8-1992

# Text Retrieval Software for Microcomputers and Beyond: An Overview and a Review of Four Packages.

Gerald W. Lundeen

Carol Tenopir
*University of Tennessee - Knoxville*



## Recommended Citation

Gerald W. Lundeen and Carol Tenopir. "Text Retrieval Software for Microcomputers and Beyond: An Overview and a Review of Four Packages." Database 15 (4) (August 1992): 51-57, 59-63.

# Text retrieval software for microcomputers and beyond: an overview and a review of four packages

**Authors:** Gerald W. Lundeen and Carol Tenopir
**Date:** Aug. 1, 1992
**From:** Database(Vol. 15, Issue 4.)
**Publisher:** Information Today, Inc.
**Document Type:** Product/service evaluation
**Length:** 8,706 words

Abstract:
Full text file management requires unique software capabilities because of the special characteristics of textual files. Text files vary in size, but are often large, the data is in alphabetic form, and format may differ within a file. Features important to text retrieval include word adjacency searching, truncation, Boolean logic and free text searching. Four textual file software packages - Concordance, Concept Finder, Personal Librarian and Topic - are evaluated.

Full Text:
As the capacity of hard drives grows bigger and the cost of hard drives goes down, many people want to maintain large files of text on their microcomputers. These text files can be the result of word processing, including original letters, memos, reports, manuscripts for books or articles, and the like. Or, they can be full texts and bibliographic records downloaded from various online retrieval systems, or from CD-ROM systems. Optical scanning combined with optical character recognition conversion provides yet another source of buildings huge textual files. Some textual files may be a combination of all of these date entry/document creation techniques.

Software for textual files have seen active development in the last few years as the need for such packages grows. In this article we review four such packages in-depth: Concordance, Concept Finder, Personal Librarian, and Topic. Before we describe the specific features of each package, an overview of the characteristics of textual files and the requirements for text retrieval software will help put the specific comments into perspective.

UNIQUE CHARACTERISTICS OF

TEXT

Some software packages for text are enhanced versions of packages originally developed for bibliographic files, others are newly developed with full text in mind. In either case, the search and retrieval capabilities of these packages are necessarily different from those of software made for other types of databases or files. This is because of some unique characteristics of textual files:

* the data is primarily alphabetic and

when numbers are included they are

usually treated as text

* the files are often very large

* the files are often very large

* the file may consist of a single text

(e.g., and encyclopedia) or may be a

collection of many independent texts

(e.g., journal articles on one topic

downloaded from an online database)

* the size of the texts in one file

may vary from short memos to

entire books

* collections of texts in a file may be

relatively uniform in format and size

(e.g., a correspondence file) or they

may vary considerably (e.g., correspondence,

reports, contracts,

articles, books)

In addition, there may be varying degrees of structure in the texts, including:

* records with fields, such as a typical

bibliographic record

* amorphous text with appended

fixed fields, such as a file of journal

article texts with a bibliographic

citation for each

* text with inherent structure or format

such as letters with date, inside

address, salutation, body, sender.

* amorphous text with its inherent

structure of sentences and paragraphs

If there are texts that have a typical record/field structure, such as is common with records downloaded or transferred from other databases, there may be great variety in the fields. Fields in textual files share some common characteristics in their diversity, however.

* there tend to be many fields (e.g.,

author, title, source, date, subjects,

locations, complete text, etc. in a

journal article file

* the fields are mostly variable length

* several fields (such as complete text

and subjects) may be very lengthy,

while others (e.g., date) may be

very short

* the length of most fields will vary

considerably from record to record

* fields often contain repeating values,

such as multiple authors or multiple

subject headings. but whether they

repeat and how often they repeat varies

SEARCH CAPABILITIES

With any textual file application, search capabilities are very important. If there are field, typically searchable access to most of them is needed. With or without fields, information in textual databases is represented in a variety of ways with all of the ambiguity and complexity of natural language. Most retrieval is based on searching for unknown items, rather than selecting a desired record. Therefore, sophisticated retrieval techniques are needed.

Certain search and retrieval capabilities are by now standard for software for structured bibliographic files and should be considered base minimum level for any textual file. These include such features as truncation, word adjacency searching, comparison operators (greater than, less than, equal to), field specification, free-text searching, set building, and Boolean logic. All of these are familiar to anyone who has ever searched a commercial online information system or built a bibliographic database on a microcomputer.

In addition, there are other search features that are needed for full texts that may not be available with software intended for bibliographic files.

These include:

* searching within a specified number

of words

* searching making use of grammatical

structure, including within a

sentence or specified number of sentences

and within a paragraph or

specified number of paragraphs

* left hand and internal truncation

* automatic stemming for singular/plurals

and other word stem variations

* automatic language enhancement,

including automatic abbreviation

expansion and other equivalencies,

such as British/American spelling

* thesaurus features, such as synonym

expansion, word profiles, etc.

* key-word-in-context displays with

search terms highlighted

Additional search and retrieval capabilities that enhance the search process are available in some text retrieval software. These include (but are not limited to) such things as hypertext links, word occurrence information, ranked output, fuzzy sets, relevance feedback, sound-alike retrieval or other partial match algorithms, and the ability to include images. These non-standard features may be unique to on package, so they will be discussed in more depth as they are available in the package being reviewed.

