

Lawrence Berkeley National Laboratory

Recent Work

Title

DEVELOPMENT OF AN INTEGRATED RECORD INPUT SYSTEM (IRIS) FOR DOE'S TECHNICAL INFORMATION CENTER

Permalink

<https://escholarship.org/uc/item/2xg615qq>

Author

Cerny, B.A.

Publication Date

1982-11-01

c.2



Lawrence Berkeley Laboratory

UNIVERSITY OF CALIFORNIA

Engineering & Technical Services Division

RECEIVED
LAWRENCE
BERKELEY LABORATORY
JUN 8 1983
LIBRARY AND
DOCUMENTS SECTION

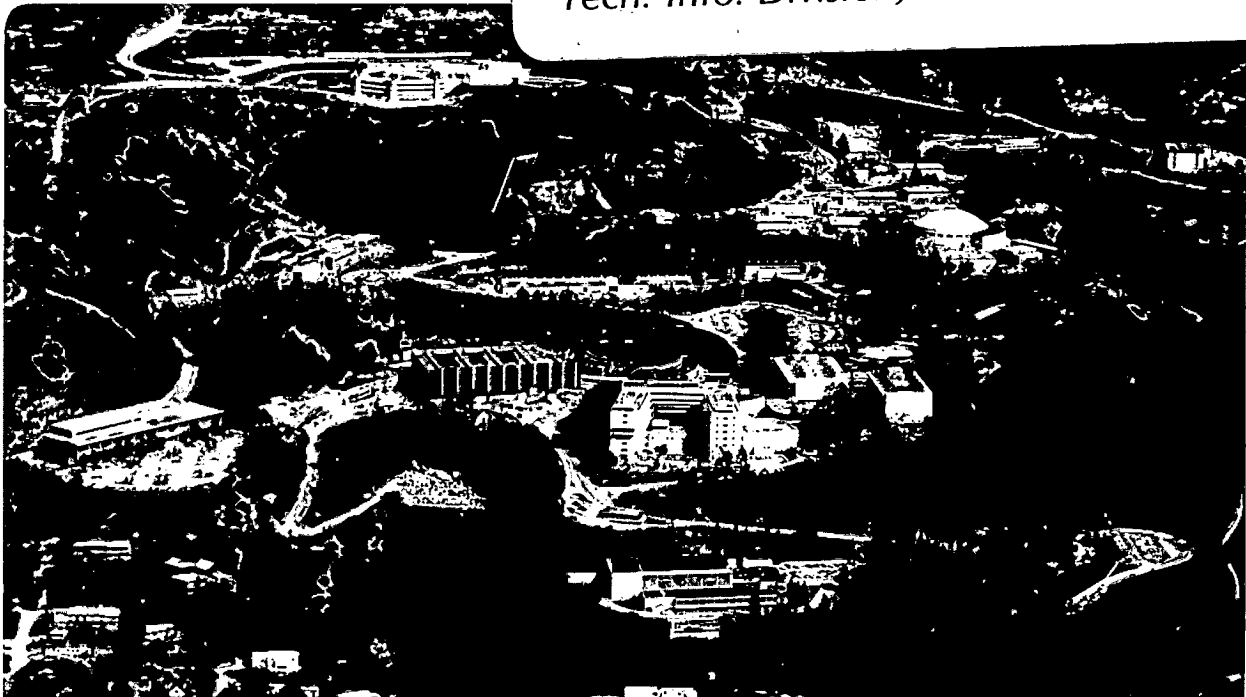
DEVELOPMENT OF AN INTEGRATED RECORD INPUT SYSTEM
(IRIS) FOR DOE'S TECHNICAL INFORMATION CENTER

Barbara A. Cerny, J. Dennis Lawrence,
Todd Hammond, and Anna M. Okseniuk

November 1982

TWO-WEEK LOAN COPY

*This is a Library Circulating Copy
which may be borrowed for two weeks.
For a personal retention copy, call
Tech. Info. Division, Ext. 6782.*



LBL-15723
c.2

DISCLAIMER

This document was prepared as an account of work sponsored by the United States Government. While this document is believed to contain correct information, neither the United States Government nor any agency thereof, nor the Regents of the University of California, nor any of their employees, makes any warranty, express or implied, or assumes any legal responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by its trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof, or the Regents of the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof or the Regents of the University of California.

DOE/TIC-11615
LBL-15723

Development of an Integrated Record Input System (IRIS)
for DOE's Technical Information Center

Barbara A. Cerny, J. Dennis Lawrence,
Todd Hammond, and Anna M. Okseniuk

Lawrence Berkeley Laboratory
University of California
Berkeley, California 94720

November 1982

This work was supported by the Director, Office of Energy Research,
Office of Basic Energy Sciences, Materials Sciences Division
of the U. S. Department of Energy under Contract No. DE-AC03-76SF00098.

ABSTRACT

This report describes the Integrated Record Input System (IRIS) that is being developed by the Lawrence Berkeley Laboratory to provide computer-assisted aids to document indexing for the Department of Energy's Technical Information Center Energy Information Data Base. Background and motivation and described, project scope and goals and given, and the software available in September 1982 in discussed in detail. An up-to-date user's manual is available as a companion report, *User's Manual for the DOE Technical Information Center Computer-Assisted Indexing System*, DOE/TIC--11616.

Acknowledgements.

Many people have contributed to the progress we have made, both at TIC and LBL. An early version of program **THSGRF** (see Section 4.3.3) was developed under the supervision of David Cahn at LBL. The RATFOR compiler we use (see Section 4.1) was produced by Joel Figen, and some of the application programming for the current software was done by Tricia Coffeen and Chris Brown. This system could not have been produced without the active cooperation and encouragement of DOE/TIC. In particular, the following people have been exceptionally helpful: Ed Coppock, Al Craig, Sue David, and Julia Redford.

CONTENTS

1. INTRODUCTION	1
2. THE NEED FOR AN INDEXING SYSTEM	2
2.1. Background — TIC	2
2.1.1. What is TIC?	2
2.1.2. The Energy Information Data Base (EDB)	3
2.1.3. TIC Processing Procedures	4
2.2. Statement of the Problem	4
2.2.1. TIC Current System	4
2.2.2. The Problem and Characteristics of a Solution	5
3. PROJECT SCOPE	9
3.1. Methodology	9
3.2. Stage 1	9
3.2.1. Indexing Behavior	10
3.2.2. The Computerized Indexing System	10
3.2.3. Morphological Analysis	11
3.3. Stage 2	12
3.3.1. Automatic Text Analysis	14
3.3.2. Statistical Analysis	14
3.3.3. Syntactic Analysis	14
3.4. Stage 3	15
4. DESCRIPTION OF THE TIC INDEXING COMPUTER SYSTEM	17
4.1. Overview	17
4.2. The File System	17
4.2.1. Areas	17
4.2.2. Master Area Files	17
4.2.3. Indexer Area Files	20
4.2.4. Supervisor Area Files	20
4.2.5. The Citation Record Structure	20
4.3. Computer Programs	20
4.3.1. Program SORT — Distribute Citations to Category Files	22
4.3.2. Program TICEDT — Edit Citation Records	27
4.3.3. Program THSGRF — Thesaurus Display	37
4.3.4. Program APPEND — Retrieving Completed Citation Files	44
4.3.5. Program ROUTE — Manual Dispersal of Citation Records	45
4.3.6. Miscellaneous Programs	45
5. THE SELECTION OF FIRST LEVEL CATEGORIES	47
5.1. Statement of the Problem	47
5.2. Words in Abstracts	47
5.3. Data Collection Phase	48
5.4. Algorithms Tested	48
5.5. Testing Results	52
5.6. The Near Future	53
6. FUTURE PLANS	58
REFERENCES	60

ILLUSTRATIONS

1. Sequence of Current Processing Steps	6
2. Sequence of Interactive Indexing	8
3. Display of a Citation Title and Abstract	13
4. The TICIS Area and File Structures	18
5. TICIS Program Structure, by Area	23
6. Example of a Citation File Table of Contents	27
7. Record Selection Menu	28
8. Example of a Record Editing Menu — Part 1	29
9. Example of a Record Editing Menu — Part 2	30
10. Example of Keyword Editing — Part 1	33
11. Example of Keyword Editing — Part 2	34
12. Example of Keyword Editing — Part 3	35
13. Example of Keyword Editing — Part 4	36
14. Example Page of EDB Thesaurus	38
15. Extract on Nucleons from EDB Thesaurus	39
16. THSGRF Display of Thesaurus Term NUCLEONS	40
17. THSGRF Display for ELECTRIC BATTERIES	41
18. THSGRF Display for ELECTRIC BATTERIES with Narrower Term Scrolling	42
19. THSGRF Display for ELECTRIC BATTERIES with Related Term Scrolling	43
20. Secondary THSGRF Display for ELECTRIC BATTERIES	43
21. THSGRF Display of All Main Terms Beginning with 'Electric'	44
22. THSGRF Display of All Main Terms Beginning with 'Electric Mo'	44
23. THSGRF Display of Thesaurus Terms Containing the Word 'Electric'	45
24. Plot of x versus $f(x,a,b,g)$ for different values of β	51
25. Plot of Test Results	56
26. Plot of Test Results	57

LIST OF TABLES

1. Citation Record Fields	21
2. TIC First Level Categories	24
3. Translating INIS Categories to EDB Categories	25
4. Possible Fields to List in Record Editing Menu	29
5. Number of Categories vs. Number of Citations	48
6. Maxima of Adhesion Coefficients for Equations (1) and (2) for Each Category	50
7. Example of Testing a Prediction Algorithm	53
8. Testing Parameters	54
9. Results of Testing	55

1. INTRODUCTION

This report will discuss the motivation for and design of an Integrated Record Input System (IRIS) to provide computer assisted aids to document indexing for the Department of Energy's Technical Information Center (TIC) Energy Information Data Base (EDB). This computer-searchable database covers all scientific and technical energy areas. It includes more than one million citations, and grows at the rate of 18,000 citations per month.

Large numbers of documents are currently processed daily with a combination of manual indexing and batch computer processing. The project described here is to develop an on-line computer system which will replace the current manual processing with a state-of-the-art interactive computer system which will allow the indexers to interact directly with a computer terminal. The programs we are developing, while drawing on automatic indexing techniques, rely heavily on human decision making, by including the indexer as a fundamental participant in the system.

The report serves two purposes: it provides a general overview of the project to interested readers, and gives a status report at the end of September, 1982. There are five chapters (other than this introductory one), describing different aspects of the project.

Chapter 2, "The Need for an Indexing System", gives background for the project. The sponsoring agency, DOE/TIC, is briefly described insofar as it interacts with the project, and the precise problem being solved is specified. Project goals are listed.

Chapter 3, "Project Scope", is an overview of the project. We describe the method of solution, divided into three stages of increasing difficulty. These stages correspond, more or less, to morphological analysis, syntactic analysis, and semantic analysis.

Chapter 4, "Description of the TIC Indexing Computer System", is the status report. It presents the computer system as it existed at the end of September, 1982. The many files used by the system are described, and each of the main programs is discussed in considerable detail. An important part of this discussion is a clear presentation of the various display screens that are presented to the user.

Chapter 5, "The Selection of First Level Categories", describes some indexing investigations we have started recently. This chapter is also a progress report.

Finally, Chapter 6, "Future Plans", briefly discusses both short-range and long-range goals for the future.

2. THE NEED FOR AN INDEXING SYSTEM

2.1. Background - TIC ¹

2.1.1. What is TIC?

The Department of Energy Technical Information Center (TIC) is the central point for collecting, processing, and disseminating scientific and technical information.

One of the primary objectives of TIC is to ensure that DOE-sponsored research is reported promptly and that reports are distributed within DOE and to its contractors and, when suitable, made available to the general public. Copies of all DOE reports issued come to TIC to be printed; distributed; cataloged, abstracted, and indexed; added to the TIC bibliographic data bases; made available for public purchase; and announced in TIC abstracting and indexing journals.

Another important TIC objective is to provide bibliographic data bases that identify the world's scientific and technical energy literature and aid the administration of DOE's programs. In meeting this objective, TIC locates and acquires energy-related scientific and technical documents or information on these documents worldwide through bilateral agreements with foreign countries, special exchange programs, organization-to-organization agreements, gifts, or purchases. All documents and information on documents acquired are carefully evaluated and analyzed to determine if the subject content is within DOE's scope of interest. Selected items then become part of DOE's science information archives and bibliographic data bases. These items are indexed and abstracted, and the indexing terms, abstracts, and bibliographic descriptions are input into computer-readable form.

TIC maintains a complete and unique publishing capability, including planning, editing, illustrating, composing, layout, printing, announcing, and distributing, for DOE prestige publications and any publication of special interest to DOE programs.

TIC provides technical reference services, document and film request services, and carries out an educational services program devoted to aiding students and teachers in their studies of energy and the general public in their understanding of energy issues.

In support of the government's international obligations, TIC supplies complete coverage of U.S. literature on nuclear energy to the International Atomic Energy Agency's International Nuclear Information System (INIS).

Scientists, linguists, editors, craftsmen, educators, writers, engineers, librarians, computer specialists, and information specialists perform TIC's strong centralized technical information activities.

TIC identifies, locates, and acquires the world's energy-related scientific and technical literature regardless of language or form and determines its validity and worth to the DOE Energy Information Data Base. Literature is generally acquired by the following methods:

1. Routine distribution to TIC by DOE contractors and some U.S. government agencies.
2. Bilateral agreements with foreign establishments.
3. Publication exchanges initiated either by TIC or by the exchange partner.
4. Gifts (books, conference proceedings, journals, and other published literature).
5. Direct purchase.
6. Directly from the publishing outlets of technical societies, such as the American Institute of Physics, and from private industry.
7. From other government agencies, such as the National Technical Information Service (NTIS) and National Aeronautics and Space Administration (NASA).

TIC becomes aware of literature items by scanning accession lists, bibliographies, documents of other abstracting services, listings of meetings and conferences, etc. Many research and development reports are identified and procured as a result of their being requested from TIC. If a requested report is not already on file, TIC procures it for the requester and then evaluates it for the

¹ Extracted from [1] and [2]

TIC data base.

The evaluation process is divided into two main streams. Research and development literature forms one stream, and published literature forms the second. All report literature retained by TIC is subject to document management and control functions that are not applied to published literature; the two processing streams facilitate the differences in handling.

For DOE contractor research and development reports, the security classification status, the patent clearance, the DOE contract number, and the distribution category must be verified.

The largest evaluation work load is journal literature; each issue of over 900 journals must be scanned to identify articles that should be added to the TIC data base. Since many of the journals are in a foreign language, the evaluator must be able to read and comprehend foreign languages. Much of the scanning is done under contract.

