OKAPI: Evaluating and Enhancing an Experimental Online Catalog

STEPHEN WALKER

THIS ARTICLE ORIGINATES from work carried out on the design and evaluation of experimental online catalogs at the Polytechnic of Central London (PCL). The research and development has been funded by the British Library Research and Development Department (BLR&DD) and the Department of Trade and Industry (DTI).

The initial phases of the project—investigation of relevant work, design and development of a prototype catalog, trial single-terminal installation at PCL, live evaluation, and publication of a substantial report[1]—were carried out by a team consisting of Gillian Venner, Nathalie Mitev, and myself. These phases occurred from November 1982 to May 1985. The prototype catalog was named OKAPI (Online Keyword Access to Public Information).

After a hiatus of some months, further funding was granted by BLR&DD (starting in July 1985) to investigate various methods of improving recall or rather ways of reducing the considerable proportion of OPAC searches which fail for various reasons. This is referred to as the "fuzzy matching" project. Further funding has now been granted for a concurrent project on the use of relevance feedback during catalog searching.

To understand some of what follows it is necessary to have some idea of what the Mark 1 OKAPI looks like and how it behaves. There is a fairly full description in *Designing an Online Public Access Catalogue* by Mitev et al.[2]

Stephen Walker is Research Fellow, Online Catalogues, Polytechnic of Central London, London, England.

Environment

At the physical level, OKAPI operates on a local area network. The user stations are Apple IIe microcomputers equipped with Z80 cards. The stations are joined in a Nestar PLAN 4000 local area network, using network interface cards designed and supplied by Nestar. The network contains one file server station which is a 68000-based computer controlling a 140 megabyte Winchester disk drive with connection by coaxial cable. Data transmission is in variable-sized packets at 2.5 million bits per second, and the network protocol is described as a virtual ring. This means that devices can be connected and disconnected arbitrarily, provided there are no closed loops. If a station is connected or disconnected or fails for any reason, the network reconfigures itself and carries on without any noticeable complaint.

The User Stations

Apple IIe's were chosen because Nestar only provided network interface cards for Apples and for IBM PCs, and the IBMs were too expensive. In some ways the Apples are very satisfactory. They have proven extremely robust and reliable, and, apart from the auto-repeat function (which should never be provided on catalog terminals), the keyboard is very satisfactory. Most computer terminals are quite unsuitable for people who are inexperienced with keyboards, computers, numeric pads, and obscurely labeled function keys, and generally have too many keys. Watch a new keyboard user trying to pick out letters, make a space, or try to correct something. The Apple IIe has a few of these superfluous keys—such as ESCAPE, TAB, and some "arrow" keys—but these were painted red, yellow, blue, green, white, and black, and used as "function" keys.

Files and Storage

OKAPI uses a bibliographic file (source file) derived from the United Kingdom MARC format but reduced to nine fields. The source file is generated from MARC exchange tapes of PCL's union monograph catalog (some 100,000 bibliographic records). The nine fields contain:

—"main" author (i.e., MARC 1XX)
—main title (with subtitle and parallel title)
—publication data
—series and part titles

—"added" names (MARC 7XX)
—Dewey class number(s)
—subject headings (LC and/or PRECIS)
—control number
—accession number(s)

The length of the OKAPI records varies between about 150 and 800 characters, and the mean is under 250 characters. This reduction (compared with MARC's 800-1000 character mean) is partly due to:

1. personal names being held as surname and initials only,
2. no statements of responsibility,
3. no physical description, and
4. no notes (except contents notes for analyticals).

The choice of record content was influenced by the Centre for Catalogue Research Report[3] which demonstrated that almost all catalog requirements were satisfied by a record that is very short indeed compared with a conventional entry.

The source file is held on the network file server's disk drive together with extensive indexing. There is no facility for amending or adding catalog records so OKAPI is entirely dependent on the normal PCL cataloging. As in many other United Kingdom libraries, there was a good deal of retrospective conversion at the time when PCL went over to microfiche. The records do not conform to a single consistent standard. Many have no subject headings, some have Library of Congress subject headings, some have PRECIS headings, and some have both.