All search and retrieval features must, of course, be balanced with ease of use. Some of the more sophisticated features may not be of equal importance for you or for your application. For example, left hand truncation may be essential for a chemistry database, but of limited use in other topics. Few packages have everything, and prioritizing the importance of features must be done on an individual basis. In addition, there is often a trade-off between the power of a software package and its ease of use. We recommend

doing a personal needs analysis for your application before evaluating software[1].

## SOFTWARE CATEGORIES

Not all software that can be used for textual files is the same. Often packages are not even labeled full text or text retrieval software, or several packages labeled the same may be quite different. Because different packages are better suited for different things, we find it helpful to categorize text retrieval software by general characteristics[2]. As with any attempt at categorizing, not every package fits neatly into one (or into only one) category, but the categories describe typical strengths and features.

Structured Text Retrieval packages were often originally developed for bibliographic databases. They require fields and field characteristic to be specified before any records go into the file. Usually there is a database definition language or configuration module that is used for this initial set-up. The field structure can make searching more precise and faster and allows more control over output formatting. Of the four packages reviewed here, Personal Librarian is of the structured text retrieval type.

Unstructured Text Retrieval packages have no initial set-up and accept incoming text files without fielded structure. (Fields may be present in the input texts, but they are not utilized for search or display.) These packages often recognize inherent grammatical structure of text, such as sentences and paragraphs, for searching and for output formatting. Unstructured text retrieval software are particularly useful for existing unfielded word processing files or for files downloaded from a variety of incompatible fielded databases. Few packages are totally unstructured, and none of the four in this review fall in this category. Packages that do include ZyINDEX, Lotus Magellan, and Gofer.

Hybrid (or combination) text retrieval packages are more common these days. They provide the best of both worlds, supporting pre-defined field structure in addition to unstructured text. The fielded information in some of these packages must be forced into fixed length fields, others allow more flexibility in field definition. A typical use of the combination is to put bibliographic citation information into fields with full texts unfielded. Hybrid packages often have powerful and unique search features. Three of the four packages in this review fit here: Concept Finder, Concordance, and Topic.

## ANOTHER WAY TO CATEGORIZE

Text retrieval packages can also be categorized from the perspective of storage options, as suggested by Ernest Perez[3]. There are two basic ways storage is handled: 1) archival text retrieval and 2) text file indexer.

Archival Text Retrieval software requires a copy of each text to be loaded into the software before they can be searched. If the text is from active word processing files this means that two copies of the file are needed - one in the text retrieval system and one in the word processing system - but they typically have the most powerful search and output features. Structured text retrieval software packages are usually archival. Concordance, Concept Finder, and Personal Librarian are of the archival text retrieval type.

Text File Indexer software searches files in place. The packages in this category were designed for managing retrieval of large word processing files and can deal with most of the popular word processing formats. Text file indexers don't duplicate files, instead they maintain an index that provides rapid access to the existing text files. Popular text file indexers include ZyINDEX and Lotus Magellan. Of the packages we review her, Topic can be used as either a text file indexer or an archival text retrieval type.

Table 1 gives names, contact addresses, phone numbers, and prices for the four packages reviewed here. Details of each package are given below. Major search features for each are summarized and compared in Table 2.

TABLE 1 PRODUCT INFORMATION Concordance Dataflight Software 10573 West Pico Boulevard, Suite 68, Los Angeles, CA 90064 213/398-2787 Standard Edition: $495 Professional Edition: $995 Runtime Modules: $125 Network versions: Standard: $1,500 Professional: $3,000 Runtime: $3,000 DOS 2.1 or higher, or OS/2. Concept Finder MUMPS Medical Information Management Systems, Inc. (MIMMS) 2210 Midwest Road, Suite 212 Oak Brook, IL 60521 708/575-0090 DOS 386 version: $1995 286 version: $1595. Personal Librarian Personal Library Software 15215 Shady Grove Road Rockville, MD 20850 301/926-1402 DOS/Windows: $895 XENIX/UNIX (Single user): $1295 Network and multiuser: $5000 and up. Topic Verity, Inc. 1550 Plymouth Street Mountain View, CA 94043-1230 415/960-7600 Servers: $15,600-$150,000 DOS or Mac interface for client server: $795 OS/2 or UNIX workstations: $1,000

## CONCORDANCE

Dataflight Software sells a regular and a Professional version of Concordance. The Professional version comes with a built-in programming language that supports the development of custom applications. With the Professional version comes the option of a run-time module for distribution of customized applications. A network version is also option.

### Maintenance

Concordance supports fielded data (up to 100 fields per database), with one "paragraph" type, which accepts free text up to 65,000 characters. More than one paragraph field can be created in a file, but since the text import feature of Concordance only allows importing text files into one field, the effective document length limit is 65,000 characters. Longer text files are accommodated by splitting them into multiple documents. When the character count reaches 60,000 during import, the imported file is split into two (or more) documents with the break at the end of the sentence after the 60,000th character. In our test data, a file that exceeded 60,000) character (but was less than 65,000) was split into two separate documents. One problem with this approach is that the individual documents are not linked so that a search that pulls up the first document of a multiple document text will not display the rest of the text. In the case of our 61,000 character file it was possible to go into edit mode to cut and paste the contents of document 2 onto the end of document 1.

