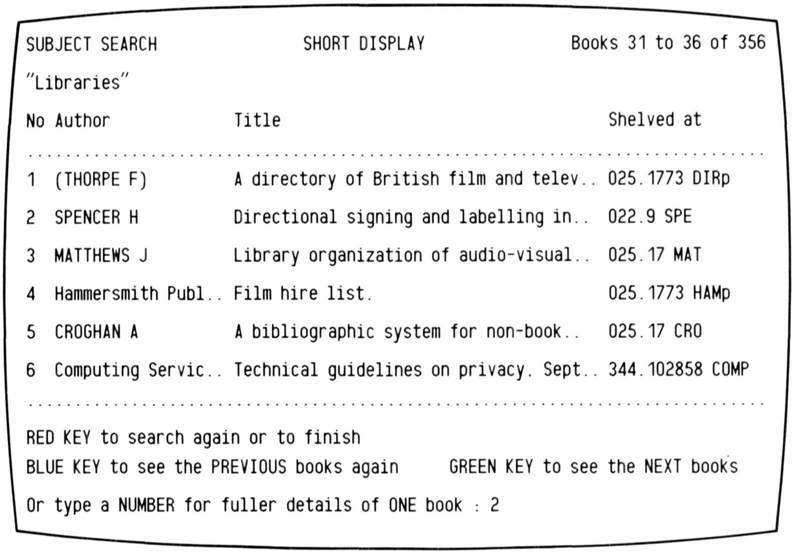
```
SPECIFIC BOOK SEARCH              AUTHOR INDEX DISPLAY              ** OKAPI
``friendlich''
                                                               No. of books
---------------------------------------------------------------------------
1  Friend J  . . . . . . . . . . . . . . . . . . . . . . . . . . . . . (1)
2  Friend J K  . . . . . . . . . . . . . . . . . . . . . . . . . . . . (3)
3  Friendly F W  . . . . . . . . . . . . . . . . . . . . . . . . . . . (2)
4  Friends of Dr. William's Library  . . . . . . . . . . . . . . . . . (1)
5  Friends of the Earth  . . . . . . . . . . . . . . . . . . . . . . . (14)
6  Fries C C  . . . . . . . . . . . . . . . . . . . . . . . . . . . . . (3)
---------------------------------------------------------------------------
RED KEY to search again or to finish
BLUE KEY to browse BACKWARDS                 GREEN KEY to browse FORWARDS
Or type a NUMBER to look at the book(s) : █
```

Figure 7.24. Okapi author index display



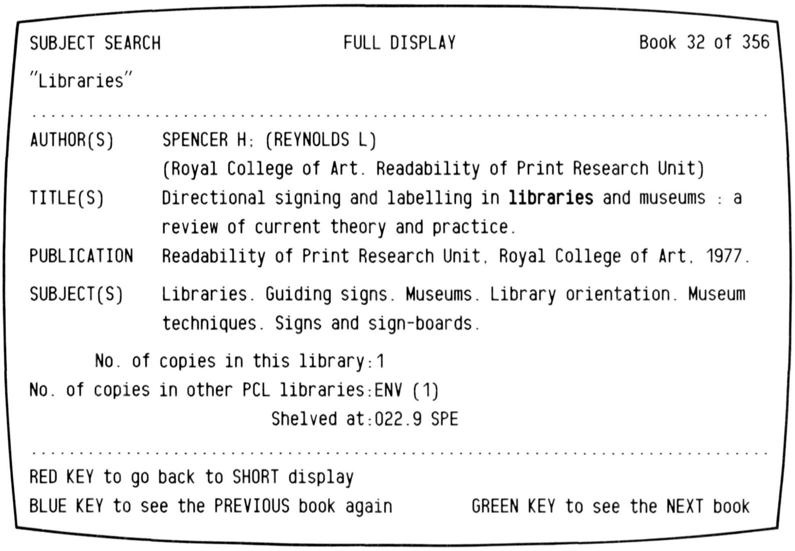
```
SPECIFIC BOOK SEARCH              TITLE INDEX DISPLAY               ** OKAPI
``women''
                                                               No. of books
---------------------------------------------------------------------------
1  Women, a bibliography of bibliographies  . . . . . . . . . . . . . (1)
2  Women: a feminist perspective  . . . . . . . . . . . . . . . . . . (1)
3  Women: a psychological perspective  . . . . . . . . . . . . . . . (1)
4  Women alone: the disaffiliation of urban..  . . . . . . . . . . . (1)
5  Women and achievement: social and motiva..  . . . . . . . . . . . (1)
6  Women and alcohol  . . . . . . . . . . . . . . . . . . . . . . . . (1)
---------------------------------------------------------------------------
RED KEY to search again or to finish
BLUE KEY to browse BACKWARDS                 GREEN KEY to browse FORWARDS
Or type a NUMBER to look at the book(s) : █
```

Figure 7.25. Okapi title index display



```
SUBJECT SEARCH                SHORT DISPLAY              Books 31 to 36 of 356
"Libraries"
No Author               Title                                 Shelved at
..............................................................................
1  (THORPE F)           A directory of British film and telev.. 025.1773 DIRp
2  SPENCER H            Directional signing and labelling in..  022.9 SPE
3  MATTHEWS J           Library organization of audio-visual..  025.17 MAT
4  Hammersmith Publ..   Film hire list.                         025.1773 HAMp
5  CROGHAN A            A bibliographic system for non-book..   025.17 CRO
6  Computing Servic..   Technical guidelines on privacy. Sept.. 344.102858 COMP
..............................................................................
RED KEY to search again or to finish
BLUE KEY to see the PREVIOUS books again     GREEN KEY to see the NEXT books
Or type a NUMBER for fuller details of ONE book : 2
```

Figure 7.26. Okapi brief record display



```
SUBJECT SEARCH                     FULL DISPLAY                  Book 32 of 356
"Libraries"
..............................................................................
AUTHOR(S)      SPENCER H; (REYNOLDS L)
               (Royal College of Art. Readability of Print Research Unit)
TITLE(S)       Directional signing and labelling in libraries and museums : a
               review of current theory and practice.
PUBLICATION    Readability of Print Research Unit, Royal College of Art, 1977.
SUBJECT(S)     Libraries. Guiding signs. Museums. Library orientation. Museum
               techniques. Signs and sign-boards.
       No. of copies in this library:1
No. of copies in other PCL libraries:ENV (1)
                          Shelved at:022.9 SPE
..............................................................................
RED KEY to go back to SHORT display
BLUE KEY to see the PREVIOUS book again        GREEN KEY to see the NEXT book
```

Figure 7.27. Okapi full record display