OKAPI is almost entirely dependent on what is in the source file. There are no authority files and no cross references. (In United Kingdom MARC there are 9XX fields which can be used for *see* and *see also* references. At least one United Kingdom OPAC—the Cambridge University in-house system—has made good use of these fields.)

Access Points

The indexing provides for access by:

1. author or added name by "phrase"—(i.e., surname plus initials or corporate name;
2. surnames and individual words of corporate names;
3. title phrases (including series and part titles);
4. title and subtitle words;
5. 4/4 title/author keys from main title and all names (the user does not have to know how to construct these—they are generated automatically from users' input);

6. subject heading phrases;
7. subject heading words;
8. Dewey numbers; and
9. date of publication index (although this has never been used).

Software

Unlike almost all other online catalogs, each user station has its own copy of the search program and the top level indexes. All processing except access to the central disk store is carried out locally and is entirely independent of the system as a whole. Fixed data for screen displays are downloaded from the file server as required. This means that far more attention can be given to fine details of user interaction than is possible with systems sharing a single processor. To take a fairly trivial example, there is no need for commands to be terminated—e.g., by RETURN or SEND—and single keystroke commands can more readily be implemented than in conventional configurations. Also, response times for actions which are not dependent on disk access—such as returning from a full record display to a screen of brief records—are constant and do not depend at all on the overall load on the system.

Although the Apple is one or two orders of magnitude slower than the minis or mainframes used in most systems, the effect of the distributed processing is to provide a good deal more computing power than most other systems at a cost which is comparable or even lower.

User Interaction

These are some of the assumptions on which the design was based:

—most users are either looking for "a specific book" or for "books about something;"
—users who are looking for a specific book generally know both the author and the title (although they may not have a very accurate citation);
—users who are looking for books about something will rarely describe a subject in a form which achieves even a partial match with a subject heading. Furthermore, the language of subject headings is not current, inverted order is confusing and inconsistent, headings are often too broad or too specific to match users' topics, and subdivisions are not always helpful;
—a large number of catalog uses are casual, and users cannot be expected to be persistent, enterprising, or enthusiastic;

—the catalog must be instantly usable without any training, experience, or knowledge of either library practice and terminology or of computers and computing terminology—few people would know what a "corporate author" is: words like "entry" and "subject heading" and "control number" are meaningless to most people;
—every display should be self-explanatory—"help" which has to be requested is rarely used;
—screen displays must be very clearly laid out. Where choices are offered it is confusing to give more than four at a time. Many people have trouble deciding between more than two options. It follows that to avoid making the system tedious to experienced users, the system should respond to memorized sequences of command keystrokes without going through all the intermediate screen displays (this is something which is rather easy to implement on a distributed system).

The original design team (the authors of *Designing an Online Public Access Catalogue*[4]) derived these precepts from a number of sources including study and observation of catalogs and their users, of interactive information retrieval (IR) systems in general, and from published material on online catalogs.

It may be felt that the result of working toward the earlier discussed design assumptions will result in a catalog which may be easy to use, but one which will not satisfy the needs of experienced users or those with specialized requirements. This view is cogently expressed by Anne Lipow.[5] (It may also result in a catalog which is boring to use—see the concluding section of Lipow's article.) The primary aim of the OKAPI experiments is to determine whether it is possible to make an online catalog that satisfies the usability criteria while providing a high degree of effectiveness.

OKAPI tries to do this by behaving a little less mechanically than most IR systems. In a search by title and author, for example, if there is no match on both fields, OKAPI searches for each one separately and may find the given title but not the author or the author but not the title. In either case the user is informed that there is no exact match and is given the choice of seeing either titles or authors which may provide what the user was looking for.