TIC receives and evaluates over 1,000,000 items of literature annually. Of this total only about 16% is considered to be within the scope of interest and to possess the credentials necessary for becoming a part of the TIC data base.

Much of the research and development information acquired by TIC is in standard printed report form. For effective collection, control, dissemination, and archival retention of this information, the management of documents, per se, is a necessary element in the DOE technical information system and is a major function of TIC. This function is vitally important in regard to DOE-originated reports.

The TIC document collection, numbering nearly 550,000 separate items, dates from the beginning of U.S. interest in atomic energy development and constitutes the DOE scientific archives, having significant historical value as well as current-use value. From this document reservoir, requests from DOE and its contractors and other government agencies and their contractors are filled. Since the establishment of DOE, document files have expanded to cover nonnuclear energy and are currently growing at an accelerating rate.

2.1.2. The Energy Information Data Base (EDB)

In response to the Energy Reorganization Act of 1974, TIC created and maintains on a current basis the world's largest bibliographic data base on energy. For each item entered, this computer-readable data base contains descriptive cataloging, subject category, subject indexing, and an abstract. Currently the data base contains over 1 million items, and about 215,000 new items are added each year.

The DOE Energy Information Data Base (EDB) grew out of the nuclear data base that TIC created and maintained for *Nuclear Science Abstracts*, the AEC abstract journal covering the world-wide literature on nuclear energy. When it became clear that ERDA would come into being, TIC began a major search of current and past nonnuclear energy literature to locate items to add to the data base. Therefore, when the legislation was actually enacted, TIC already had laid the foundation for its expanded data base. Although the publication of *Nuclear Science Abstracts* was discontinued in June 1976, the items that would have been published are still being identified and added to the data base along with the nonnuclear items. With the creation of DOE in October 1977, TIC further expanded the EDB subject scope to match DOE's program interests.

There are basically two means by which TIC obtains items (i.e., reports, journal articles, books, patents, dissertations, engineering drawings, conference papers, and translations) for the data base. One is to acquire the item itself and to process it into computer-readable form. The other is to acquire computer-readable bibliographic information and abstracts from outside sources and add any information unique to the TIC data base.

TIC has contracts with organizations such as the American Institute of Physics under which these organizations provide data from their data bases in a format that can be added directly to the TIC data base with a minimum of modification. TIC also has an exchange agreement with the Federal Republic of Germany and is negotiating agreements with other nations under which they will provide TIC with magnetic tapes covering their current literature in a form that can be added to the TIC data base with minor changes and additions. In addition to these arrangements, TIC

obtains tapes from other abstracting-indexing services, such as the National Technical Information Service and the International Nuclear Information System. From these tapes, TIC selects items that are within its subject scope and makes the necessary changes and additions for inclusion in the data base.

2.1.3. TIC Processing Procedures

At TIC, bibliographic data are input through an on-line, interactive, real-time computer system housed at TIC. Such a system allows on-line duplicate checking, which minimizes the possibility that an item will be entered into the system more than once. It also allows controls and information of a recurring nature to be entered into storage for recall rather than requiring that it be input by the operator each time it is needed. Bibliographic data comprise descriptive cataloging elements, abstracts, and subject indexing.

Descriptive cataloging is the process of identifying and inputting as discrete data elements each characteristic that describes a particular information item. These characteristics include title; personal author; issuing source; page count; type of literature; language; report number, availability, and price for reports; and journal name abbreviations for articles from journals. Each data element is tagged with an identifying number unique to that type of data element so that later the element can be selected or manipulated when indexes and publications are being prepared from the data base.

Abstracting involves synthesizing and structuring the essence of an information item so that a user can determine whether or not he wishes to obtain the complete document. TIC uses author-prepared abstracts when they are acceptable; when they are not, TIC prepares the abstract. Foreign-language author abstracts are translated, edited, and used.

Subject indexing is the most vital part of bibliographic processing since the primary value of the data base is in supplying on demand citations to the world's literature on a particular subject. Specially trained scientific personnel prepare the subject indexing at TIC. Subject indexing involves selecting descriptive terms from the TIC thesaurus, proposing new terms for new concepts if necessary, grouping these terms in main subject heading and modifying terms for printed indexes, and augmenting the title with free language words to add meaning to titles appearing after subject headings in the printed subject index.

Various bibliographic authorities and guides are prepared and maintained at TIC to ensure accuracy, consistency, and technical validity, e.g.,

Subject Thesaurus (TID-7000-R4)

Subject Category Authority (TID-4584-R4)

Journal Title Authority (TIC-4579)

Report Number Authority (TID-85)

Corporate Author Authority (TID-4585)

Descriptive Cataloging Manual (TID-4602)

Guide to Abstracting and Indexing (TID-4583-R1)

Each of the first five authorities is maintained in the computer for checking new data inputs.

2.2. Statement of the Problem

An understanding of the problem we are solving requires some appreciation of the current work flow within TIC. This section begins with a brief explanation of the work flow, continues with an explanation of the precise problem, and ends with solution objectives.

2.2.1. TIC Current System

TIC receives from 20,000 to 30,000 citations per month that must be processed for possible inclusion in EDB [3]. An average of 18,000 citations are actually included in the database. Approximately 70% of these citations are received on magnetic tape from other indexing

organizations, as explained above. The remaining 30% are printed documents...

The sequence of indexing steps is shown in Figure 1 [4]. Initial steps for processing tape input are quite different from those for processing printed documents. We consider tape processing first.

The citation, as received on tape, contains a variety of bibliographic detail: title, author, author affiliation, source, abstract, and subject descriptors (index terms). The tape¹ is run through a batch computer process that reformats it into TIC Citation File Format, if necessary, and performs various accuracy checks. One important check is to match the descriptors against the on-line EDB thesaurus.

The batch process prints the tape on paper, with an indication of any errors found. The printout is routed around a group of people known as Indexers. There are approximately twenty indexers, each a subject specialist. Each indexer pages through the printout, searching for citations that have not been examined by other indexers, and that fall within one of his subject specialities. Corrections are marked on the printout by handwriting. Spelling corrections are made, grammar in abstracts is corrected, and subject descriptors are generated. Some citations are deleted, because they duplicate earlier citations for a document, or do not fall within the subject scope of EDB.

When all citations have been edited, the printout is sent to a Descriptive Cataloguer, who makes further corrections (of a bibliographic nature), if necessary, and enters all changes into the batch computer system. The new file is checked for accuracy by the batch system; rejected records are "kicked-out" and cycled past the indexers and cataloguers again. This continues until no more errors are detected; the file is then ready to be entered into EDB.

Printed documents are handled somewhat differently. The document itself is routed to an indexer responsible for the subject area. He generates appropriate subject descriptors, and writes an abstract if the document has none. The result is sent to a cataloguer, who generates bibliographic data, and enters the resulting information into the computer. This is checked by the batch system, rejected records are "kicked-out", and the cycle described above is followed.

Certain interesting conclusions can be drawn from this description. Assume an average of 25,000 citations received per month, 18,000 sent on to EDB, and 16 indexers (a staff of twenty, reduced to sixteen effectives by vacations, holidays, illness, etc.). Each indexer must process 1562 citations per month, or approximately 70 per day. This yields 6-7 minutes per citation, assuming a full eight hour day with no interruptions. A more probable estimate is 5-6 minutes per citation.

2.2.2. The Problem and Characteristics of a Solution

The basic problem is to increase the effectiveness and efficiency of the indexers. In 1981, we proposed to create an on-line interactive system to support the indexers in citation processing, as shown in Figure 2 [4]. The main features and objectives of the proposed system were (and are) the following. As in all such development projects, the objectives change occasionally; the current set is commented on here. There is no particular order.

- 1) The indexer will have direct access to each citation via the interactive computer system.
- 2) Citation records are to be routed directly to the proper indexers, with a high degree of accuracy. That is, the computer system will predict the subject area of the citation with sufficient accuracy that most citations will be seen by only one indexer. Our goal is to achieve 85-90% accuracy.
- 3) The indexing program is to have the following features:
 - a) Editing will be done using a full screen editor, tailored to this specific application.
 - b) Program control is to be menu-driven.
 - c) It will be possible to display portions of the EDB thesaurus at the terminal during an editing session.

¹ or a portion of it — long tapes are broken into segments of approximately 100 citations for further processing.

SEQUENCE OF CURRENT PROCESSING STEPS **LBL**

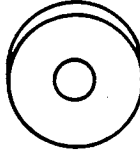


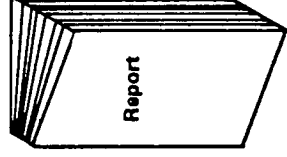
Source	Batch check	Indexer	Descriptive catalogue	Batch check	Indexer	Batch check	Descriptive catalogue	Recon tape
<p>Tape:</p>  <p>GRA EI AIP etc</p>	✓	<p>Printout</p>  <p>Manual proofing, additions, deletions</p>	Keyboard changes	✓	<p>Verification by eye and hand at rejected records</p>	✓	<p>Enter other fields</p>	
<p>Document:</p>  <p>Report</p>		<p>Manual writing of keywords, categories, (abstract)</p>	Keyboard all	✓	<p>Any further changes</p>			

Figure 1. Sequence of Current Processing Steps

- 4) Terminal response time is to be rapid enough that the indexers are not slowed down. In practice, it appears that in most cases, the response to a terminal command should begin in 1/4 to 1/2 second (or less) after the command is entered.
- 5) Errors are to be corrected when noticed. This is an automatic benefit of the full screen editor, but is an important objective in its own right.
- 6) Keystrokes are to be minimized. Most commands are to consist of a single keystroke. The actual typing of words (such as index terms) is to be avoided when possible; use of light pen, touch panel, joystick, etc. will be considered in attempting to reach this goal.
- 7) Citations are neither to become lost nor duplicated. Once a citation enters the system, it will exist in one and only one location until it "officially" leaves the system. Some method is necessary to locate any specified citation on demand.
- 8) There is to be minimal disruption of current activities. Transition to the computer system is to be smooth and gradual, with no decrease in TIC's throughput.
- 9) Management control and supervision of the indexing process is to be improved.
- 10) The computer software is to be as portable as possible, since the ultimate production computer has not yet been selected.
- 11) Production costs per citation are to be reduced.
- 12) The clerical demands on the indexers are to be reduced, and indexer decision making abilities enhanced.

In many ways, the last two objectives are the most important. Cost savings should come about by reducing the number of times a citation is examined, reducing the number of times a correction is handled, and providing instant response to most erroneous entries. These activities reduce the clerical activities required of the indexers, and thus improve their effectiveness. Interactive access to the thesaurus should further enhance this effectiveness.

It is important to note that this computer system in no way replaces indexers. Instead, the indexers must remain the vital link in processing citations for inclusion in EDB.

SEQUENCE OF INTERACTIVE INDEXING _____ **LBL**

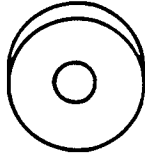
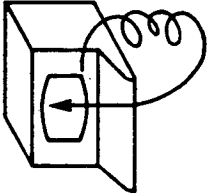



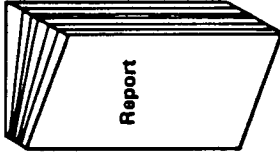
Source	Batch check	Indexer	Descriptive cataloguer	Batch process	Indexer	Descriptive cataloguer	Batch check	Indexer	Batch check	Descriptive cataloguer	RECON tape
<p>Tape:</p>  <p>GPA EI AIP etc</p>				✓			✓	 <ul style="list-style-type: none"> - Screen editor for text correction - Keyword suggestions - Category suggestions - Light pen entry 	✓	<p>Enter other fields</p>  <p>Any further changes</p> 	
<p>Document:</p> 			<p>Enter title, (abstract), keywords other fields</p>	✓			✓				

Figure 2. Sequence of Interactive Indexing

3. PROJECT SCOPE

3.1. Methodology ¹

The design of the Integrated Record Input System (IRIS) encompasses techniques from computer science (CS), information retrieval (IR), computational linguistics (CL), and artificial intelligence (AI). Separating these overlapping disciplines in this way allows a conceptualization of the man-machine boundary that permits system development simultaneously on several fronts. That is, an interactive indexing-editing system can be built in parallel with research on how to build "intelligence" into it. In the broadest view, the ultimate goal for any IR system has to be the effective retrieval of document citations for the end user, and the "intelligent" features under consideration will lead towards more precise indexing, better communication between indexer and user, and ultimately more effective retrieval. In the short term, however, there is the practical consideration of eliminating a backlog of work in a cost effective manner.

From a theoretical perspective, Smith [5] applies O'Connell's [6] three stages of evolution of a technology to the development of a retrieval system; this breakdown is equally applicable to the development of computer assisted aids to indexing:

The first stage in the evolution of technologies is one in which what is being done now can be done cheaper, faster and better with the help of technology than without.

The second stage occurs when we can do things to match the new capability that the new technology gives us.

The third stage occurs when we change our behavior and our ways of doing things to match the new capability that the new technology gives us.

This breakdown also roughly parallels our use of tools from CS, IR, CL, and AI. Thus far, we have worked primarily in the first stage with standard computer science techniques. We are emulating current indexing behavior with a menu-driven system of displays to be used in a browsing mode by the indexer. He can explore text, thesaurus and inverted index displays via terminal and make intelligent choices about document content just as he would have done manually.

Secondary indexes reflect the second stage. Using IR and CL techniques to analyze the text of title and abstracts, subject descriptors (keywords) and categories will be suggested to the indexer. This work relies on theory and experimental evidence from Salton [7], van Rijsbergen [8], Sparck-Jones [9], Stiles [10], and others on automatic statistical and linguistic text analysis.

The third stage represents the use of CL and AI techniques. Although there is superficially some overlap between IR and AI, the philosophical slants of IR and AI vary sufficiently to lead to diverse methodologies, particularly if we follow van Rijsbergen's lead in using Lancaster's definition: "[an IR system] does not inform (ie, change the knowledge of) the user of the subject of his inquiry. It merely informs on the existence (or non-existence) and whereabouts of documents relating to his request". This definition excludes most work in AI such as query systems, pattern recognition, representation, etc., which will be the heart of stage 3 work. This third stage is first evaluative and then prescriptive. Does the suggestion of terms lead to greater consistency across indexers or does it just bias them? And does this consistency or bias lead to better retrieval from the RECON user's perspective? Can the thesaurus be enriched by these suggested terms? How much of the human component is efficient and how much can machine-aided indexing (MAI) assist? As words are extracted, if they are added to the thesaurus keywords, are they more effective in document recall performance than the thesaurus alone?