In addition to the paragraph field type, Concordance includes defined length fixed fields. The defined length fields can be either textual, numeric, or date. These fixed length fields can be searched with comparison operators (=,<,>), range searching, and Boolean operators. These fields are useful for sorting the database, and for searching that requires comparison (not supported in paragraph fields). Sorting can be done on paragraph fields, but in that case only the first 32 characters of the first line are used and sorting is much slower.

Data can be entered form the keyboard a field at a time with the usual full-screen editing capabilities. For long documents the file import option is likely to be more important. All imported text goes into a single paragraph field. The import option will not load multiple fields or fixed length fields. Fielded text files can be loaded with the LOAD option. This requires that the data be in ASCII format with field values enclosed in quotes and delimited by commas. This option is useful for porting data from one Concordance database to another, or for transferring data from another software package that exports in comma delimited format.

The overlay option loads comma delimited ASCII files into existing Concordance documents, replacing selected fields with new data. Field not selected are left intact. LOAD allows changing the order of fields. To minimize the chances for confusion with comma delimited ASCII files, Concordance allows the use of special characters not normally found in text to take the place of the comma, quote, and newline characters.

Newly entered documents need to be indexed as a separate operation before being searchable via paragraph fields. Fixed fields are immediately searchable because they are not indexed. A separate packing operation does two things: it physically removes documents that were marked for deletion; and it optimizes the search files (indexes). The manual claims that for a packed database indexing will be 600% faster and free-text searching will be up to 700% faster. It seems that this kind of optimization would be automatically done along with indexing.

Since the LOAD option works only with comma delimited files, and many users have fielded text files in plain text format, Concordance comes with a separate utility program, Convert, that converts plain text files into comma delimited ASCII format. There are specific requirements that the text file must meet (field labels must be in capital letters and appear flush left, must be exactly the same as defined in the database, etc.)

Retrieval Capabilities

Fixed fields can be searched by means of comparison operators (=,<,>,<=,<>,$,! or their two letter equivalents, EQ,LT,GT,LE,GE,NE, CO (contains), NC (not contains). For text searching, all paragraph fields in a database are searched together, but since the import module limits importing to one paragraph field this probably will not be critical. Searches are numbered sequentially and can be referred back to in later searches. Search number 0 is always the entire database. Searching is case insensitive. In text search mode ten search operators are available: Boolean operators (and, or, not, xor), proximity (adj, adj1-adj99, near, near1-near99), context operators (same, notsame) and limiter operators (the single period . and the double period ..). Unfortunately, the same, notsame, and the limiter operators operate on fields and not grammatical paragraphs. Since text is all inn one paragraph field, the text search expert same systems is equivalent to expert and systems. Proximity searching will find words close to each other but independent of grammatical structure. The ability to search for words in the same sentence or grammatical paragraph would be useful additions.

The limit operators (. and ..) restrict searches to specified paragraph fields, the single period finds those documents with the search expression being satisfied in the specified paragraph field. The double periods finds those documents where the search expression is satisfied in paragraph fields other than the one specified.

For example.

online.title. will find documents with the word online in the title;

online..title. will find documents with the word online in other text fields (it may also be in the title).

(vocabulary near3 control).title, abstract, text. will find documents with the expression satisfied in any of the three fields.

5.title limits search 5 to the title field.

A user definable stop word file is used to eliminate noise words and to cut down on the size of the index. This stopword list can be defined for each database. If none is defined, a default stop list is used (also modifiable by the user).

The index is browsable. Entering a term or term fragment brings up a screen with the part of the index where the entered term is or would be . The cursor control keys can be used to browse up or down within the index.

Truncation is supported in text search mode. The truncation or wild card character is user definable (default is the asterisk), In select mode both a single character wild card (?) and truncation character (*) are supported (Figure 1).

Queries, both search and select, can be saved for later reexecution. This facilitates searching several databases in the same session, and searching a single database over time.

Output Options

Search results are first summarized in a results table showing the number of occurrences of each term and number of documents for each term and for the query. The retrieved document set can be browsed going forward or backward a document at a time or jumping

to a specific documents. Within a document the text can be browsed line by line, page by page (screen by screen), or by highlighted page (containing query terms). Query results can be sorted ascending or descending on up to ten fields. This makes sense for fixed length fields. but probably not for text fields. The sorted results can be browsed, printed, unloaded, overlaid, edited, or used to produce columnar reports. The total length of all sorted fields cannot exceed 150 characters.

PRINT allows documents to be sent to the printer or to a file. When printing, Concordance allows the used to specify what field to print and their order. Print formatting allows control over page numbering, document numbering, page length, margins, field labels, and whether a new page will be issued for each document.

Columnar reports produce output in table format. This option will not be of much use for full-text files but other types of databases can make use of this feature. Fields containing numeric data can be subtotaled and totaled for reports.

The unload option can do two things: it can create an empty database with the structure of the current database, or it can create a file with data from the current database in comma delimited ASCII format. As with print, unload allows selected fields to be output in specified order. The output can be sorted and a selected portion of a query set can be specified for output.