These "search trees"[6] are fairly elaborate and cannot be described here. In any case, since they were designed without precise knowledge of the types of and reasons for specific item search failures, they have proved not to be altogether satisfactory. In particular, it was not recognized that the most common cause of a "zero hits" result in specific item searches in a small catalog is that the library does not hold the item

sought. OKAPI tends to make the user continue fruitlessly. It is more helpful to the user to offer to search for the author only or for the title only while suggesting that "If you are sure you have the author and the title right, this book is not in the library."

Operation of OKAPI

OKAPI's initial screen offers a choice between searching for "SPECIFIC BOOK(S) (if you know the author and/or title)" and "BOOKS ABOUT SOMETHING." If the former is chosen, there follows a form-filling screen requesting title, author (surname only, if a person), and initials. If the user does not enter anything in the title box, the search is processed as an author search. If the author box is left empty the initials prompt does not appear and OKAPI does a title search. No distinction is made between corporate and personal names in the index.

The result of a specific item search (in OKAPI Mark 1) is a display of matching records in the case of unequivocal success, a browsing display of the author or title index in the case of partial success, or a failure message (usually with an option to see an index display).

The subject search is extremely simple. The user is prompted to enter "word(s) or a short phrase which describes your subject." The individual words of the query are looked up in the index for any source fields which may have subject content—i.e., titles and subtitles, subject headings, corporate names, or contents notes. If two or more words occur in the index they are combined using an implicit Boolean AND. If the AND succeeds, the user is shown the records.

If the AND fails ("no book exactly matches your search") but at least three words of the query occur in the index, OKAPI carries out a "best match" search ("looking for similar books"). The user's words are assigned weights which are inversely proportional to their frequency in the file. Thus, in "Skiing Holidays in Great Britain" the words *skiing* and *holidays* would have much higher weights than *Great* and *Britain*. Records are then ranked according to the sum of the weights of the words they contain in common with the query. The example query would result in the output of all records with *skiing* and *holidays*, then *skiing* or *holidays*, before records containing only *Great Britain*. A cutoff rule prevents the retrieval of records bearing little similarity to the query.

The procedure used for the best match search is similar to that in the National Library of Medicine's CITE catalog,[7] except that CITE also takes explicit account of the *number* of words common to the records and the query. Similar techniques have been used in a number of

experimental IR systems, and are now provided in one of the commercially available integrated library systems (the LIBERTAS system from SWALCAP Library Services Ltd.).

Evaluating OKAPI

One station was installed in PCL's Riding House Street site library in November 1984 and a further three stations in 1985. The users of this library are mainly undergraduate students of economics, social sciences, and communications.

The system logs itself comprehensively, and the usual raw statistical information can be obtained automatically. For example, the proportion of search types is roughly 40 percent author and title, 40 percent subject, 12 percent title only, and 8 percent author only. About 10 percent of searches contain at least one spelling or keying mistake. These are usually fatal in subject searches but often do not affect the result of author/title searches.

Some observation and interviewing has also been done. The initial results of this were reported in *Designing an Online Public Access Catalogue*. As with almost all online catalogs, the attitude of most users is favorable. What is far more difficult is to evaluate OKAPI's effectiveness. It is not sufficient to look for searches resulting in no retrievals. Much can be gained by "eyeballing" the transaction logs and by repeating real searches—particularly subject searches—to try to assess whether they were successful or not. The following two experiments are described in more detail in an article by Richard Jones (research officer on the current OKAPI projects).[8]

Some Recent Experiments

Success Rate in Subject Searches

One of the major difficulties in using transaction logs to evaluate OPACs arises from the fact that it is usually impossible to determine session boundaries with reasonable certainty. This is simply because catalog users cannot be expected to sign on or log in. It is often quite clear from looking at a log that a sequence of searches was done by the same person. One sees spontaneous broadening or narrowing of searches, and one search following within a few seconds of its predecessor is almost certainly by the same person. Conversely, if there is a substantial period (say three minutes) during which a terminal was inactive, then there has almost certainly been a change of users. Unfor-

tunately the terminals are not usually unused for as long as three minutes.