These stages are examined in more detail below.

3.2. Stage 1

Although the bulk of work at this stage involves system design, including the writing and implementation of computer programs, the first step was a consideration of the indexing process

¹ Extracted, with modifications, from [4].

itself. What does an indexer actually do, both intellectually and physically? We will describe this next, drawing on both published research and protocols taken from the TIC indexers.

3.2.1. Indexing Behavior

The primary function of indexing is to create representations of document content in the indexing language. This representation is accomplished in the Energy Data Base by assigning two elements to each citation:

- 1) One or more of 861 subject categories [11].
- 2) Any number of subject descriptors from the EDB controlled vocabulary [12].

What constitutes "good" indexing is universally expressed in the literature as that which promotes effective retrieval (eg, Cooper [13]). The intellectual effort involved in this procedure is summarized by Digger [14]. Paraphrasing his breakdown and adding the steps relevant to TIC's system gives the following:

- 1) Scanning the text (or abstract), title, author, and other cataloging material.
- 2) Assessing the nature of the document.
- 3) Identifying the concepts.
- 4) Relating the concepts to user requirements.
- 5) Generating an abstract, if necessary.
- 6) Generating a title augmentation, if necessary, to replace the title with one that has information content, or to allow retrieval when a descriptor pair and title are combined into a subject entry [15].
- 7) Selecting the concepts to be indexed:
 - a) Forming concepts with respect to subject categories.
 - b) Translating concepts into thesaurus descriptors.
(This includes descriptor splitting to prevent false coordination for users and descriptor flagging to identify descriptors as main headings or qualifiers [15].)
- 8) Checking the work.

Most of these steps involve decision-making. When presented with terminal displays, as opposed to hard copy, data entry sheets and a pencil, how does the indexer choose to use them? The answer to this question will provide better definitions of the indexing process, of the information that is the most influential on indexing decisions, and how this information should be best presented.

Considering first the partially indexed magnetic tapes, we can define information fields from each record as falling into two classes.

- 1) Primary fields. These are the title, abstract, subject descriptors, and category fields that are always presented to the indexer. He may edit them and request a "link map" or path through the thesaurus or its inverted index which links terms and leads to those appropriate for concept definition.
- 2) Secondary fields. These are the author, corporate author, category fields for other systems, and so forth. They can be requested to provide additional information to the indexer.

Since the indexer can choose to go through the link map in any order, we will include a log that will keep track of the requests made for each of the fields to see what the most frequently used options are. This could lead in Stage 3 to a state space representation [16] where operators map the progress of a path from a start state to a goal state.

3.2.2. The Computerized Indexing System

The process is as follows: Citation records are brought in and an output file is created that contains the changes the indexer has made to the original records. An indexer can operate in any of

five modes. (This classification is for purposes of discussion only — the indexer is aware only of one menu-driven procedure.)

- 1) Display mode. The primary or secondary fields can be displayed on the terminal screen. The choice is menu driven.
- 2) Edit mode. Any of the primary fields can be edited with a powerful screen editor that provides such features as:
 - a) positioning the cursor,
 - b) deleting characters or words preceding or following the cursor,
 - c) inserting words or characters at the cursor,
 - d) searching for a text string,
 - e) replacing the character at the cursor,
 - f) changing words or letters to upper or lower case,
 - g) moving sections of text around,
 - h) tabbing forward or backward on words,
 - i) and scrolling up or down.
- 3) Help mode. There are two forms of help — help with the editor commands and help with the main program control.
- 4) Link mode. When in the display mode, an automatic check is made of the stem of each word in the title and abstract against the thesaurus. If a match is found, the word is highlighted on the terminal screen. It is then possible for the indexer to trace associations to that word with his link map from the inverted index or from the thesaurus. If he enters the thesaurus display, terms associated with the designated term are also displayed. These associated terms are: *broader terms*, *related terms*, *narrower terms*, *definitions*, *scope notes*, *date updated*, *used for*, and *use terms*. It is possible to move through this concept space by keyboarding the next term to be examined. Eventually, a touch of a term with a lightpen (or similar device) will enter this word as the next display.

Since an abstract is a condensation of a document, the vocabulary in the title and abstract can only indicate the choice of words an author happened to use; it will not necessarily reveal all or even most of the concepts and ideas in the document. It is the indexer's problem to find his way from the natural language of the title and abstract to the allowed thesaurus terms, and the highlighting gives an entry point into this process. It was found (in a similar experiment [17] mapping text into the INIS controlled vocabulary) that after morphological analysis only 10% of the title and abstract text matched the thesaurus. Another experiment [18] showed that only 40% of the assigned descriptors contained in the text title and abstract of *Petroleum Abstracts* matched the *Exploration and Production Thesaurus* used to index that publication. Matching in neither case provided a substitute for manual indexing, but with our conception of the indexer as a component of the system, we believe that matching and highlighting will provide a significant entry point to the indexing process.
- 5) Enter mode. When a keyword is chosen for entry into the final subject descriptor list, the indexer will be able to enter it from any position on the screen by a touch of the light pen (or similar device) or by keyboarding it into the citation record.

3.2.3. Morphological Analysis

If a manual check against the thesaurus were being performed, it would be a simple matter to disregard morphological differences that do not affect meaning. Morphology, or the study of word formation, is the least debated aspect of grammatical theory (as opposed to syntax and semantics which will be our concern in stages 2 and 3), so it is the basis of our stage 1 attempt to use the system to give additional clues about directions to search in the link map. For example, in Figure 3, it

can be seen that "condensers" provides a match with the thesaurus while "condenser" does not. While a person could easily map "condenser" onto "condensers", a computer cannot, nor could it relate these forms to "condensing" and "condensed". Hence we are developing a stemming routine based on the work of Lovins [19] which will be used to stem the thesaurus, as well as the text of each document. This will have several benefits.

- 1) It will provide more matching clue words with which to build the link map.
- 2) It will allow better statistics in stage 2 since occurrences of "condenser" and "condensers" will be counted together and not as unique character strings.
- 3) This compression will reduce the size of our word files so they will require less storage space and can be searched more efficiently.
- 4) It can be used in stage 3 with phrases to form conceptual relationships of interest. For instance, "physics applications", "applications of physicists", "applied physics", "physical applications", "application of physical", etc., might be considered as instances of the same concept.

Sparck-Jones [9] reports that "stems never perform worse than word forms and sometimes perform better". The use of stemming varies, depending on the subject matter in the database and the extent to which human decision making is tolerated in the stemming process.

The EDB database is, of course, devoted to all aspects of energy — technological, economic, social, and political. This has a definite impact on what stems must be recognized, and Lovins' lists of suffixes will be expanded and contracted accordingly. For example, "acetoacetates" and "acetoacetic" should be considered as equivalent by a stemming routine — as, indeed, they are by Lovins' suffix list. However, "autoradiography" and "autoradiolysis" should also have the same stem, but -ography and -olysis are not present in Lovins' lists. We will modify the list to better reflect the contents of the EDB database. We are also considering creating an algorithm for removing certain prefixes. As Lovins' algorithm works quite well, such a prefix-stripping algorithm will be patterned after the suffix algorithm. Only a few prefixes will be candidates for removal — "antineutron" should go with "neutron", "exoskeleton" with "skeleton", and "ultracold" with "cold". Only prefixes peculiar to science will be removed; such prefixes as de-, ex-, and un- will be left alone.

Stemming algorithms may err by identifying words that should be different (such as "aerial" and "aeration", which both stem to "aer-"), or by failing to identify words that should be equivalent. Because of the interactive nature of this system, the first of these is not particularly serious — the indexers can easily reject a suggested index term. The second type of error is more serious, since indexing terms are liable to be missed. Since decreasing one type of error tends to increase the other, we choose to err on the side of excessive identification.

Totally automated systems require near-perfection of a stemming algorithm, a standard that is rather difficult to meet. By reserving final decisions on the acceptability of indexing terms for a human indexer, we can have a very successful system with somewhat less than perfect stemming.

3.3. Stage 2

There is a parallel between the evolution of on-line searching and the development of systems for interactive indexing. As machines became capable of storing and searching large amounts of data, it became possible for users to interactively browse, formulate, and change search requests on-line and get nearly instant system response. With batch searching, the computer was made to mimic the manual search process, albeit faster and with a greater volume of documents. The development of on-line searching capabilities, and the ability to refine a search underway (eg, MEDLINE [20]), represent stage 2 in the user domain as on-line interaction of the indexer with classification and vocabularies does in the indexer domain.

Additions to Smith's basic model [5] of an IR system shifts the role of the indexer from a creator of the retrieval file to a user as well. Such an interactive system gives the indexer access to collection statistics, word weights, on-line thesauri, etc.

Title: Gerdien condenser instrumentation for measuring high-latitude middle atmosphere electrical parameters. Special report

Abstract: Gerdien **condensers** for measuring electrical conductivity, ion **mobility** and charge number **density** were flown in recent rocket programs to investigate the high-latitude middle atmosphere. The instruments were launched in two coordinated programs (Aurorozone I and II) at Poker Flat, **Alaska** to study the effects of auroral energetics on electrical parameters and in a solar **eclipse** rocket program at Red Lake, **Canada**. The **design** of the Gerdien condenser instrumentation for the Aurorozone II program and the solar **eclipse** program is considered. In addition, electrical parameters measured for the two auroral programs are presented and discussed. The initial results from the measurements indicate that high-latitude middle atmosphere electrical parameters are strongly influenced by the auroral energetics. (Author)

Figure 3. Display of a Citation Title and Abstract
(Words in the Inverted Index to the EDB Thesaurus are Highlighted)

3.3.1. Automatic Text Analysis

Automatic text analysis breaks down generally into statistical and linguistic approaches, the latter including morphological analysis, syntactic and semantic methods. Given the magnitude of the Energy Data Base (over 1,000,000 citations), the number of subject categories (861), and the size of the controlled vocabulary (over 25,000 main terms), statistical techniques seemed the most accessible, particularly when combined with morphological and (eventually) syntactic analysis. The basis of our work is that pioneered by Luhn [21] who used frequency counts of words in the document text to determine which words were significant in representing the document content. He compiled a list of "keywords" for each document. This technique has been refined and widely used in subsequent years for such purposes as thesaurus construction, devising measures of word associations for retrieval, and so forth.

3.3.2. Statistical Analysis

We will initially use word frequency statistics from the title and abstract of documents for the prediction of EDB subject categories. This falls into the realm of automatic classification. As van Rijsbergen [8] points out, all classification is for a special purpose; we are not concerned at this time with a "best" classification but rather with accurately fitting documents into the pre-existing EDB subject categories. Success is thus measured by the degree to which we succeed in these predictions. Most classification is eventually judged by its performance from the users perspective during retrieval, and, indeed, if the the indexer is considered as user, then the success is the degree to which the categories are predicted for him.

In terms of system design, there are two functions for this prediction:

- 1) Routing of documents to individual indexers. The 861 subject categories are hierarchically arranged with 40 first level categories, 302 second level categories, and 499 third level categories [11]. Each indexer covers the subject matter of a number of first level categories, so it should be possible to accomplish this routing with a high degree of accuracy.
- 2) A prediction of category or categories to be attached to each record for the indexer's consideration, along with the statistical weights representing our degree of belief in the prediction.

3.3.3. Syntactic Analysis

Syntactic analysis occurs in both stages 2 and 3. The simple rules that we will use to break text into two and three word phrases are a primitive form of stage 2 syntactic analysis, while natural language parsing used in computational linguistics for AI language understanding represents the other extreme. Our needs lie somewhere in the middle, due to a number of limiting factors:

- 1) The indexer is not going to be "conversing" with his documents. Hence, sophisticated AI techniques, the construction of grammatical parsers, formalized semantics and complex inference rules can be avoided.
- 2) We do not have to analyze general text. Titles and abstracts of technical documents are a special, simpler form of text with less ambiguity of meaning than the context dependent semantic information that is so difficult to handle in a newspaper or in dialogue.
- 3) We have a limited purpose system in that our goal is to produce suggested category and subject descriptors. Hence, syntactic analysis need not be very deep or profound. If we can extract simple phrases to be matched against the thesaurus or against lists of vocabulary derived from frequent usage in categories, the analysis will be a success. We suspect that the program could even miss some portion of these phrases with negligible effect on indexing success. This will obviously require further analysis, and some testing.
- 4) Again, it is the presence of the indexer as a critical component of the system that allows us to accept error rates that could not be tolerated by a completely automatic system. The discussion of this point earlier (with regard to stemming) applies here as well.

Much of the literature in MAI consists of small collections and while the results point to directions to explore, particularly in stage 3, they are not immediately applicable to our problem. An exception to this is the system of the Defense Technical Information Center (DOD/TIC) which has been operational since 1974 [22]. The document content is similar to TIC's (technical reports, journal articles, etc.) as is the size of the database. Currently 75% of their documents are automatically indexed, while 25% must be manually checked. Briefly, the basis of this system is syntactic analysis which chooses words and phrases, and checks them against dictionaries for recognition and syntactic type. This approach seems to hold promise for us and we wish to explore the feasibility of incorporating these techniques in conjunction with thesaurus terms and category assignment into our system and using DOD/TIC's computer programs, where possible, to carry this out.

3.4. Stage 3

Stages 1 and 2 emulate human indexing behavior and introduce the concept of the indexer as user of the database. Stage 3 extends these concepts by optimizing the prediction of subject descriptors (keywords) through semantic analysis, provides semi-automated thesaurus expansion for better retrieval and explores another facet of indexer as user. Much of this work falls into automatic indexing, such as the SMART system of Salton [7] or the experimental collections studied and reviewed by Sparck-Jones [9]. The emphasis in these experiments is on retrieval effectiveness and as Sparck-Jones comments, "there have been few direct controlled comparisons between manual and automatic indexing; that is, ones in which other variables are not affected". However, the ultimate role of indexing is retrieval and though we can devise tests in stages 1 and 2 that will test how well we emulate human behavior, stage 3 will ultimately move headlong into retrieval experiments. But these experiments must focus not only on the traditional relevance, recall and precision measurements but on the link between indexer and user, on the type of communication they can establish through dialogue or files. Retrieval strategies on the part of the user should be fed back to the indexer and the indexer should be able to make use of previously indexed documents. Walker [23] finds the "facilitating of effective communication between human generator and human user" to be the central problem in information science. If we now let the indexer as user refer not only to his being a user of condensed information in the database but a user from the retrieval perspective as well, then his insights on retrieval through indexing, and on query formation should give information on these processes. Walker is addressing this issue in a system he is prototyping where a major objective is to gather data about the nature of problem formation and then modify the ways in which material is organized in the files and presented to the user. Another suggestion would be to use the vocabulary assigned to a document as input for a search that retrieves similar documents in the file. These are then available to a user at search time or an indexer as part of an experiment on communication between user and indexer.