Documentation

The 160 page Concordance manual contains a 35 page tutorial section, 89 pages covering all the capabilities and features of the system (other than the programming language of the Professional version), a lists of error messages with explanations, a detailed table of contents and a fairly detailed index. The writing is clear and pictures of screens are used where appropriate. The manual is surprisingly adequate considering its size. Context sensitive online help is available. Help messages are in a text file that can be browsed once in help mode.

The Programming Language

A built-in programming language allows Concordance to be used for customs applications. Concordance Programming Language (CPL) is similar in syntax and structure to Pascal, the difference is that all of the capabilities of Concordance are available for use in programs. The package comes with several source code programs that serve as examples. The manual for the language is primarily a reference manual and not a tutorial on programming, but anyone with a knowledge of a modern structured programming language, like Pascal or PL/I, should have no difficulty in learning to use CPL. With the programming feature, graphic images can be attached to Concordance records. A run-time module is also available so the developer can distribute customs systems without having to purchase additional copies of the Concordance program.

General Comments

Concordance is an easy-to-use package with a well-designed user interface. It offers the standard search features needed for text retrieval and includes integrated editing features and a report writer. The Professional Edition offers powerful programming capabilities for custom applications.

CONCEPT FINDER

Concept Finder from MIMMS, Inc. offers some unique capabilities. It runs under the MUMPS operating system, which in turn is run from DOS. MUMPS is a multiuser operating system so a 386-based microcomputer can support up to ten users. The system is also available for minicomputer hardware. The file and record structure is similar to Concordance and Topic with fixed length fields plus full text.

Maintenance

Concept Finder divides documents into three parts: database; textbase, and annotations. Documents are arranged in logical groupings called files, which in turn are arranged in subjects. When setting up the Concept Finder system, the user is required to expand the database to the anticipated maximum size. The database can be further expanded at a later time, but this will result in a fragmented file with slower response time.

The system comes with a standard file defined with 21 fields: document name, document number, date/time created, document type, source, location, reference, action, general, addressed to, author, copies to, marginalia, number of pages, annotations page, annotation, key word, follow -up note, entered, file name, and text. Custom files can be defined also. Fields are specified as being one of eight types: date, numeric, set of codes, free text, word processing, computed, pointer, and variable pointer. Fields can be multiple, and can be declared mandatory.

Concept Finder allows a great deal of control over values that go into database fields. When defining a field, default values can be specified. Freetext fields can have pattern match templates specified, for example X?3N1"-"2N1"-"4N for a social security number of the format 999-99-9999.

Fields may be cross reference (indexed) in any of six ways:

* regular (the first 30 characters of the field value is used)

* KWIC (each word of three or more characters (except stop words)

* MNEMONIC (the field values index along with the NAME field; allows searching of, for example, the MAIDEN NAME field along with NAME values)

* MUMPS (Programmers can devise special operations to perform on field values)

* SOUNDEX (Field value is transformed into a 4 character Soundex string)

* TRIGGER (Whenever the field is updated, a different field is updated at the same time., The other field may be in a different file.)

The fields types pointer and pointer variable support multiple file systems. Much of the power and flexibility of Concept Finder is derived from this relational feature. Three kinds of file linking possibilities are provided for:

1. A field in the present file points to another file.

2. There is no pointer in the present file but an entry in the other file can be looked up based up based on some value in the current file.

3. The other file contains a field that points back to the present file.

Computed access fields support the usual mathematical operators, as well as comparison operators, Boolean operators, and functions for square-root, modulo, absolute value, maximum, and minimum. Also provided are an extensive set of functions for manipulating character strings, date and time values, multiple valued fields (total, count, maximum, minimum, 1st, 2nd, ..., last), text processing formatting. Computed fields can also refer to data in other files by using the three types of file linking described above.

Data security is provided by sign-on passwords and a Programmer's Access Code for every system user. This gives the system administrator control over who can read, write, delete, modify data dictionaries, templates, etc. An audit file can be activated for any file to keep track of changes.

Data entry can be via the keyboard or imported from a disk text file, an optical character reader, or from a modem. Two types of automatic data input are supported: batch loading full text and batch loading document registration. Before full text can be loaded into a file the documents must be registered. This involves creating a database record with at least one field for each document to be loaded, so that the text file can be associated with a record in the database file. The requirement of registering documents before loading the full text was a nuisance. It would be much better if it were possible to load both the database fields and the full-text fields at the same time.

Text files to be loaded can be in either ASCII format (carriage returns and line feeds at the end of each line and page feeds at the end of each page), or in Informatics/Baron/Atlis Systems format (all lines padded to 80 characters with spaces, no linefeeds or formfeeds, each page begins with a five-digit page number on a line by itself). We loaded our test documents from ASCII disk files and found the process straightforward. The only complication was with page numbers. Our file did not have from feeds at each page so they loaded as single page documents unless page markers were entered as the text was being loaded. This was an imprecise method of marking pages and the result was that the NEXT PAGE and PREVIOUS PAGE commands didn't allow browsing to all parts of the document. Going back and entering formfeed characters in our text files before loading would take care of this.