In an attempt to estimate the success rate of users' first attempts at a search for a given subject, Richard Jones and this author carried out the following experiment. The logs of thousands of searches were processed to isolate those subject searches which started at least three minutes after the termination of the previous search (of any type). This resulted in a set of just under 300 queries. In some cases, of course, the selected searches were followed by further searches on apparently related topics, but this information was not used.

Each of these searches was repeated and the results were classified as "successful," "unsuccessful," and "indeterminate." The criterion for "success" was that, in the opinion of the experimenter, at least one record in the first ten retrieved was relevant to the query *as understood by the experimenter.* The results were that 62 percent were successful, 13 percent indeterminate, and 25 percent unsuccessful. About one-fifth of the unsuccessful searches were apparent collection failures (nothing relevant found after thorough searching).

It is obvious that the success criterion is not a very realistic one. It does not take into account users' varying requirements in, for example, exhaustiveness. While many OKAPI users will be satisfied by finding one book on their topic, some (a few) will be trying to do an exhaustive search or may already have seen the book(s) which the experimenters judged to be relevant.

It is tempting to say that the only criterion for measuring success is to ask the user, but this question can often only be answered after the user has been to the shelves and had a look at the book(s) and so cannot even be asked until after the *session* rather than the *search* is complete. What we were trying to estimate was the proportion of searches that are successful at the first attempt with the user's first spontaneous formulation of the subject. To put it a different way, how well does the terminology of users' initial subject search statements match the vocabulary of the source file?

It would be interesting to repeat the experiment using title words only and subject headings only, but the source file is not homogeneous enough to allow this. It is suspected that the majority of searches which succeed do so on title words rather than on subject heading words.

Success Rate in Author/Title Searching

Jones[9] studied 214 consecutive author/title searches made at one station on three consecutive days. He found that 12 percent of these

searches failed to locate an item which the library held. (Some of these searches succeeded after reformulation or correction by the user.)

This figure is similar to those obtained by Dickson from the NOTIS system at Northwestern University library.[10] She found that about 11 percent of author searches and about 14 percent of title searches failed to find entries which were in the catalog (these figures are deduced from the figures for the proportion of each type of search and the causes of failure which Dickson gives—a zero-hit search is classified as a success if it convinces the user, correctly, that the sought item is not in the library). Dickson's initial selection of searches for study was made by scanning the logs for searches which resulted in no hits. The results are not strictly comparable (some of the NOTIS searches retrieved records without the users finding the one they wanted—e.g., "Smith, S" will retrieve everything from "Samuel Smith" to "Szymon Smith," and it is a large library). It is almost trivial to say that the most frequent cause of search failure—for all types of search—is that the user's terminology does not match that of the catalog.

Enhancing Subject Access

Current OPACs offer two (or perhaps three) approaches to subject access—i.e., by *headings* and by *keywords*. Some offer both methods, but how does the user know which to choose? (The third approach is not to offer subject access at all except via a printed subject index followed by a class number or shelf mark search.)

In searching by heading, there is the well-documented difficulty that users have in finding an entry to controlled subject headings. After all, subject headings were not designed for online searching—they are intended to be subject descriptions that users would *recognize* rather than be able to *formulate*. This difficulty in matching Library of Congress Subject Headings appears to be so general that subject headings are scarcely worth considering as the primary means of subject access in an online catalog.

Some United Kingdom libraries have built up subject indexes over the years in response to users' queries. Many of the headings in these take the form of *see* references, but references can be used invisibly in an online catalog. There is no need for the user to know that the switch has been made. This is one of the devices which we are using in OKAPI Mark 2 (see discussion following).