These speculations might represent a leap beyond stage 3, however, and into another dimension. If we return instead to the previous lines of thought and extend them, a framework for these suggestions is the representation issue in AI. In general, representation refers to how knowledge is organized in a system, and it covers the range from what elements are chosen for document surrogates, to how keywords are related to the text, to how to represent the content of a query and the content of the data and relate one to the other. In our system, with indexer as user, we have used a simple matching procedure between words in text and words in the thesaurus; the query is implicit. But we could go beyond this level to a more complex representation using, for example,

- 1) Synonym dictionaries to expand choices for thesaurus terms.
- 2) Semantic networks [24] as a variant of the thesaurus representation. A thesaurus is a semantic network in the sense that the relationship between terms (nodes) are arcs labeled BT, NT, RT, UF, or USE. But if each arc could now have a weight associated with it, a semantic space would be created that would give clues for indexing and retrieval of documents.

- 3) A more sophisticated morphological analysis [25] such as diagrams where pairs of consecutive characters are the attributes to be compared.

4. DESCRIPTION OF THE TIC INDEXING COMPUTER SYSTEM

4.1. Overview

The preceding two chapters provided some background on this project, and discussed short-range and long-range goals. This chapter is a status report, and describes what actually exists at the end of September, 1982.

The first section describes the file system. This indexing system relies heavily on the network of files to keep track of progress. Citations are moved from file to file while processing continues, until they are finished.

The last section describes the computer programs that have been written to manage these files, and aid TIC in editing and indexing citations. The three main programs, **SORT**, **TICEDT**, and **THSGRF** are described in some detail; other programs are described more briefly.

The programs are written in **RATFOR**, a structured preprocessor to **FORTRAN**, using an enhanced **RATFOR** processor specially developed for this project [25]. **FORTRAN** was chosen as the base language because of its availability on the computers we are using in the project, the degree of portability it provides, and its flexibility. The features available in **RATFOR** increase both portability and flexibility, as well as significantly improving programmer efficiency.

It is important to note that the indexing system is required to fit into an existing document processing system. This has imposed many constraints on project design and development, preempting some decisions that might have been different in another environment. These constraints have not been a serious impediment to the project.

4.2. The File System

In this section, we consider a static view of the computer system, known as **IRIS** (Integrated Record Input System). **IRIS** is file based, so a static view can be obtained by examining the files, and the paths by which citations can move from one file to another. See Figure 4.

4.2.1. Areas

From a file management perspective, there are three phases in processing citations from input tapes or documents to the generation of a **RECON** tape. A citation may be assigned to a particular indexer, to a particular supervisor, or to no one in particular. Citations can easily move from one phase to another.

Coinciding with these phases, the various computer files are divided among three types of "area": indexer areas, supervisor areas, and a master area.

When a citation first enters **IRIS**, it is placed in a file in the master area. Sooner or later, an indexer will want to edit that citation; it is now moved to that indexer's area. When the indexer has finished with the citation, it is moved back to the master area. If a supervisor wishes to examine the citation, it is moved to his area, and eventually back to the master area. From this perspective, **IRIS** consists of a set of loops, from master area to indexer area and back, and from master area to supervisor area and back. A citation may travel around these loops many times before it leaves the system; however, most citations will only go around once.

There is just one master area. There is one indexer area for each indexer, and one supervisor area for each supervisor. In the **DEC 10** implementation, each area coincides with a **DEC 10** project/programmer area.

4.2.2. Master Area Files

The master area holds input files containing citation records that are being introduced into **IRIS**, working files containing citations that are in process, the location file that contains the current location of every citation known to **IRIS**, and a number of authority files containing tables of various kinds.

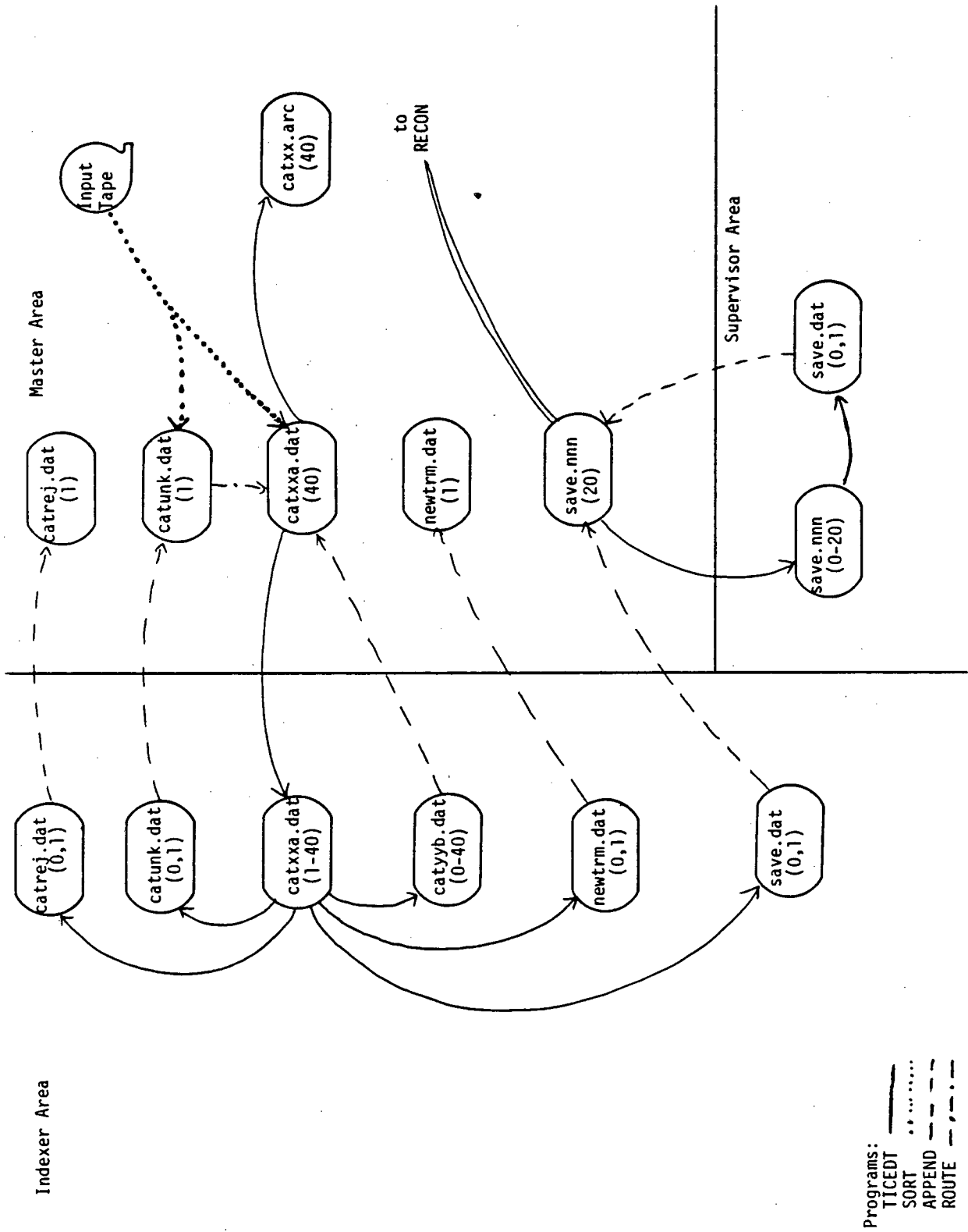


Figure 4. The IRIS Area and File Structures

Input Files

Input files are to be created by the TIC batch computer programs, much as is being done now. They generally contain 100 or fewer records, taken from a single input source. There may be any number of such files in the master area at any given point in time.

Working Files

There are over 100 working files, containing citation records-in-progress. There is one category file for each of the forty subject categories, called CATxx.DAT, where 'xx' is the category number. Citation records from many different sources may occur in a category file. A category file may contain records taken directly from an input file, records processed by one or more indexers, and records processed by both indexers and supervisors, all mixed together. Any particular category file may be empty, or have a potentially unlimited number of records.

There is an archive file for each of the forty subject categories. A citation record is placed in an archive file each time it is moved from the master area to an indexer or supervisor area. Thus, the archive files contain "snapshots" of the citation records as they move around the system, taken as they pass through the master area.

File CATUNK.DAT contains citation records that haven't been classified into one of the subject categories. Citations end up here whenever the classification program, **SORT**, has insufficient information to determine category number, or when an indexer has determined that the record is improperly classified, but doesn't know the proper classification. This file is normally expected to be fairly small.

File CATREJ.DAT contains citation records that have been rejected by an indexer or supervisor. Records are rejected because they duplicate citations previously processed, or because they do not pertain to EDB's subject area. Rejection is not automatic — this file provides a "buffer area" to allow for review of rejection decisions. As about 40% of the citation records that enter the system are ultimately rejected, this file may be quite large.

File NEWTRM.DAT contains citation records that contain potentially new thesaurus terms. Whenever an indexer or supervisor believes a citation requires a subject descriptor that does not presently occur in the thesaurus, it is placed in a special field in the record, and the record is routed here for review by the thesaurus authorities. This is an infrequent occurrence, so this file is generally empty or quite small.

Finally, there is a save file, called SAVE.nnn (where 'nnn' is an indexer's initials), for each indexer. As the indexer completes each citation, and believes it is ready to be added to EDB, it is routed here for potential review by the indexer's supervisor. It will normally contain several day's work.

The Location File

The location file is an index to the entire file system. Each citation record present in a working file, in an indexer area, or in a supervisor area has an entry in the location file, giving its current location (by area number and file name), and the date and time it was placed in that file. Each program that moves citation records from one file to another updates the corresponding record in the location file to reflect the new location of the record. In a system that may have over 30,000 records moving around on the scale described here, some way to find individual records on demand is necessary. The location file is our solution to this problem.

Authority Files

Authority files contain tabular data that change rarely, if at all. One such file contains a list of the forty category codes, and would only change if TIC's category scheme changes. Another contains an inverted index to the thesaurus, and changes somewhat every time the thesaurus is modified. There are a half-dozen or so of these files, varying from very short to quite large.

4.2.3. Indexer Area Files

Each indexer area contains a set of working files, used by the indexer to edit and index citations. There are, potentially, over 100 such files per indexer; in fact, there will be fewer than half a dozen most of the time.

There is one category file, CATxxA.DAT, for each category the indexer is currently editing. These files are obtained by copying the corresponding CATxxA.DAT file from the master area, and then deleting the master area copy.

Corresponding to each category file CATxxA.DAT is a state file, STAxX.DAT. These files are used to keep track of progress in each category file, and to contain all edited versions of the citation records. By this means, editing of a category file can be extended over several terminal sessions, and it is possible to return at any time to an earlier version of a citation record for further editing. For example, if a citation record had been edited four times, each of the earlier five versions (the original plus four changes) is kept in the state file. It is possible to return to any of these versions, make additional changes, and save the result as the sixth version of the record.

An indexer may decide that a particular citation he is editing has been miscategorized. If he knows the proper category, he may redirect the record to file CATyyB.DAT, where 'yy' is the proper category number. If the proper category is not known, he may redirect the record to file CATUNK.DAT. In either case, program APPEND will eventually move the record back to the master area, and make it available for editing by some other indexer.

File CATREJ.DAT contains rejected citation records. These records are moved to the master area file CATREJ.DAT periodically. File NEWTRM.DAT contains citation records with potentially new thesaurus terms, and is also moved to the master area by APPEND.

When editing of a category file is finished, all citation records that the indexer wishes to send on to EDB are placed in file SAVE.DAT. This file is copied to SAVE.nnn by APPEND, where 'nnn' are the indexer's initials.

4.2.4. Supervisor Area Files

The file system for a supervisor area is much like that of an indexer area. Since supervisors are reviewing the work of their indexers, their source files in the master area have the form SAVE.nnn, where 'nnn' is the indexer's initials. When this is moved to the supervisor's area, it is named SAVnnn.DAT, and the state file is called STAnnn.DAT. All other working files have the same names and functions as the corresponding indexer files: CATyyB.DAT, CATUNK.DAT, CATREJ.DAT, NEWTRM.DAT, and SAVE.mmm (where 'mmm' is the supervisor's initials).

4.2.5. The Citation Record Structure

A Citation record consists of a record header and a series of fields. Each field is identified by a three digit tag, and consists of a character string. Field length may vary from one to several thousand characters. Each record contains only those fields that are relevant to the document being described; a technical report has no journal title, for example. The fields are listed in Table 1. For further information, see reference [26].

4.3. Computer Programs

In contrast to the static view of IRIS given in the previous section, we adopt a more dynamic approach here, by concentrating on the various computer programs. For each program, we discuss:

- the function of the program,
- any underlying analysis that is necessary to understand this function,
- the interaction between the user and the program,
- and any other interesting aspects of the program.

This discussion does not purport to be a user's manual, so most of the routine details of program interactions are not discussed here. See Figure 5.