Retrieval Capabilities

The Concept Finder search commands are basically those of Mead Data Central's LEXIS/NEXIS systems. To begin searching, one selects a subject and then one or more files within the subject. Search connectors supported are: AND, OR, AND, NOT, PRE/n, W/n, NOT W/n, W/FLD. Plurals are automatically found if they end -s, -es, or -ies. Irregular plurals can be automatically found by entering them into the thesaurus. Both single character wild card and variable length right truncation are supported.

The thesaurus included in Concept Finder can broaden a search by automatically including synonyms and near synonyms for search terms. Thesaurus terms are not limited to single words; phrases, abbreviations, or complete sentences are allowed. The thesaurus feature is normally turned on but can be turned off for a term or for an entire query by using a single or double quote mark, respectively.

A retrieved set can be further searched by selecting the secondary search option. This allows narrowing a search in a stepwise manner (Figure 2). Search strategies may be saved for later execution.

Output

The initial result of a query is a report of the number of hits. To view the search results a display format must be entered. A KWIC displays shows highlighted search words with 19 searchable words on either side. FULL display shows the entire document with search words highlighted. CITE displays the bibliographic reference for the documents. SUMM provides a summary table of found words and the document, word, and page numbers. FLDS displays a list of fields from which the searcher can select any or all for display. PAGE VIEW displays each physical page containing search terms. With the various view options the searcher can browse among pages in a document, and among documents, and can print or write to disk single documents or the entire retrieved set. Search results may be sorted on any combination of fields.

A built-in report writer allows the definition of customs report formats for the database fields. The report writer is relatively easy to use and offers a great deal of control of output. It even includes limited statistical processing such as descriptive statistics (mean,

standard deviation, minimum and maximum), scattergrams, and histograms. The full-text field is not accessible to the report writer, so reports that include full-text will have to be made by exporting to an ASCII disk file for processing outside of Concept Finder.

Use Interface

Concept Finder uses a character based line-oriented interface. This is probably due to its MUMPS multiuser, terminal-oriented foundation. This is the weakest part of the package. In these days of windows and graphical user interfaces, having to deal with line-oriented editors is a real frustrations. Fifteen years ago, when we would have taken the line-oriented interface as standards, this would have been a dynamite package. Now, were it not for all the power and flexibility that the packages offers, we would consider it too outdated. As it is, because of its unique features, there are applications where this is the obvious choice. A screen-oriented editor and more point-and-shoot menus would go a long way toward making this an easy-to-use package.

The program dumped us out to the DOS prompt on occasion. When this happened, the next logon brought up a message insisting that we run a check of the files to be sure nothing was corrupted from the abnormal exit.

Documentation

Concept Finder comes with a 235 page user manual and a 60 page manager's manual. Both are well illustrated and well indexed. The writing is clear and enough detail is given to make most tasks reasonably straightforward. Online help is available and the help screens can be customized by the systems manager. In interacting with the system a question mark in place of a response will normally elicit a list of acceptable options or choices.

General Comments

Concept Finder's strength is in its multifile capabilities and control over data. Complex relationships can be set up among data fields in multiple files. This , plus its LEXIS/NEXIS compatible search language, make it an attractive package for those using these online systems. Updating the user interface and making it a bit more resistant to user error would make this still more attractive.

PERSONAL LIBRARIAN

Personal Librarian began life a SIRE, first as an experimental system at Syracuse University and later as a commercial product. The name has been changed and the software has continued to evolve but the main feature that makes this package exciting remains the word frequency-based retrieval and ranking of output.

Maintenance

Personal Librarian requires much of the preparation and maintenance of the databases to be done at the DOS prompt using a text editor. Databases in PL have fielded structures and the field labels must be in a specific format. Up to 255 fields are allowed and at least one is required. A text field can contain the full text of documents with any additional information (authors, title, etc.) put in separate fields. Database definition involves creating an ASCII file, which specifies the fields, the type of data that will go into each, whether they will be searchable, displayable, and other information about the database.

Text files must be imported into the system and must be in ASCII format with field tags in the form -TEXT- and an -end- tag at the end of each document. Database files may be on more than one drive and subdirectory. A database directory file keeps track of where they are. Maximum record/document size is four Gigabytes.

Personal Librarian is very flexible. Many options are controlled by means of an editable ASCII file of system options (STRINGS.DAT) file. Other options are indicated in the database definition file and Windows WIN.INI file.

Personal Library Software sells the search engine separately as Callable Personal Librarian. This allows programmers to use all the retrieval capabilities of Personal Librarian in custom applications.

Retrieval Capabilities

PL supports the standard Boolean operators and field specification but adds to this some powerful capabilities for text searching based on word stem frequency of occurrence statistics. Due to the ranking of output, a natural language query with key terms included will probably retrieve some relevant documents. Once some relevant documents are found they can be used as queries. This "like document" mode of searching uses all the significant words in the query document ORed together as a query. Again, ranking of the results puts the most likely candidates at the top of the list.

The weighting and ranking is not controlled by the user or database administrator. It is built into the software and is based on frequency of occurrence. Retrieval is Boolean based and the ranking is then done to the retrieved set of documents. Free text entered as a query is searched using OR logic. For example, the query VOCABULARY CONTROL translates to VOCABULARY OR CONTROL. This default logic can be changed to AND or ADJ if the OR logic is bringing up too many irrelevant documents. Even with OR logic the ranked output will put the documents with both terms ahead of those with only one.