For all their failings, subject headings undoubtedly can perform useful functions. One function is that of helping the user to recognize whether or not a retrieved item is likely to be relevant. Unfortunately,

many United Kingdom catalog records do not display subject headings. Even if the record does include subject headings, they cannot be displayed in brief entries. Hence, OKAPI Mark 2 generally displays all records in full—one per screen—following a subject search. Another function of subject headings is as a linking device—i.e., if a record is judged relevant, other records with the same subject heading may also be relevant, so the user should be given an option to "see other books described in the same way." (Unfortunately we can't do this in OKAPI because too many of the records have no subject headings—but classification codes can be used in much the same way.) It follows that postcoordination of the individual words or subphrases of the query should be the primary initial means of subject access.

Conventionally, in second generation OPACs, this is done by using an implicit Boolean AND. This leads to too many zero-hit searches. Most searches containing three or more words fail on an AND except in the very largest catalogs. There is little doubt that some form of "combinatorial" or "best match" search (as used in CITE, in OKAPI, and in the SWALCAP LIBERTAS system) is the best way of providing postcoordination. It has the additional advantage of automatically providing ranked output (users must be informed that "the most similar items should appear first") thus going a long way toward eliminating the problem of "too many hits." Few IR theorists would now hold that there is much to be said for conventional Boolean reference retrieval systems. They are of little use without trained intermediaries. The only satisfactory way of outputting records is in decreasing order of probability of relevance. However there is little agreement between theorists on how this should be achieved. The schemes used in the earlier-mentioned systems do have the merit of being relatively light computationally and of using a conventional inverted index structure.

Related Terms and the Synonym Problem

There remains the "synonym problem." A search intermediary knows that "infants" and "newborns" are to be treated as synonymous in a search for "Kidney disease in infants and newborns," but a computer program doesn't know this and a record containing both these words will be given a falsely high weight. The other side of the synonym problem is that of bringing in related words. An intermediary will often do this by using truncation to include morphologically related terms or by "ORing" such terms as "infants" and "newborns."

Automatic Stemming

Truncation can be done automatically—with reasonable precision—by using automatic stemming or suffix-stripping algorithms that will produce, for example, "comput" from "computer," "computers," "computational," and so on. We tried this in OKAPI using a compact stemming procedure developed by Martin Porter.[11] This has been tested on fairly realistic collections and searches and found to behave as well as explicit truncation by skilled intermediaries. Unfortunately, even fairly conservative automatic stemming does not always work well if it is applied to all searches. It can generate unacceptable amounts of "noise" if applied indiscriminately. It is particularly dangerous when applied to those OPAC subject searches (and there are many) which consist of only one word. It proved to be impossible to retrieve records on "communism" without retrieving everything on "communication." An intermediary would not, of course, truncate "communism." On the other hand it is difficult to use linguistic knowledge in a computer program to decide when to stem and when not to.

One solution would be not to apply stemming to single term searches, but we think it may be better to use a two-stage stemming procedure. The first stage—weak stemming—reduces regular English plurals to singulars and removes the verbal noun suffixes "ing" and "ed." It also conflates alternative spellings so far as this can be done without extensive look-up tables. It can, for example, cope with "iz" and "is" alternatives and with terminal "our"/"or," but not with "aluminium"/"aluminum."

The second stage—strong stemming—removes a fairly wide range of suffixes. The intended search procedure is to take the words of the user's input, subject them both to weak and to strong stemming, and feed all the resulting terms into a combinatorial search with the additional rule that if a record is indexed both by the weak stem and the strong stem, no additional weight is given for the occurrence of the strong stem. We have designed a combinatorial search procedure that does this, and it will be evaluated later in the year.

Synonym Tables and Cross Reference Lists

Conventional subject indexes sometimes attempt to deal with terms that have related meanings but that are not alphabetically close by using *see* and *see also* references. For personal and corporate names, many

libraries use authority control, and there is no reason why an online catalog cannot automatically switch from the form entered to the "preferred" form. However, Arlene Taylor's study of failed name searches on the NOTIS system at Northwestern University library,[12] shows convincingly that name authority control, as currently practiced, would have helped in only a small proportion of the searches. She concludes that enhancement of the search programs to make them perform an automatic "flip" of forename and surname and to retrieve the best possible partial matches with user input would have been far more useful.