Table 1. Citation Record Fields

Tag	Mnemonic	Description
000	AN	Abstract Number
010	SN	Serial Number
020	TY	Type of Item
030	CL	Classification
040	LI	Literary Indicator
060	AUA	Personal Author (A)
070	AUM	Personal Author (M)
090	TLA	Primary Title (A)
100	STA	Subtitle (A)
110	TLM	Primary Title (M)
120	STM	Subtitle (M)
130	SET	Primary Title (S)
150	RN	Primary Report Number
170	AFA	Affiliation (A)
190	AFM	Affiliation (M)
200	OTM	Original Title (M)
210	SR	Secondary Report Number
220	PN	Patent Number
230	ICC	International Classification Codes
240	CN	Contract Number
250	CO	CODEN
260	JT	Original Title (S) (Journal Title)
300	AS	Assignee
310	CY	City of Publication
320	PB	Name of Publisher
340	JV	Volume
350	JI	Issue
360	JP	Pages (S)
370	DA	Date
380	PD	Filed Data (Patents)
390	PGM	Pages (M)
410	TN	Translation Note
420	LA	Language
430	AV	Availability — Price
440	DN	Drop Note
450	CT	Conference Title
460	CP	Conference Place
470	CD	Conference Date
480		Report Number Guidelines
490	TS	Thesis Statement

Table 1 (cont). Citation Record Fields		
Tag	Mnemonic	Description
500	FGC	Contract Code
510	DI	Distribution Code
520	RO	Report Origin
530	SF	File Selected For
540	SC	Subject Categories
550	BI	Source of Bibliographic Information
560	CF	Country of Affiliation
570	CPC	Country of Publication
580	STC	State Code
590	TRN	INIS Temporary Record Number
600	IT	INIS Type
610	IC	INIS Categories
620	AUG	Title Augmentation
640		NTIS Note
670		Original Bibliographic Source
700	CCM	Corporate Code (M)
710	CM	Corporate Author (M)
781		Record History
782		Suggested New Thesaurus Terms
801-899		Subject Descriptors
950		Abstract

4.3.1. Program SORT - Distribute Citations to Category Files

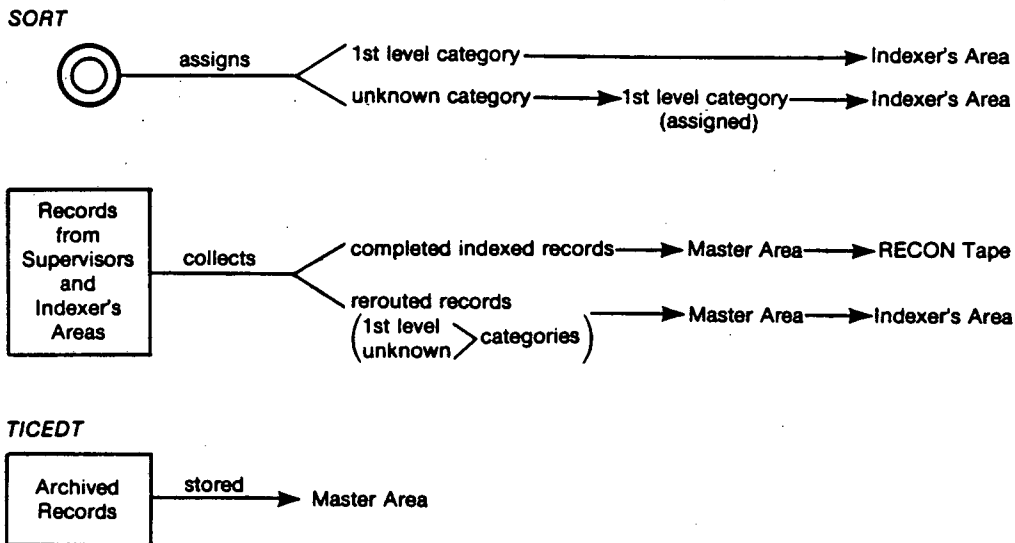
This program is the connection between the batch computer system that prepares input files and IRIS. It will read through a designated input file, and distribute the records among the forty category files, CATxxA.DAT, and the "unknown category" file, CATUNK.DAT, in the master area. Goals of the program are as follows:

- Accuracy. **SORT** should properly predict subject categories most of the time. We take a pragmatic definition of "properly" by defining it to mean that the indexers do not route the citation to another category. We hope for a success rate in excess of 90%.
- Robustness. **SORT** must be protected against computer failure. If the computer should go down while **SORT** is running, it will be necessary to restart with no loss of data and no duplication of data.
- Efficiency. **SORT** should operate efficiently, provided that the first goals are not affected.

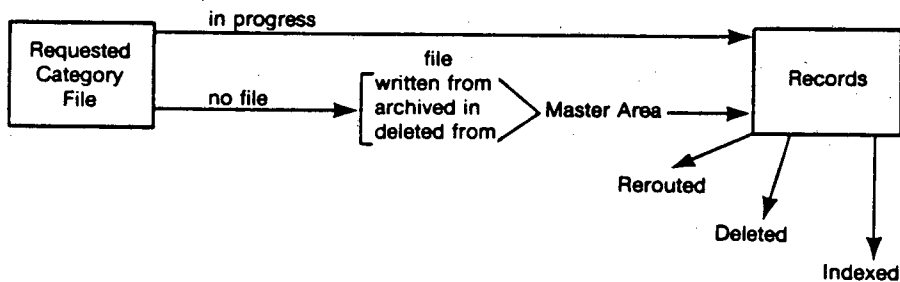
Subject Categories

TIC has divided up the World of Energy using a hierarchical scheme with three levels, as was explained earlier. There are 40 first level categories, 302 second level categories, and 499 third level categories. The first level is used to divide up indexing responsibility, and is therefore of prime concern to **SORT**. These forty categories are shown in Table 2 [11].

MASTER AREA



INDEXER AREA



SUPERVISOR AREA

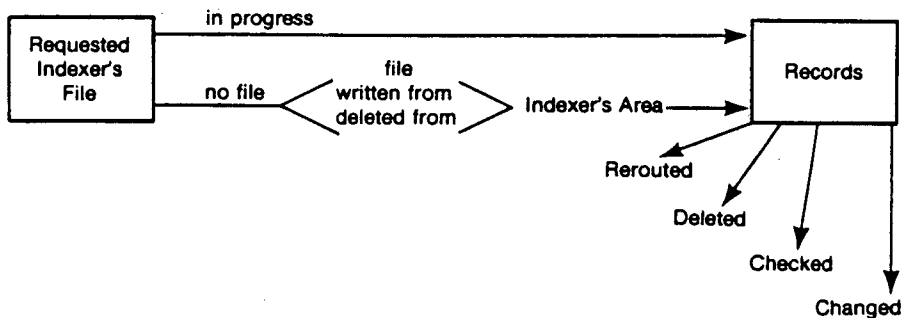


Figure 5. IRIS Program Structure, by Area

Each citation is assigned to one or more of the 861 categories. Most citations have only one category; the average is 1.3. Some citations, such as national laboratory annual reviews, have many categories assigned.

Category assignment is made by document purpose and content. When these are the same, or nearly the same, one category is assigned. If neither purpose nor content fits into a single category, multiple assignments are used.

Table 2. TIC First Level Categories

Category Number	Description
01	Coal and Coal Products
02	Petroleum
03	Natural Gas
04	Oil Shales and Tar Sands
05	Nuclear Fuels
06	Fusion Fuels
07	Isotope and Radiation Source Technology
08	Hydrogen
09	Other Synthetic and Natural Fuels
13	Hydro Energy
14	Solar Energy
15	Geothermal Energy
16	Tidal Power
17	Wind Energy
20	Electric Power Engineering
21	Nuclear Power Plants
22	Nuclear Reactor Technology
25	Energy Storage
29	Energy Management and Policy
30	Energy Conversion
32	Energy Conservation, Consumption, and Utilization
33	Advanced Automotive Propulsion Systems
36	Materials
40	Chemistry
42	Engineering
43	Particle Accelerators
44	Instrumentation
45	Explosions and Explosives
50	Environmental Sciences, Atmospheric
51	Environmental Sciences, Terrestrial
52	Environmental Sciences, Aquatic
53	Environmental-Social Aspects of Energy Technologies
55	Biomedical Sciences, Basic Studies
56	Biomedical Sciences, Applied Studies
57	Health and Safety
58	Geosciences
64	Physics Research
65	Nuclear Physics
70	Fusion Energy
99	General and Miscellaneous

Categories differ significantly from subject descriptors. The descriptors are much more specific of the document contents than categories, and subject descriptors reflect the content of the document, not its purpose. Thus, both are necessary to fully describe the document.

Automatic Category Assignment

SORT must use the information available in the citation to predict one first level category. This is to be done in one of two ways:

1. Information may exist in the citation that allows direct assignment. If an EDB category has already been assigned, it can be used and citation routing is trivial. INIS citations include a category known as the INIS category [27]. A mapping exists from the 131 INIS categories to EDB first level categories (see Table 3 for a sample), so most INIS citations can be routed quite easily.
2. If direct assignment is not possible, then other information in the citation must be analyzed and a category number deduced.

At the time of writing, only the first method is available in **SORT**. Some preliminary work has been done in analyzing a citation; this is discussed in a later chapter.

Recovery

Citation records may be added to category files in the master area by a number of programs: **SORT**, **APPEND**, **ROUTE**, and **SELECT**. Records are moved out of category files by **TICEDT**, **ROUTE**, and **SELECT**. Additional programs may be added on either end in the future. Recovery schemes for all these programs must be coordinated.

The problems that must be considered are lost records, duplicated records, and partially copied records. The general solution has the following elements:

- 1) Each program that adds records to a category file has a state file, called STATE.nnn, where 'nnn' represents the program name (SRT for **SORT**, APP for **APPEND**, etc.). This file is used to keep track of progress; in case the program fails to complete for any reason, the state file will enable the program to restart close to where it crashed. The state file is deleted upon normal termination.
- 2) When started, each of these programs first checks to see if its state file exists. If it does, recovery is automatically initiated — the program cannot be used for any other purpose until the previous task is successfully completed. Both input and output files are repositioned to their approximate positions when the crash occurred, by using the state file, and then to their exact positions by direct comparison. The program then runs to completion (or crashes again).
- 3) Each of these programs, plus **TICEDT**, checks to see if any other state file exists. If so, the program quits — it cannot be invoked until the crashed program completes. This rule prevents a file from having partially written records in the middle, and allows the interrupted program to complete knowing that the file to which it was adding records has no unexpected records at the end. While the rule may seem overrestrictive (since only one file is in doubtful condition — all others could be used), it is easy to implement, and pre-empts the question of what to do if a second program wants to write to the doubtful file after running for some time. In the case of **TICEDT**, a file is completely copied to an indexer area, and deleted from the master area. If this be a doubtful file, recovery could not proceed (since the master area file is gone), and **TICEDT** might be faced with a file with a partial record at the end.

Table 3. Translating INIS Categories to EDB Categories

INIS Category	INIS Description	EDB Category
A00	Physical Sciences	64
A10	General Physics	64
A11	Mathematical and General Theoretical Physics	64
A12	Atomic and Molecular Physics	64
A13	Solid-State and Fluid Physics	36
A14	Plasma Physics and Thermonuclear Reactions	70
A15	Astrophysics and Cosmology, Cosmic Radiation	64
A16	Direct Energy Conversion	30
A17	Low-Temperature Physics	64
A20	High-Energy Physics	64
A21	Elementary Particles (Theory)	64
A22	Elementary Particles (Experimental)	(none)
A30	Neutron and Nuclear Physics	65
A31	Neutron Physics	65
A32	Radiation Physics	65
A33	Nuclear Theory	65
A34	Nuclear Properties and Reactions	65
B00	Chemistry, Materials and Earth Sciences	40
B10	Chemistry	40
B11	Chemical and Isotope Analysis	40
B12	Inorganic, Organic and Physical Chemistry	40
B13	Radiochemistry and Nuclear Chemistry	40
B14	Radiation Chemistry	40
B15	Corrosion	36
B16	Fuel Processing and Reprocessing	05
B20	Materials	36
B21	Metals and Alloys (Production and Fabrication)	36
B22	Metals and Alloys (Physical Properties and Structure)	36
B23	Ceramics and Cermets	36
B24	Other Materials	36
B25	Radiation Effects on Physical Properties of Materials	36
B30	Earth Sciences	51
B31	Land	51
B32	Water	52
B33	Atmosphere	50
C00	Life Sciences	56
...
D00	Isotopes, Isotope and Radiation Applications	07
...
E00	Engineering and Technology	42
...
F00	Other Aspects of Nuclear Energy	21
...

4.3.2. Program TICEDT — Edit Citation Records

TICEDT is the heart of IRIS — all other programs exist to support TICEDT one way or another. It is used both by indexers and supervisors to edit and index citation records, in order to prepare them for entry into RECON. We discuss it mostly from the indexer's viewpoint. Program requirements were given earlier, in section 2.2.2, and will not be repeated here.

TICEDT is menu driven. There is a hierarchy of menus; frequently, a choice in one menu will lead to a new menu (with, usually, some action inbetween). The exact contents of the menus is somewhat context-dependent, so some lines appear only if certain conditions are met. This is discussed more fully below, in the individual menu item descriptions.

When the program is started, the indexer is asked which category he wishes to work with. A file of citation records for that category is made available — from the indexer's own area, if he had previously been working on that category, or from the master area, otherwise.

Citation File Table of Contents Menu

The first display is a table of contents of the file (see Figure 6 for an example), giving the title of each citation record and an indication of its disposition if it had been previously edited by the indexer. The first column gives the citation number; the second, the disposition; and the third, the title. Disposition can be changed by the indexer; only the latest disposition is displayed. They are as follows:

- @nn Record is to be rerouted to category 'nn'.
- ^ Record is to be rerouted to the "unknown category" file.
- % Record is to be rejected.
- & Record has potentially new thesaurus terms.
- ! Record is to be saved.

Record Selection Menu

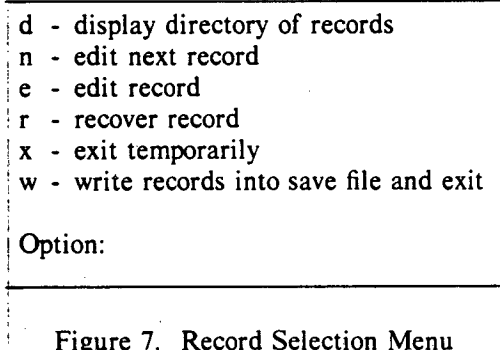
After the space bar is depressed, the screen is cleared and the menu shown in Figure 7 is displayed.

The indexer types one of the indicated letters: d, n, e, r, x, or w. TICEDT reacts as follows:

- d The Citation File Table of Contents menu is displayed, as discussed above.
- n The next unedited record is made available for editing, as discussed below. In Figure 6, this would be record 4.
- e TICEDT asks which record to edit — the indexer may choose any of them, whether they have been edited already or not. If the selected record has been edited, the latest version is made available for re-editing.

[1]	!	Arrangements for scattering electrons
[2]	@40	Angular divergence for electron beams
[3]	^	Models for superconducting cyclotrons
[4]		Kyoto University cyclotron field stability
[5]		Chromaticity improvement for KEK main ring
[6]		Magnetic field trimming effect
[Type a space to continue]		

Figure 6. Example of a Citation File Table of Contents



- r This option permits the indexer to return to a previous version of a record. **TICEDT** asks which record is to be edited. Then, the indexer is told how many versions are available, and asked which one is to be used. This version becomes the current version, and is made available for editing.
- x **TICEDT** exits. It will restart at the Citation File Table of Contents menu when invoked again for this category. This is a preferred way to terminate **TICEDT** when the citation file hasn't been completed.
- w This entry appears in the menu only if all records in the category file have been directed to one file or another. It causes the directed dispositions to actually take place, and **TICEDT** to terminate. If there are more records from this category in the master area, this fact is noted before termination.

Record Editing Menu - Part 1

After a record has been selected for editing, the Editing Menu is displayed. This menu comes in two parts — the second part is displayed only on demand. Options in the second part are invariant, and the indexer quickly becomes familiar with them, so there is no need to constantly redisplay them.