In addition to the Boolean operators the system provides truncation, proximity (adjacent or within a specified number of words), document as query, index browsing, display of statistically associated words, search using this statistically associated word list, and comparison operators.

>, <, =, <=, >=, <>

When indexing text, Personal Librarian truncates index terms to 12 characters (this can be changed to other lengths). The system can be configured to apply automatically a stemming algorithm to query terms so that, for example, FILE will retrieve documents with FILE, FILING, FILES, FILER. If not the default setting, stemming can still be activated for a given term by adding the stemming operator (the plus sign). Likewise, if stemming is the default it can be turned off for specific terms by adding the exact match operator (double quote). Truncation and wild card characters can be applied to the left or right or internally.

Personal Librarian offers a thesaurus option that allows for the substitution of a list of synonymous terms for the one entered with the thesaurus indicator character (@). This requires that thesaurus entries have been made first. For example, the use may make the following thesaurus entry:

car automobile auto van sedan vehicle motorcar car

A search for car@ will then find documents with any of the words in the list of synonyms. Only single word synonyms are allowed. Searching a word with the thesaurus character will result in the word being dropped from the query if it is not in the thesaurus (car@ or bus@ reduces to car@. It would seem more reasonable simply to drop the thesaurus character and search for the word (car@ or bus@ reduced to car@ or bus). Thesaurus expansion only works for entry terms in the thesaurus. Searching for auto@ will not work with the above thesaurus entry.

For a small number of documents the co-occurrence (expand) feature gives strange results. In our test database, the words vocabulary and control occurred together more often than not. Expanding one didn't bring up the other. If the file is too small - in number of documents, not size of file - the expand feature doesn't work at all. The system simply does nothing, no indication that there is a problem. Adding more documents got this feature to work, but more are required before the results are useful. With larger databases this expand feature works well and affords a type of topic or concept retrieval that has the advantage of not requiring any pre-processing on the part of the database administrator. Even more impressive, a document can be expanded, thereby listing (or searching) words that frequently co-occur with the words in the document that was expanded. Still more mind-boggling is the possibility of expanding on a query. This takes the words from the retrieved set generated by the query and creates a list of words that frequently co-occur with them.

Four types of hypertext links are supported (in the Windows version):

1. links to other documents in the

database 2. links to images 3. links that execute searches 4. links to external actions (e.g.,

executing external programs).

Hypertext links are implemented in three ways: imbedded in text, as hypertext link fields that are displayed from the Links pull down menu, and as index entries. All hypertext links are built into the documents before loading them into the database.

Multiple databases can be searched at one time, with search results merged into one ranked set. Other query options in the Windows version are forms searches, table of contents searches, and index searches. Forms searching requires that the database administrator create screens with fields to be filled in for the database. This serves to remind the searcher of the fields available for searching and provides a convenient way to do field specific searches. Index searching uses a prepared list of subjects. In index searching a window of subject words and phrases is presented and entries can be selected for retrieval or other actions. Entries in the index may have hypertext links associated with them, which may retrieve documents or may initiate other activities. Table of contents searching similarly uses hypertext links to provide access to groups of documents in a database.

Output

The initial response to a search can be set for either a bar chart showing the ranking of the retrieval set, or a brief listing of the first several hits in ranked order. In either case, the full records can then be called up and browsed, scrolling, paging, or jumping to query terms. Documents can be written to a disk file or sent to the printer. In database definition the fields to output are specified. This can be altered during searching using the SET command.

Sorting can be done on one or more fields, ascending or descending. Sorting is done on the entire retrieved set. If you have a set of 50 documents and want to sort the 10 most relevant by author, you first will need to write them to a disk file and sort them with another package.

User Interface

We looked at both the DOS and Windows version of Personal Librarian. Both are easy to use. The Windows version is one of the few Windows applications that we have found that actually makes Windows enjoyable to use. The Windows version also offers more options for searching. We did experience some "Unrecoverable Application Errors," which are probably more the fault of Windows than Windows Personal Librarian. With Windows 3.1 this is supposed to occur less often.

Personal Librarian software is being used for CD-ROM applications and the Windows version is well suited to this kind of use. For the searcher the interface is easy to use. The database administration functions could be streamlined a bit. This is a complaint that applies to all the packages in this review and is probably a fact of life when dealing with externally generated text files. It is not easy

to accommodate the variety of sources of text files in a well integrated interface. Many packages however do offer the option of keyboard entry and editing. The ability to annotate documents and to create hypertext links in existing databases would be another welcome feature.

Documentation

Documentation consists of a Database Administration Manual and a user manual for the DOS version and one for the Windows version. The Windows version User Manual has a brief (1 1/2 page) index. The other manuals have no index at all! The DOS version User Manual is included as a text file. We were occasionally frustrated with the documentation. Detailed indexes are needed in all the manuals. Even having the manuals online doesn't take the place of the printed manual with a good index. There is the increased browsability of the paper manual plus the added accessibility provided by the kind of conceptual analysis done by a skilled human indexer. (This is a telling comment about the state of text retrieval.)