For subject searching in OKAPI Mark 2 we are incorporating lists of "synonym classes" of subject terms. Here the word *term* means not only single words but also phrases like "United States of America."

Because of the wide subject coverage of most catalogs, care has to be taken not to equate terms which are synonymous in one context but not in others. For example, "plant" can be synonymous with "factory," but biologists will not want to retrieve material on the manufacturing industry. In some cases, words can be equated when they occur in specific contexts: for example, "underdeveloped" equals "developing" equals "third world" when followed by "countries." (Note that there is no need for one member of a class to be regarded as the preferred form. In the aforementioned example, "third world" is current and is the most likely term to be used by searchers, but older material may be indexed under either of the other terms, and all records should be retrieved no matter which member of the class the user enters.)

Our list of synonym classes is derived from a study of the terminology used in some 6000 OKAPI subject searches. Generally, one or more of the members of a class is a noun phrase or an abbreviation. Some of them serve simply to relate irregular plurals to their singular forms, and some serve to handle alternative spellings—for examples where this cannot be covered by a rule, see the following:

"United States of America" = "US" = "USA" = "United States"
"child" = "children"
"BBC" = "British Broadcasting Corporation"
"Tsar" = "Czar"

Incorporation of these synonym classes into the index involves the use of a "go" list for phrases. Automatic indexing is extremely simple if the index consists of words because there are very simple rules for splitting a field into words. Using a "go" list at indexing time makes the process slower and more complicated. The individual words of these phrases also contribute to the index. A user looking for material on "broadcasting" might well be satisfied by items indexed under "BBC."

Spelling and Typographical Mistakes

There is a vast literature on the automatic detection and correction of keying errors, and this is not the place to discuss them in any generality. There are now computer programs which can automatically correct a large proportion of mistakes, but the more effective ones are computationally heavy. With conventional library automation equipment it is rarely possible to attempt spelling correction in real time. Such procedures also require very large dictionaries.

At the present stage in its development, OKAPI Mark 2 helps users to recognize possible mistakes by clearly displaying messages like "Can't find SOCIALOGY." It also incorporates some modified Soundex-type indexes. The algorithm used is less "fuzzy" than the conventional Soundex procedure that reduces keys to four-character codes after removing all vowels (except an initial vowel). Our codes are not limited in length, and vowels or vowel groups (apart from terminal *e*) are represented by a single character. The procedure will therefore tend to give higher precision and lower recall than the standard Soundex. It will rarely produce a match if a word has undergone character transposition, but it quite often succeeds with errors which are misspellings rather than miskeyings ("socialogy," "psycology"). In a sample of 621 subject words taken from the OKAPI transaction logs, 64 were misspelled but immediately recognizable to the human eye. Thirty-two of these would generate the same modified Soundex code as their correctly spelled equivalents, and in many cases this would have been unique (the Soundex key would only have arisen from one source word). This suggests that a considerable proportion of words which are (1) misspelled, and (2) occur in the index, could be automatically corrected by using a subsidiary index of modified Soundex codes.

The same modified Soundex coding has been applied to personal surnames. In this case, when OKAPI fails to find a surname and the Soundex code is present in the index, it displays a list of "names which sound similar" for the user to choose from. It is doubtful if this feature will be used much because OKAPI users seem to spell personal names rather accurately. Even when a surname is misspelled in an author/title search, the search often succeeds because the primary access is by a 4/4 author/title key which will ignore any errors after the fourth character of the title and of the author's name.

Relevance Feedback

Much research in IR has been directed toward investigating ways of automatically finding records that have a high probability of being

similar in subject content to those which match the terminology of the query. Such methods include *clustering techniques* (terms are divided into groups on the basis of their probability of cooccurrence, and the system will retrieve records indexed by as many terms as possible which regularly occur in conjunction with index terms matching the query), and the use of *relevance feedback*—i.e., the query is modified by, for example, adding terms from records judged by the user to be relevant to the query and then reprocessing the search. Harper's thesis[13] contains a rather comprehensive account of relevance feedback techniques.