The first Editing Menu begins with the citation title at the top of the screen, with field options below (see Figure 8 for an example). Certain fields have been identified as being important to the indexer; the subset of these fields that actually occurs in this citation record is listed in the menu. Available fields are shown in Table 4.

Most of the fields are simply available for display and editing; a few require further comment.

- c Categories. Each citation record is required to have one or more categories assigned. This field must be selected before **TICEDT** will allow the record to be written to the save file. The program will check all entries for validity, by comparing the six digit category numbers entered by the indexer against a list of valid numbers.
- n Suggested new terms. If the indexer believes new subject descriptors must be added to the thesaurus to properly describe the record, they are entered here. If this occurs, the record must be routed to the new term file, NEWTRM.DAT.
- h Record history. This field can be examined, but not changed. **TICEDT** will enforce this restriction. The field contains a complete history of the record's movements from file to file, giving date and time, program name, area, and file name.
- k Keywords (subject descriptors). **TICEDT** exists to help indexers index citations, so this option is of primary importance. Great care is taken to give every aid possible to the indexer. The process is described below, under its own heading.

```

Title: Arrangements for scattering electrons

Enter a field/operation

m ----- (primary title (M))
e ----- (patent assignee)
f ----- (file selected for)
c ----- (categories)
h ----- (record history)
k ----- (keywords)
a ----- (abstract)

> ----- (operations - subordinate menu)

here:

```

Figure 8. Example of a Record Editing Menu - Part 1

Table 4. Possible Fields to List in Record Editing Menu

Letter	Tag	Description
t	090	Primary title (A)
s	100	Subtitle (A)
m	110	Primary title (M)
b	120	Subtitle (M)
p	130	Primary title (S)
u	170	Affiliation (A)
l	190	Affiliation (M)
o	200	Original title (M)
e	300	Assignee
f	530	File selected for
c	540	Categories
g	620	Title augmentation
n	780	Suggested new terms
h	781	Record history
k	801	Keywords
a	950	Abstract

Record Editing Menu - Part 2

If the ">" option is selected in Part 1 of the Record Editing Menu, this one is displayed, as shown in Figure 9. You will note that many of the options given in this menu are shift keys on the terminal. On the Hazeltine 1552 terminal, the indexer has the option of pressing the key unshifted — **TICEDT** will properly interpret this. That is, '2' is interpreted as '@', '4' as '\$', etc. The various options are as follows:

- * Display entire record. The entire citation record is displayed on the display screen. The screen editor is used for this display, so that scrolling will be available; however, no updating is allowed for this option.
- + Add a new field. This option allows a new field to be added to the citation record. The program will request the three digit field tag, and then use the screen editor to allow the indexer to enter the field.

```

Enter a field/operation
* ----- (display entire record)
+ ----- (add a new field)
- ----- (delete an existing field)
# ----- (edit a field other than the above)
@ ----- (reroute the record)
^ ----- (reroute to the unknown category file)
% ----- (reject the record)
& ----- (record has potentially new thesaurus terms)
! ----- (save the edited record)
? ----- (help)
$ ----- (terminate the run)

here:

```

Figure 9. Example of a Record Editing Menu — Part 2

- Delete a field. Many of the fields may be deleted from the citation record by using this operation. **TICEDT** will request the three digit field tag, and carry out the deletion if permitted. The record history field (781), for example, cannot be deleted.
- # Edit any field. Most fields in the record may be edited by this option. It is intended to permit occasional editing of fields not mentioned in the first part of the menu, and thus will be rarely used.
- @ Reroute the record. If the indexer examines the record, and decides it does not belong in this category, he has the option of rerouting it to another category file by this operation. The program will request the two digit category number of the proper category (say, 'yy'), and mark the record to be saved in local file CATyyB.DAT. The actual record movement, of course, will not take place until all of the records in the file have been edited and the 'w' option is selected in the Record Selection Menu.
- ^ Reroute to Unknown Category. Reroute the citation record to the file containing records whose category number is unknown.
- % Reject the Record. A record may be rejected either because it duplicates a record previously indexed, or because it does not pertain to a subject area of interest to TIC. All rejected records are written to the rejected records file, CATREJ.DAT in the indexer's area. Eventually, program **APPEND** will copy this file to file CATREJ.DAT in the master area for final disposition.
- & Potential New Thesaurus Terms. Occasionally, the 24,000+ terms in the thesaurus are not sufficient to satisfactorily index a citation — particularly in rapidly changing areas of science. If an indexer believes a new thesaurus term is required, he places it in field 780, and directs the record to a special file, NEWTRM.DAT, by using this operation. This file is copied to the corresponding file in the master area by **APPEND**, from which it goes to thesaurus specialists for a decision on whether or not to place the term in the thesaurus.
- ! Save the Record. When all desired changes have been made to the citation record, and the indexer believes it is ready to go on to **RECON**, this operation is requested. The record is sent to file SAVE.DAT, and later to SAVE.nnn (where 'nnn' is the indexer's initials) in the master area. Some effort is made by **TICEDT** to be sure the record is complete, by rejecting this operation unless the category and keywords fields have been examined by the indexer.

- ? Help. Help files are available to assist in general use of the IRIS system, and in specific use of the screen editor.
- \$ Terminate the Program. This is the other preferred method of temporarily stopping **TICEDT**. When restarted, the program will begin again at the Table of Contents Menu.

Screen Editor Facilities

A full screen editor is used by **TICEDT** to edit the various fields. The editor we are currently using works only on Hazeltine 1552 or DEC VT 52 terminals — hence the restriction mentioned earlier on the terminals upon which **TICEDT** can operate. It makes use of the numeric keypad on these terminals for many single-key operations; other operations involve use of control keys. Functions available are as follows:

- Move one space in any direction. The four arrows on the numeric keypad may be used to move the cursor one space right, left, up, or down.
- Home cursor. The *home* key may be used to position the cursor to the upper left corner of the screen.
- Tabbing. The indexer may use the *tab* key to jump forwards or backwards through the text on the screen, in units of single words.
- Next line. The *return* key may be used to move the cursor to the beginning of the next line.
- Scrolling. Two keys are available to scroll forwards or backwards through the text, if there is more text in the field than will fit on a screen. This generally occurs only with the abstract and keywords fields. Scrolling is in units of eight lines.
- Searching. It is possible to search either forwards or backwards for a string of characters that matches a specified string. The indexer can also search for a word that matches a given word; in this case, it is only necessary to position the cursor at the word, and press two keys (as contrasted to string searching, which requires entering the string).
- Text replacement. The indexer may replace text by typing over it.
- Text insertion. By pressing the *period* key on the numeric keypad, it is possible to insert text. All text that is entered until the *period* key is pressed again will be inserted at the position indicated by the cursor.
- Text deletion. There are several modes of text deletion. The character under the cursor may be deleted by pressing the 4 key on the numeric keypad; the character preceding the cursor, by the 5 key; the rest of the current line, by the 2 key; the entire line by a control key, *CTRL-F*; and any block of text, by indicating the beginning and end of the block with the cursor and a pair of keys.
- Change of case. Single keys on the numeric keypad are available to change the character indicated by the cursor to upper case or lower case. It is also possible to change a block of text to all upper or all lower case with the cursor and a pair of keys.
- Text movement. A block of text can be moved to another place on the screen by indicating the beginning and end of the block, and the new position for the block. The movement may leave the original block unchanged, or delete the original block, at the option of the indexer.
- Termination. Three methods of leaving the editor are available. The normal method is to return keeping all of the changes. If the indexer decides not to make the changes after all, there is a method of exiting the editor without making the changes. Finally, it is possible to re-edit the field, discarding all changes made. Once the editor has been exited, control returns to the Record Editing Menu.

Editing of Subject Descriptors (Keywords) — background

Keyword editing can be a complex process, because of the complicated nature of the system of keywords used by TIC. The complications arise from the two methods used to structure descriptors: flagging and splits.

The flagging technique involves arranging some descriptors into a two level hierarchy of *main headings* (M) and *qualifiers* (Q). In general, main headings are used to describe the material or system investigated in the document, while the qualifiers are chosen to indicate properties, characteristics, processes, or actions relating to the main heading. There are five general rules that govern whether a M/Q pair is acceptable [15]:

- The Q term is a direct property or quality of M, or can imply an intrinsic property or quality of M.
- The Q term is a component part of M or a closely related accessory to M.
- The Q term is an action, operation, or process on, directed at, applied to, or being done for M.
- The Q term is an action by, or an involvement, interaction, or performance of M.
- The Q term is something happening or occurring in M or describes an aspect of something happening or occurring in M.

The M/Q pair is unacceptable if

- The Q term indicates the state, shape, or use of M.
- The Q term is something induced by M.

The first level of an M/Q pair may be designated as M (main heading), T (title), or A (Augmentation); the choice affects the appearance of the citation in TIC's computer generated publications. As far as **TICEDT** is concerned, the three letters are treated just alike. We will use M, below, to mean "M, T or A".

There may be several M/Q pairs in the descriptor field; to insure proper pairing, a number is added to the symbol. Then, M1 goes with the Q1's; M2 with the Q2's; etc. A given descriptor may be an M of one pair and a Q of another, or Q's of different M's. An M may be paired with more than one Q. Headings that have no qualifier are designated simply as M (or A or T) with no number attached.

There are a number of rules that govern flagging which are enforced by **TICEDT**:

- 1) A descriptor can have only one M flag.
- 2) All Q flags must have numbers to tie them to their respective M flags.
- 3) All M flags that are part of pairs must have numbers to tie them to their respective descriptors. Note that rules 2 and 3 apply even if the citation has only one M/Q pair.
- 4) M flags *must not* be numbered if the descriptor is to be used as a main heading with no qualifier.
- 5) For citations containing splits (see below), both descriptors of each M/Q pair must be in the same split. One or both descriptors of the flagged pair can be in the zero split, since the descriptors of the zero split are common to all numbered pairs.

The splitting procedure involves subdividing the set of descriptors chosen to represent the information content of a citation to prevent the potential false coordination of descriptors in an information retrieval interrogation of the descriptors after they are stored in the data base. If the citation requires a partial split (some descriptors are applicable to each of two independent concepts), these generally applicable terms should not have split indicators assigned. Instead, they are to be placed in the zero split. Similarly, no split indicators are required for documents that do not require splitting.

Citations dealing with two or more topics, such as project reports, are prime candidates for splitting. Other examples are citations treating several alloys, organic compounds, nuclear reactions, etc.

Editing of Subject Descriptors with TICEDT

The method of editing descriptors can best be explained by a series of examples. Let us suppose an indexer has requested keyword indexing for a citation that discusses hydroelectric, solar, and geothermal power. TICEDT responds with the screen display shown in Figure 10. The indexer must now decide which keywords should be marked as M's, and which as Q's. These are indicated by using the screen editor to flag the keywords on the screen with :m's and :q's, as shown in Figure 11.

If the indexer now enters CTRL-Z, the program will save the keywords in the subject descriptor field. If keywords are again requested, the M's and Q's will be sorted out and displayed as in Figure 12. Note that in Figures 10 and 11, the keywords were displayed in alphabetical order; in Figure 12, they are in numerical order. All unflagged keywords are shown first, any unnumbered M's next, and all numbered M's and Q's last.

At this point, suppose the indexer decides to use three splits — one for hydroelectric power, one for solar power, and one for geothermal power. The split numbers are entered as prefixes to each line, in the form 'n=keyword', where 'n' is the split number. If CTRL-Z and then 'k' are entered, the keywords are displayed as shown in Figure 13.

If any errors are detected by TICEDT during keyword processing, these are displayed at the top of the screen, and must be corrected before proceeding. That is, after CTRL-Z is entered, TICEDT checks for errors. If it finds any, they are displayed, and the editor is re-entered automatically so that the indexer may correct them.

Supervisor Mode

TICEDT uses an authority file, INDEX.LIS, to determine if the user is an indexer or a supervisor. Each individual is assigned to a particular area, and the authority file "knows" which areas are used for each of the two functions. (If someone is both indexer and supervisor, two areas must be assigned.)

```

** Accepted Keywords **
. Chemical Composition
. Columbia River
. Emissivity
. Environmental Effects
. Fabrication
. Geothermal Resources
. Hot Springs
. Hydroelectric Power Plants
. Intake Structures
. Measuring Methods
. Nevada
. Optical Properties
. Performance
. Pipes
. Reflectivity
. Spectrally Selective Surfaces
. Thermal Waters
. Titanium Nitrides
. Zirconium Nitrides

```

Figure 10. Example of Keyword Editing — Part 1

** Accepted Keywords **

Chemical Composition:q8
Columbia River
Emissivity:q3,q4
Environmental Effects
Fabrication
Geothermal Resources:q7
Hot Springs:t7,q6
Hydroelectric Power Plants:t1
Intake Structures:m2,q1
Measuring Methods
Nevada:a6
Optical Properties:q5
Performance:q2
Pipes
Reflectivity:q3,q4
Spectrally Selective Surfaces:t5
Thermal Waters:t8
Titanium Nitrides:m3
Zirconium Nitrides:t4

Figure 11. Example of Keyword Editing — Part 2

** Accepted Keywords **
Columbia River
Environmental Effects
Fabrication
Measuring Methods
Pipes
Hydroelectric Power Plants:T1
Intake Structures:Q1
Intake Structures:M2
Performance:Q2
Titanium Nitrides:M3
Emissivity:Q3
Reflectivity:Q3
Zirconium Nitrides:T4
Emissivity:Q4
Reflectivity:Q4
Spectrally Selective Surfaces:T5
Optical Properties:Q5
Nevada:A6
Hot Springs:Q6
Hot Springs:T7
Geothermal Resources:Q7
Thermal Waters:T8
Chemical Composition:Q8

Figure 12. Example of Keyword Editing — Part 3

Most of the operations carried out by supervisors are identical to those of the indexers. When **TICEDT** starts, it asks for an indexer's initials rather than a category number. A file of edited citation records from that indexer is made available — from the supervisor's area, if he had previously been reviewing that indexer, or from the master area otherwise. The supervisor may have partially-reviewed files in his area for several indexers at the same time, just as indexers may have several partially-edited category files in his area.

From here on, the instructions given for the indexer are followed, with a few minor changes. For example, the restriction discussed for menu item "!" does not hold for supervisors.