A help command gives brief explanations of the commands; more information on specific commands can be requested. HELP messages are customizable. new HELP screens can be added and existing ones can be changed.

General Comments

The search methodologies in Personal Librarian are well suited to ad hoc searching by end-users. Meaningful results can be obtained without knowledge of Boolean logic. At the same time, all the standard features, such as Boolean logic, proximity, truncation, are there for those who know about them. Its powerful search capabilities do not require intellectual effort on the part of those creating the database. The Windows version is used as a search engine for CD-ROM applications and lends itself well to this environment.

TOPIC

Topic is available for a wide variety of systems including DOS, OS/2, Macintosh, VMS, and UNIX. It is primarily being marketed for multiuser systems and normally is sold with several days of training. The fact that we were able to get it up and running and to load our test data and create topics for our data is an indication of good documentation and overall design.

Topic's most noteworthy feature is its topic search capability. Topics are essentially groups of related words arranged into a tree structure with weights assigned to the words and relationships indicated between and among. The presence of some or all of the words in a document are a measure of the degree to which the document is about the topic of concept. Once topics are defined the searcher can retrieve ranked output without having to know Boolean logic.

Maintenance

Getting Topic installed and a Topic database up and working is not a trivial task. Several files must be created with the use of an ASCII text editor to define the database. The Topic software provides little help with this - it gives a cryptic error message when you do it wrong and try to use the files. Once the database definition files are in place, text files are loaded by running a sequence of utility programs that take care of indexing, keeping track of file locations, etc. All of this requires knowledge of the operating system and is complex enough to require going back to the documentation to load more text after a few weeks of having done it. Building a Topic database involves 13 steps (two are optional), which are done at the operating system prompt by running a series of utility programs.

Text files searched by Topic can reside anywhere on the system so long as access is permitted. Support files (indexes, data dictionaries, etc.) are created in the user's database subdirectory but the text itself does not have to be loaded as a copy. Text files can be in ASCII or in any of 17 word processing formats. Individual text documents can be included from where they were created or, optionally, multiple documents can be combined in one file with a marker separating each document.

A Topic database contains one or more partitions. Each partition can have one or more text files associated with it. Partitions speed searching in databases with more than 1000 documents, accommodate different document formats, allow distribution of documents on the network, and allow searching only portions of the database.

An optional Topic Real Time module supports continual screening of incoming text, as, for example, in a newswire setup. This functions as a filter to bring text of interest to the attention of the user.

Image files can be incorporated into documents using the optional Topic HyperLink feature. Typically, image files will use their own external viewers. Image availability in a document is indicated by an icon in the text display or by a notation at the bottom of the screen display.

Retrieval Capabilities

Topic's retrieval capabilities are what make it stand out from the crowd. Topic supports three kinds of retrieval: word, topic, and Boolean. Word retrieval allows any words or stem (other than stop words) to be selected from an index display and used as a query. It is rare that one will be satisfied with single word queries, however, so this is of limited usefulness.

Topic searching depends on the existence of predefined topics, which are browsed and selected for retrieval in much the way words are used in word searches. The difference is that the topics can represent very complex relationships among text words. Normally topics are defined with a hierarchical structure. A topic will have "children," which can be words, phrases, or subtopics. The subtopics are topics that can be used alone as well. These in turn can be defined in terms of words, phrases, or further subtopics. Ultimately all topics are defined in terms of words. (Phrases are broken down to words.) At each level in a topic definition the "children" used to

define that level can be related to one another by means of several operators including the standard Boolean operators. Other operators extend Boolean logic and allow for the assignment of weights to words and computation of topic weights based on the prescribed logic.

The operators fall into three classes:

1) All present: And, All, Paragraph,

Sentence, Phrase; 2) Any present: Or, Any, Wordgroup;

and 3) "The more, the better": Accrue.

Of these, only And, Or, and Accrue children may have weights. The weighting in effect provides fuzzy Boolean logic with the additional Accrue operator for added flexibility.

While it is true that one can search a Topic database without knowing Boolean logic, the person who sets up the topics that make this possible needs to understand Boolean logic and the other operators and how to define a search effectively using these. Detailed subject knowledge will also be required to create effective topics. In effect, defining a topic is creating a "canned' search and giving it a name, much like saving a search strategy in DIALOG, the difference being the ability to create hierarchies of topics to map knowledge structures more systematically, and the added power of the term weighting and additional operators. The effectiveness of topic searching thus depends on the creation of appropriate topics. To support general searching of a wide ranging database by a large number of users with wide ranging interests will be a formidable challenge. This approach will be more likely to succeed where well defined information needs are ongoing. Topic searching is, in effect, an extension of the notion of SDI services.

Document weights are computed based on the presence of topic words, their weights, and their relationship as defined for the topics (all present, any present, the more the better). The frequency of occurrence of any word does not affect the document weight (unlike the weighting in Personal Librarian).

The Random House Thesaurus is built-in in topic edit mode to facilitate the choice of synonyms and related terms. A soundex feature presents a list of sound-alike words for possible inclusion in a topic definition.