The OKAPI relevance feedback project is concerned with investigating ways of obtaining relevance assessments, their reliability (how well can users judge the relevance of material just from a bibliographic description and subject headings), and how to use them automatically, in real time, to improve the precision and recall of searches.

After showing a record, the system can ask "Is this at all the sort of thing you were looking for?" Following a positive response there are various approaches that can be used automatically to reformulate and reprocess the search so as to try to find closely related records. In particular, other records classified at the same Dewey or LC number may well be relevant. The use of classification (Dewey or Library of Congress) alone will often decrease the precision of a search. This effect may be minimized by using a combination of classification together with title and subject words from relevant records. It may also be possible sometimes to exclude records on the grounds that they contain terminology in common with record(s) judged nonrelevant.

Conclusion

In a forthcoming review of the OKAPI report for *Program*,[14] Charles Hildreth suggests, parenthetically, that OKAPI is not *fun* to use. This author agrees with him and also would like catalogs to be fun to use. They should allow those who are involved or interested enough to have a great deal of control over the search process and offer multidimensional browsing of related material. This author submits, though, with some trepidation, that (1) most of our files of bibliographic records and headings do not contain enough information nor information of the right kind, and (2) that it is more useful to the general patron to be provided with a catalog that produces good results most of the time without demanding much in the way of knowledge, experience, or skill. Most catalog uses are quite casual. They are attempts to satisfy a real need, but this is generally a need that the user feels should not demand much involvement on his or her part. Perhaps attitudes to catalogs will

change if we offer OPACs that are both clever and easy—there are signs of this happening. It ought to be possible for the computer to determine a user's degree of skill, experience, and involvement, and to adjust its interaction accordingly. A future OKAPI project may be concerned with work toward the development of such a self-adaptive catalog system.

References

1. Mitev, Nathalie, et al. *Designing an Online Public Access Catalogue* (Library and Information Science Research Report No. 39). London: British Library, 1985.

2. Ibid.; and Venner, Gillian, et al. "OKAPI: A Prototype Online Catalogue." *Vine* 59(July 1985):3-13.

3. Seal, Alan, et al. *Full and Short Entry Catalogues*. Bath, England: Centre for Catalogue Research, Bath University Library, 1982.

4. Mitev, et al., *Designing an Online Public Access Catalogue.*

5. Lipow, Anne G. "Practical Considerations of the Current Capabilities of Subject Access in Online Public Catalogs." *Library Resources & Technical Services* 27(No. 1, 1983):81-87.

6. Mitev, Nathalie N., and Walker, Stephen. "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). London: Aslib, 1985.

7. Doszkocs, Tamas E. "From Research to Application: The CITE Natural Language Information Retrieval System." In *Research and Development in Information Retrieval*, edited by Gerald Salton and Hans-Jochen Schneider, pp. 251-62. Berlin: Springer, 1985.

8. Jones, Richard M. "Improving OKAPI: Transaction Log Analysis of Failed Searches in an Online Catalogue." *Vine* 62(May 1986):3-13.

9. Ibid.

10. Dickson, Jean. "An Analysis of User Errors in Searching an Online Catalog." *Cataloging & Classification Quarterly* 4(Spring 1984):19-38.

11. Porter, Martin F. "An Algorithm for Suffix Stripping." *Program* 14(July 1980):130-37.

12. Taylor, Arlene G. "Authority Files in Online Catalogs: an Investigation of Their Value." *Cataloging & Classification Quarterly* 4(Spring 1984):1-17.

13. Harper, D.J. "Relevance Feedback in Document Retrieval." Ph.D. diss., University of Cambridge, 1980.

14. Mitev, et al., *Designing an Online Public Access Catalogue.*

This Page Intentionally Left Blank