** Accepted Keywords **	
1	= Columbia River
1	= Environmental Effects
1	= Pipes
1	= Hydroelectric Power Plants:t1
1	1 = Intake Structures:q1
1	= Intake Structures:m2
1	1 = Performance:q2
2	= Measuring Methods
2	= Titanium Nitrides:m3
2	2 = Emissivity:q3
2	2 = Reflectivity:q3
2	= Zirconium Nitrides:t4
2	2 = Emissivity:q4
2	2 = Reflectivity:q4
2	= Spectrally Selective Surfaces:t5
2	Optical Properties:q5
3	= Nevada:a6
3	3 = Hot Springs:q6
3	= Hot Springs:t7
3	Geothermal Resources:q7
3	= Thermal Waters:t8
3	Chemical Composition:q8

Figure 13. Example of Keyword Editing — Part 4

4.3.3. Program THSGRF — Thesaurus Display

The EDB Thesaurus [12] contains over 24,000 main terms, linked together to form a network. Each main term may have any of the following relationships and attributes.

- SN Scope Note. Delimits the scope of the term.
- DEF Defines the meaning of the term, as used in the thesaurus.
- DA Dates. Gives the date upon which the term was entered into the thesaurus, and the date the entry was last modified.
- SEE Provides a link for the user from a non-preferred term to a preferred term.
- SF Seen from. Reciprocal of SEE.
- USE Provides a link for the user from a non-preferred term to a term that must be used instead. Terms with USE attributes cannot be used to index citations.
- UF Used for. Reciprocal of USE.
- BT Broader Term. Identifies main terms of broader scope than the current term.
- NT Narrower Term. Identifies main terms of narrower scope than the current term.
- RT Related Term. Identifies main terms related to the current term in ways other than broader and narrower.

For further information on thesauri in general, see [28]. An example page of the EDB thesaurus is shown in Figure 14.

Program THSGRF provides interactive searching of the thesaurus for the indexer. By merely entering a main term through the keyboard, the indexer has available on the display screen a picture of the main term with most of its relationships and attributes. For example, suppose the main term "nucleons" is typed. If we were to look up this term in the thesaurus, we would see the entry as shown in Figure 15. The corresponding display produced by THSGRF is shown in Figure 16.

By comparing Figures 15 and 16, the transformation of the one-dimensional list in the printed thesaurus to the two-dimensional arrangement on the display screen becomes apparent. Because of the limited size of the display screen, only a portion of the thesaurus entry can be displayed. Options are available for adding the undisplayed portions to the display, replacing some of the existing display.

The display is basically in three columns. The main term is displayed centered on the screen, with some of its attributes. The dates are always shown here, with a few UF terms (if the main term has any). Two levels of broader term are shown to the left — BT1 terms in mixed case (upper/lower) nearest the main term, and BT2 terms in all upper case to the far left of the column. In the example, "baryons" is broader than "nucleons", and both "fermions" and "hadrons" are more general than "baryons". Only two levels are shown; the most general term shown in Figure 15, "elementary particles", does not appear on this display.

Narrower terms are shown to the right, also with two levels. NT1 terms are in all upper case, to the left of the column (closest to the main term), and NT2 terms in mixed case to the right. Related terms are shown in the center column, spread more-or-less evenly above and below the main term.

Another example is shown in Figure 17, for main term "electric batteries". In this example, there are no BT2 terms. There are, however, four UF terms.

It is possible to scroll any of the three columns up or down, to display terms for which there was no room on the initial display. Figure 18 shows the results of scrolling the narrower terms down one frame, and Figure 19 shows the further results of scrolling the related terms down one frame. Note in the latter case that the center part of the center column has been compressed, by suppressing all of the attributes of the main term. It is possible to list all of these attributes on a display of their own, as shown in Figure 20. Note that "Electric Batteries" has dates, a *scope note*, and several *use for's*. *Definitions*, *see*, *seen from*, and *use* are also shown on this display, if appropriate to the main term.

BIOLOGICAL RADIATION EFFECTS

NT2 Radiation Burns
 NT2 Radiodermatitis
 RT Biological Stress
 RT Oxygen Enhancement Ratio
 RT Radiation Chimeras
 RT Radiation Doses
 RT Radiobiology
 RT Radioimmunology
 RT Radioinduction
 RT Radiosensitivity
 RT RBE
 RT Strand Breaks
 RT Teratogenesis

BIOLOGICAL RECOVERY [01]

UF Recovery (Biological)
 UF Restoration
 BT1 Recovery
 NT1 Biological Regeneration
 NT1 Biological Repair
 NT2 Host-Cell Reactivation
 NT2 Photoreactivation
 NT1 Healing
 NT1 Liquid Holding Recovery
 RT Homeostasis
 RT Post-Irradiation Therapy
 RT Response Modifying Factors
 RT Therapy

BIOLOGICAL REGENERATION [01]

UF Regenerating Liver
 UF Regeneration (Biological)
 BT1 Biological Recovery
 BT2 Recovery
 RT Growth
 RT Organs
 RT Tissues
 RT Viability

BIOLOGICAL REPAIR [01]

UF Dark Repair
 UF Excision Repair
 UF Repair (Biological)
 BT1 Biological Recovery
 BT2 Recovery
 BT1 Repair
 NT1 Host-Cell Reactivation
 NT1 Photoreactivation
 RT Biological Pathways
 RT LET
 RT Molecular Structure
 RT Nucleic Acids
 RT Radiation Injuries
 RT Ultrastructural Changes

Biological Research Reactor JANUS
 USE JANUS Reactor

BIOLOGICAL SHIELDING [01]

BT1 Shielding
 RT Radiation Protection

BIOLOGICAL SHIELDS [01]

BT1 Shields

BIOLOGICAL SHOCK [01]

(For all types of shock in biology and medicine.)

UF Shock (Biological)
 UF Shock (Medical)
 UF+ Traumatic Shock
 BT1 Pathological Changes
 BT2 Diseases
 RT Anaphylaxis
 RT Biological Stress
 RT Electric Shock

BIOLOGICAL STRESS [01]

UF Stress (Biological)
 RT Anoxia
 RT Biological Fatigue
 RT Biological Radiation Effects
 RT Biological Shock
 RT Chronic Exposure
 RT Exercise
 RT Fasting
 RT Hypertension
 RT Hypotension
 RT Physiology
 RT Predator-Prey Interactions

Biological Testing

USE Bioassay

BIOLOGICAL VARIABILITY [01]

UF Variability (Biological)
 NT1 Genetic Variability
 RT Biological Adaptation

BIOLOGICAL WASTES [01]

UF+ Radioactive Biological Wastes
 BT1 Biological Materials
 BT1 Wastes
 NT1 Feces
 NT1 Sweat
 NT1 Urine
 RT Excretion
 RT Liquid Wastes
 RT Organic Wastes
 RT Sewage Sludge
 RT Solid Wastes

BIOLOGY [01]

NT1 Biological Evolution
 NT1 Botany
 NT2 Geobotany
 NT1 Cytology
 NT1 Genetics
 NT1 Radiobiology
 NT1 Taxonomy
 NT1 Zoology
 RT Animals
 RT Biochemistry
 RT Biogeochemistry
 RT Biological Effects
 RT Biosphere
 RT Ecosystems
 RT Medicine
 RT Microorganisms
 RT Organs
 RT Plants
 RT Symbiosis
 RT Tissues

BIOMASS [01]

DA July 1975
 UF Standing Crop
 BT1 Renewable Energy Sources
 BT2 Energy Sources
 RT Bioconversion
 RT Biological Materials
 RT Biomass Plantations
 RT Cellulose
 RT Harvesting
 RT Hemicellulose
 RT Lignin
 RT Oleoresins
 RT Plants
 RT Renewable Resources
 RT Wood
 RT Xylans
 DEF All growing organic matter such as plants, trees, grasses, and algae.

BIOMASS PLANTATIONS

DA September 1976
 RT Agriculture
 RT Algae
 RT Biomass
 RT Crops
 RT Farms
 RT Trees
 DEF Terrestrial or marine area and plants for the growing, harvesting, and collection of energy or combined energy/food crops for conversion into fuels.

BIOMEDICAL RADIOGRAPHY [01]

UF Radiography (Biomedical)
 UF+ Angiography
 BT1 Diagnostic Techniques
 BT1 Radiology
 BT2 Nuclear Medicine
 BT3 Medicine
 NT1 Fluoroscopy
 NT1 Ionographic Imaging
 NT1 Osteodensitometry

RT CAT Scanning
 RT Contrast Media
 RT Microradiography
 RT Radiological Personnel
 RT X Radiation
 RT X-Ray Equipment
 RT X-Ray Radiography

BIOMIMETIC PROCESSES

DA August 1978
 RT Photosynthesis
 DEF A method or procedure based on or derived from a living organism by imitation or mimicry. A biomimetic process is predicated on a translation or abstraction of a process used by a living organism for a similar end.

BIOPHOTOLYSIS

DA December 1977
 SF Microbial Processes
 BT1 Bioconversion
 BT1 Photolysis
 BT2 Decomposition
 BT3 Chemical Reactions
 BT2 Photochemical Reactions
 BT3 Chemical Reactions
 RT Hydrogen Production
 RT Photosynthesis

BIOPHYSICS [01]

RT Biological Effects
 RT Compartments
 RT Molecular Biology
 RT Radiation Doses
 RT Radiation Effects
 RT Radiation Protection
 RT Radiations
 RT Radiobiology
 RT Radionuclide Kinetics

BIOPSY [01]

BT1 Diagnostic Techniques
 RT Autopsy
 RT Tissues

BIOSATELLITES [01]

BT1 Satellites
 RT Biological Effects

BIOSPHERE [01]

BT1 Environment
 RT Biology
 RT Ecosystems
 RT Nature Reserves
 RT Populations

BIOSYNTHESIS [01]

BT1 Synthesis
 RT Anabolism
 RT Biochemistry
 RT Biological Evolution
 RT Coenzymes
 RT Enzymes
 RT Metabolism
 RT Molecular Biology
 RT Photosynthesis
 RT Precursor

BIOT-SAVART LAW [01]

RT Magnetic Fields

BIOTIN [01]

UF Vitamin H
 BT1 Heterocyclic Acids
 BT2 Carboxylic Acids
 BT3 Organic Acids
 BT4 Organic Compounds
 BT2 Heterocyclic Compounds
 BT3 Organic Compounds
 BT1 Imidazoles
 BT2 Azoles
 BT3 Heterocyclic Compounds
 BT4 Organic Compounds
 BT3 Organic Nitrogen Compounds
 BT4 Organic Compounds
 BT1 Organic Sulfur Compounds
 BT2 Organic Compounds

Figure 14. Example Page of EDB Thesaurus [12, p. 80]

NUCLEONS

DA December 1, 1974	NT1 Protons
BT1 Baryons	NT2 Cosmic Protons
BT2 Fermions	NT2 Delayed Protons
BT2 Hadrons	NT2 Diprotons
BT3 Elementary Particles	NT2 Photoprotons
NT1 Neutrons	NT2 Prompt Protons
NT2 Cold Neutrons	NT2 Solar Protons
NT3 Ultracold Neutrons	NT2 Trapped Protons
NT2 Cosmic Neutrons	RT Antinucleons
NT2 Epithermal Neutrons	RT Brueckner Method
NT2 Fast Neutrons	RT Charge Independence
NT2 Fission Neutrons	RT Effective Range Theory
NT3 Delayed Neutrons	RT Hard-Core Potential
NT3 Prompt Neutrons	RT Levinger-Bethe Theory
NT2 Intermediate Neutrons	RT Massey-Mohr Equation
NT2 Photoneutrons	RT Nucleon-Deuteron Interactions
NT2 Pile Neutrons	RT Nucleon-Nucleon Potential
NT2 Polyneutrons	RT Ope Potential
NT3 Dineutrons	RT Pseudovector Coupling
NT3 Trineutrons	RT Rosenfeld Force
NT2 Resonance Neutrons	RT Signell-Marshak Potential
NT2 Slow Neutrons	RT Stapp Theory
NT2 Solar Neutrons	RT Tabakin Potential
NT2 Thermal Neutrons	RT Wolfenstein Parameters
NT1 Photonucleons	RT Yamaguchi Potential
NT2 Photoneutrons	RT Yukawa Potential

Figure 15. Extract on Nucleons from EDB Thesaurus [12]

BROADER TERMS	RELATED TERMS	NARROWER TERMS
FERMIONS Baryons HADRONS	Antinucleons Brueckner Method Charge Independence Effective Range Theory Hard-Core Potential Levinger-Bethe Theory Massey-Mohr Equation Nucleon-Deuteron Interactions ----- NUCLEONS DA: 120174;120174 ----- Nucleon-Nucleon Potential Ope Potential Pseudovector Coupling Rosenfeld Force Signell-Marchak Potential Stapp Theory	NEUTRONS Cold Neutrons Cosmic Neutrons Epithermal Neutrons Fast Neutrons Fission Neutrons Intermediate Neutrons Photoneutrons Pile Neutrons Polynutrons Resonance Neutrons Slow Neutrons Solar Neutrons Thermal Neutrons PHOTONUCLEONS Photoneutrons PROTONS Cosmic Protons Delayed Protons Diprotons

Figure 16. THSGRF Display of Thesaurus Term NUCLEONS

BROADER TERMS	RELATED TERMS	NARROWER TERMS
Electrochemical Cells	Battery Paste Battery Separators Cardiac Pacemakers Charge State Electric-Powered Vehicles -----	LEAD-ACID BATTERIES METAL-GAS BATTERIES
	ELECTRIC BATTERIES DA: 120174;120174 UF: Batteries (Electric) UF: Secondary Batteries UF: Storage Batteries UF: Voltaic Cells -----	Aluminium-Air Batteries Cadmium-Air Batteries Iron-Air Batteries Lithium-Chlorine Batteries Lithium-Water-Air Batteries Nickel-Hydrogen Batteries Silver-Hydrogen Batteries Zinc-Air Batteries Zinc-Chlorine Batteries
	Electrical Equipment Electrolytic Cells Electromotive Force Energy Storage	METAL-METAL BATTERIES METAL-METAL OXIDE BATTERIES

Figure 17. THSGRF Display for ELECTRIC BATTERIES

BROADER TERMS	RELATED TERMS	NARROWER TERMS
Electrochemical Cells	Battery Paste Battery Separators Cardiac Pacemakers Charge State Electric-Powered Vehicles	Iron-Nickel Batteries Nickel-Cadmium Batteries Nickel-Zinc Batteries Silver-Cadmium Batteries Silver-Zinc Batteries Zinc-Manganese Batteries METAL-NONMETAL BATTERIES Lithium-Copper Chloride Batteries Lithium-Sulfur Batteries Sodium-Sulphur Batteries Zinc-Bromine Batteries PRIMARY-SECONDARY HYBRID BATTERIES
	----- ELECTRIC BATTERIES DA: 120174;120174 UF: Batteries (Electric) UF: Secondary Batteries UF: Storage Batteries UF: Voltaic Cells ----- Electrical Equipment Electrolytic Cells Electromotive Force Energy Storage	

Figure 18. THSGRF Display for ELECTRIC BATTERIES with Narrower Term Scrolling

BROADER TERMS	RELATED TERMS	NARROWER TERMS
Electrochemical Cells	Energy Storage Systems Hybrid Electric-Powered Vehicles Matrix Materials <hr/> ELECTRIC BATTERIES <hr/> Off-Peak Energy Storage Primary Batteries Solid Electrolytes	Iron-Nickel Batteries Nickel-Cadmium Batteries Nickel-Zinc Batteries Silver-Cadmium Batteries Silver-Zinc Batteries Zinc-Manganese Batteries METAL-NONMETAL BATTERIES Lithium-Copper Chloride Batteries Lithium-Sulfur Batteries Sodium-Sulfur Batteries Zinc-Bromine Batteries PRIMARY-SECONDARY HYBRID BATTERIES

Figure 19. THSGRF Display for ELECTRIC BATTERIES with Related Term Scrolling

ELECTRIC BATTERIES DA: 120174;120174

SN: Devices for production and/or storage of electrical energy from chemical reactions; excludes FUEL CELLS and RADIOISOTOPE BATTERIES.