For those searches that are not anticipated by predefined topics and are too complex for single word searching, Topic provides Boolean searching. This allows the combination of words with topics using the standard Boolean operators plus paragraph, sentence and phrase proximity operators. Whereas in the other two modes it is possible to browse the word or topic index, here the searcher must remember the appropriate word or topic and type it in the query box. Words must be in single or double quotes, otherwise they are interpreted as topics. If in single quotes, they are searched as stems, if in double quotes as words. Boolean searches can be saved as topics, renamed, edited, and incorporated into other topics.

In any of the three modes of searching, partitions making up the database can be selected/deselected so only parts of the database are searched. Also, filters can be defined based on field values or on a list of files. This will limit searches to those documents satisfying the filter criteria, A hypertext feature allows linking documents to other documents or other parts of the same document, to notes, and to graphics.

Output

In all search modes the results are shown as a table of one-line entries in ranked order (for word searches all hits have a weight of one, so no ranking is actually made here). Any of the document lines can be highlighted and the full text displayed. When looking at the full text one can browse up and down line by line, screen by screen, or one can jump to each occurrence of query terms.

Other options at the initial display of search results include asking for an explanation of the document's weight (which topic words were present in that document and how that affected the weight), print the document, write it to a disk file, or delete it. Results can be written to a disk file or sent to a printer (all of the documents, those with more than a set minimum score, single documents, or parts of documents).

User Interface

The user interface for the system administrator when creating databases or adding documents to existing databases is the DOS prompt. Defining topics with the topic editor is very easy in terms of the mechanics of interaction. For searchers, the system provides easy interaction through pull-down menus, browser lists, topic outline displays and the like. For the knowledgeable user, short commands may be used in place of the menus. These commands are included in the menu displays to facilitate learning. Boolean search mode would benefit from the ability to call up word and topic browsers for selection of query terms.

Documentation

Topic documentation consists of two manuals: a 200 plus page Database Administrator's Guide and a 300 page User's Guide, plus about 200 pages of release notes, application notes, and other supplemental or updating information mainly of interest to the database administrator. The manuals are clearly written, well illustrated and well indexed. Each has a detailed table of contents. In addition to the written documentation there is extensive online context-sensitive help.

General Comments

Topic is marketed primarily to organizations with large networked or time sharing systems with large amounts of text being added at various places on the system. In this type of context it has much to offer for effective management of these text files. The optional Real Time module seems to be an especially good application of topic searching. In this context, the dedication of database administrator to the upkeep of the system can easily be justified, and this person will have enough involvement with the system to maintain proficiency in its operation. For a stand-alone DOS application, Topic is probably overkill, though its unique features are intriguing.

## Conclusions

Each of the four packages reviewed offers unique capabilities. The built-in programming language of Concordance makes it an attractive option for those creating custom applications. The 65,000 character limit on field size may be a limitation for some applications. Concept Finder offers many unique features particularly in its multiple file linking capabilities. For the experienced MUMPS programmer this package can also be tailored for specific applications. Personal Librarian offers very flexible and powerful retrieval capabilities with ranked output and relevance feedback. The Windows version makes a good search front end for CD-ROM databases. Topic offers a unique approach to text searching, which requires considerable investment of intellectual effort in topic creation, but then offers powerful retrieval features for end-user searchers. Topic would seem to work best in contexts where the subjects being searched were fairly predictable and ongoing. The Real Time option is a particularly good application of its strengths.

With the increasing availability of high capacity storage devices and machine readable texts there is a lot of interest in text retrieval software. All of these packages are worthy of consideration. A careful analysis of your specific requirements is needed before a choice can be made.

## REFERENCES

[1] For more details see: Carol Tenopir and Gerald W. Lundeen, Managing Your Information: How to Design and Create a Textual Database on Your Microcomputer. New York: Neal-Schuman, 1988. [2] Tenopir, Carol and Gerald Lundeen, "Survey, Analysis and Evaluation Criteria of Full Text Systems," in Proceeding of the 54th American Society of Information Science Annual Meeting, Washington, DC,October 27-31, 1991 (Medford, NJ: Learned Information, Inc., 1991), pp. 363-365. [3] Perez, Ernest. "Managing Text." Databased Advisor 8 (June 1990): pp. 83-.

## THE AUTHORS

GERALD W. LUNDEEN is a Professor at the School of Library and Information Studies, University of Hawaii. He teaches in the areas of information retrieval, library automation, science and technology information sources and systems, and conservation of library materials. He has co-authored several books and he writes frequently on software for DATABASE.

CAROL TENOPIR is an Associate Professor at the School of Library and Information Studies, University of Hawaii. She is the author of several books and many articles on various aspects of databases and online searching. Her monthly column "Online Databases" has appeared in Library Journal since 1983. She writes frequently for DATABASE and ONLINE.

Communications to the authors should be addressed to Gerald W. Lundeen and/or Carol Tenopir, School of Library and Information Studies, University of Hawii at Manoa, 2550 The Mall, Honolulu, HI 96822; 808/956-5809 (GL); 808/956-5815 (CT); Fax 808/956-5835; lundeen@ uhunix bitnet; tenopir@uhunix. bitnet; Internet - lundeen@uhunix. uhcc.hawaii.edu;Internet - tenopir@uhunix.uhcc.hawaii.edu.

---

Please note: Illustration(s) are not available due to copyright restrictions.