UF: Batteries (electric)
 UF: Secondary Batteries
 UF: Storage Batteries
 UF: Voltaic Cells

Figure 20. Secondary THSGRF Display for ELECTRIC BATTERIES

Two other displays are available in **THSGRF** to aid the indexer in browsing in the thesaurus. The first of these shows all main terms in the thesaurus that begin with a character string entered by the indexer. This string may be a word, a portion of a word, a word plus a portion of another, etc. For example, in Figure 21, we see the display resulting from a request to display all thesaurus terms that begin with "electric". There are more such terms than will fit on the display; the indexer has the option of continuing the list if he wishes. Figure 22 shows the display of all thesaurus terms that begin with "electric mo".

The second browsing aid is the ability to display all thesaurus terms that contain a specified word. In this case, an actual word must be specified, not a string. In Figure 23, for example, we see the first screen load of all thesaurus terms that contain the word "electric". As in the stem display shown in Figures 21 and 22, the indexer has the option of continuing on to the next display

MAIN TERMS BEGINNING WITH SPECIFIED STEM	
Electric Appliances	Electric Fields
Electric Arcs	Electric Filters
Electric Batteries	Electric Furnaces
Electric Born Model	Electric Fuses
Electric Bridges	Electric Generators
Electric Cables	Electric Grounds
Electric Charges	Electric Heating
Electric Coils	Electric Hexadecapole Transitions
Electric Condensers	Electric Impedance
Electric Conductivity	Electric Logging
Electric Conductors	Electric Measuring Instruments
Electric Contractors	Electric Moments
Electric Contacts	Electric Monopole Transitions
Electric Controllers	Electric Monopoles
Electric Currents	Electric Motors
Electric Dipole Moments	Electric Octupole Transitions
Electric Dipole Transitions	Electric Potential
Electric Dipoles	Electric Power
Electric Discharge Pumping	Electric Power Industry
Electric Discharges	Electric Power Research Institute

Figure 21. THSGRF Display of All Main Terms
Beginning with 'Electric'

MAIN TERMS BEGINNING WITH SPECIFIED STEM
Electric Moments
Electric Monopole Transitions
Electric Monopoles
Electric Motors

Figure 22. THSGRF Display of All Main Terms
Beginning with 'Electric Mo'

of terms that include "electric", or terminating this display.

4.3.4. Program APPEND — Retrieving Completed Citation Files

This program is used to sweep through all indexer and supervisor areas, examining the areas for completed citation files, and pulling any such files back to the master area. It will normally be run at the end of each working day, to finish off the day's work. **APPEND** is a batch program, running with no interaction with the user.

APPEND examines each area for files named SAVE.DAT, CATyyB.DAT (where 'yy' is a category number), CATUNK.DAT, CATREJ.DAT, and NEWTRM.DAT. Contents of any such files are appended to the corresponding files in the master area, and the files in the indexer or supervisor area are deleted. SAVE.DAT goes to SAVE.nnn, where 'nnn' is the indexer's or supervisor's initials, and CATyyB.DAT goes to CATyyA.DAT. The remaining files have the same name in all areas.

MAIN TERMS CONTAINING SPECIFIED WORD	
Batteries (Electric)	Electric Born Model
Bridges (Electric)	Electric Bridges
Cables (Electric)	Electric Cables
Chugoku Electric Power Company	Electric Charges
Reactor	Electric Coils
Coils (Electric)	Electric Condensers
Condensers (Electric)	Electric Conductivity
Conductivity (Electric)	Electric Conductors
Conductors (Electric)	Electric Contactors
Contacts (Electric)	Electric Contacts
Converters (Electric)	Electric Controllers
Currents (Electric)	Electric Currents
Discharges (Electric)	Electric Dipole Moments
Electric Appliances	Electric Dipole Transitions
Electric Arcs	Electric Dipoles
Electric Batteries	Electric Discharge Pumping

Figure 23. THSGRF Display of Thesaurus Terms
Containing the Word 'Electric'

Recovery from program or system crashes is an important attribute by **APPEND** — this was discussed above in section 4.3.1, in the description of **SORT**.

4.3.5. Program **ROUTE** — Manual Dispersal of Citation Records

It is periodically necessary to distribute the citation records that are contained in a file to category files under the complete control of the user. This happens regularly when **SORT** places records in **CATUNK.DAT**, and when an indexer or supervisor running **TICEDT** disposes of citation records to the local **CATUNK.DAT** file. There are other cases when the need can arise unexpectedly.

ROUTE is used to reroute records from any existing citation file to category files. The program asks the user for a citation file name — it may or may not be **CATUNK.DAT**. It then proceeds through the file, operating on each citation in turn. Title and abstract are displayed on the terminal screen, with all words highlighted that occur in the inverted index to the thesaurus. **ROUTE** then asks the user for a two-digit category number, and adds the record to that category file. When all citation records have been rerouted, the input file is deleted.

Recovery is as discussed in section 4.3.1.

4.3.6. Miscellaneous Programs

A number of other programs are available to carry out specific tasks, and more are being written as new requirements surface.

MUT3 is a demonstration program. It reads through a citation file, displaying title and abstract of each record on the terminal screen. Words that occur in the inverted index to the thesaurus are highlighted, and the user may obtain lists of all thesaurus terms that contain these highlighted words.

LISTMT lists main terms from the thesaurus. It asks the user for a starting place, and then lists main terms from that point on until the program is interrupted by the user, or the end of the thesaurus is reached. If the string provided by the user does not occur in the thesaurus, the first

main term that lexicographically follows that string is used as the starting point.

Two programs are needed to build the inverted index to the thesaurus. **CPIWCP** reads through the thesaurus and extracts words from the main terms. Insignificant words that occur on a stop list are deleted; all remaining words are stored in a temporary file with an indication of the main term they were used in. This file is sorted, using the system sort facility, and then read by **CPIBLD**, which compresses the list (by combining identical words) and actually builds the inverted index.

LISTPI can be used to list the words in the inverted index from some point on, with the main terms in which they occur. Like **LISTMT**, the user is asked for the starting point, and the program continues until interrupted or the end of the inverted index is reached.

BLDLOC will create a location file with empty records. Program **LOCATE** is used to report the current file locations of specified citation records, as given by the location file. Records may be specified by serial number, tape number, or both. The locations of all citation records that match the specification are displayed, with the date and time they were placed in the file.

5. THE SELECTION OF FIRST LEVEL CATEGORIES

5.1. Statement of the Problem

TIC receives citations on magnetic tape from many other organizations. A typical tape will contain citations that fit in many different categories. As discussed in earlier chapters, the forty first level categories are used to match up indexers with citations, so that an indexer will normally "see" a citation only if it concerns a subject in which he is expert.

As tapes are received, they are reformatted into TIC Citation File Structure, and placed in one or more disk files. A typical disk file will contain approximately 100 citation records. The disk files are processed by program **SORT**, which distributes the records into other files by first-level category number. The proper category for any given record can only be determined by analysis of the record contents. In some cases, this is quite easy; in others, quite difficult.

Tapes received from INIS, for example, already contain an INIS category (field 610). A direct translation exists from INIS categories to EDB first-level categories, so files created from INIS tapes are processed quite easily. Details were described earlier in Section 4.3.1.

If no such direct mapping exists, it is necessary to deduce the first level category numbers from other information in the record, by some sort of approximation algorithm. Some of the suggestions we are exploring are the following:

- Analyze the words and phrases used in the abstract, and match this usage against a profile of past word and phrase usage for the different categories. If vocabulary is sufficiently distinct among the categories, then proper assignment can be made for many documents on this basis.
- Do a similar analysis on title words and phrases. There is no *a priori* reason to believe that words and phrases are used the same way in titles and abstracts, so this analysis is separate from the first one. If analysis should show that usage is the same, these two approaches can be merged.
- Do a similar analysis on any subject descriptors supplied with the citation.
- Match authors against previous work. This is based on the belief that scientists generally do not change fields, so a new citation will generally fit into a category previously used by the author.
- A similar analysis might be possible on author affiliation, for similar reasons.
- If the citation was part of a conference, the conference title may contain additional information.

The six items listed above are in order of decreasing expected value to a matching scheme. We suspect that no one method will be satisfactory in all cases — instead, **SORT** will probably need to use several approaches. Thus, analysis of an abstract may be sufficient much of the time; adding analysis of title and descriptors may take care of most of the remainder; etc.

5.2. Words in Abstracts

So far, we have been investigating only the first of the six approaches listed above — words and phrases (collectively referred to as "words" below) in abstracts. Work on the other approaches will take place later.

Abstracts contain content-bearing words as well as non-content-bearing words. The latter include articles ("the", "a", "an"), prepositions, conjunctions, and similar words. Content-bearing words can be thought of as "hints", or "clues", to the subject matter of the citation. If, for example, "galaxy" or "quasar" occur in an abstract, there is a good chance the document is about astronomy. Possible not, though — it could be about automobiles or TV sets. A single word is insufficient to be a very strong hint. However, if the abstract contains many content-bearing words that mostly hint the same way, confidence in the hint increases. If it increases enough, we are reasonably confident in assigning a category to the citation. (In our interactive indexer-oriented system, a low level of error is tolerable, since the indexers will be reviewing the record.)

All this assumes we know what subjects a word refers to, and, if more than one subject is possible, what the probabilities of the different subjects are. A good way to do this is to examine the way words have been used in the past, and hypothesize that they will continue to be used in the same way in the future. If "galaxy", for example, occurred in 100 abstracts in the past, and referred to astronomy 90 times and autos 10 times, then when we run across the word in the future, we can be 90% confident that the citation refers to astronomy. (If we sample past citations rather than exhaustively analyze them, this percent is somewhat lower than 90% due to the possibility of sampling errors.)

5.3. Data Collection Phase

This, then, is what has been done. We have examined ten past issues of EDB monthly update tapes, containing 51,945 citations. Each citation contains an abstract and one or more category assignments, as determined by indexers. Multiple category assignments occur when the citation refers to more than one subject area.

We discarded all citations with multiple category assignments (about 25% of the total) because it is impossible to assign abstract words to unique categories in such cases. This may introduce a bias in the analysis — any such bias was ignored.

Our goal was to examine 500 citations for each of the forty first level categories — by restricting each category to 500 citations, we hoped to eliminate another source of bias. We found 500 citations for twenty of the categories. There are just not many citations in some categories — after all ten tapes were processed, category 16, for example, had only 19 citations. In three cases, a few more tapes would probably have resulted in full categories, but we quit at this point. Table 5 summarizes this phase of the study. From this table, we can see that 500 citations were found for twenty categories, but only 300-399 were found for 2 categories. In all 14,248 citations were included in the detailed analysis.

From each citation, we extracted content-bearing words and 2- and 3-word phrases. These were stored in a word dictionary, along with the category number. At the conclusion of this phase of the study, we had available a large dictionary of words, with a history of past usage for each word.

5.4. Algorithms Tested

The next phase of the study was to test various algorithms for predicting category from word usage. Additional EDB tapes (not used in the first phase) were used to provide test data.

Our goal is to automatically classify abstracts in the same way as humans do. Since many citations have been assigned to multiple categories, we wish to closely match a set of categories, not just one. This is a difficult problem. Sometimes it is almost impossible to say to which categories an abstract belongs, and, indeed, different people may predict slightly different sets of categories. In such a situation, an automatic prediction algorithm cannot be better than a manual one. The best

Table 5. Number of Categories vs. Number of Citations

Nbr. Categories	Nbr. Citations/category
20	500
3	400-499
2	300-399
6	200-299
3	100-199
6	1-99

40	

result we can obtain from the computer is to find, for any abstract, two subsets of categories:

- the set of categories to which everyone will classify the abstract;
- and the set of categories to which someone will classify the abstract.

Some different algorithms have been applied, and results of the tests using EDB tapes (other than those used to make the dictionary) have been examined and compared. This makes a base for new, improved predicting methods.

The algorithms are based mainly on the two forms of "adhesion coefficients" described in [29]. They are:

$$a_k^1 = \sum_j \frac{f_{jk}^2}{f_j \cdot f_k} \quad (1)$$

and

$$a_k^2 = \sum_j \frac{f_{jk} \log f_{jk}}{f_j \cdot \log f_k} \quad (2)$$

where

- j is the word index,
- k is the category index,
- f_{jk} is the number of occurrences of word j in category k,
- $f_j = \sum_k f_{jk}$ is the total number of occurrences of word j in all categories,
- $f_k = \sum_j f_{jk}$ is the total number of occurrences of words in category k.

Simply taking the maximum of equation (1) or (2) for each abstract to predict category, as in [29] is not sufficient for our use. Table 6 shows the maxima of these coefficients for different categories.

The second step of making predictions, after computing "adhesion coefficients" from equations (1) or (2), is to use the fuzzy classification method, based on fuzzy set theory [30-32]. This method utilizes the information about the maximum of the two coefficients for a category and some other information connected with frequency of occurrence of words, to predict categories for an abstract. Briefly, it is as follows:

We treat each category as a fuzzy set of abstracts with the membership function which is an extension of the standard S function [32,33]

$$S(x, \alpha, \beta, \gamma) = \begin{cases} 0 & \text{for } x \leq \alpha \\ 2 \frac{(x-\alpha)^2}{(\alpha-\gamma)^2} & \text{for } \alpha < x \leq \beta \\ 1 - 2 \frac{(x-\gamma)^2}{(\alpha-\gamma)^2} & \text{for } \beta < x \leq \gamma \\ 1 & \text{for } \gamma < x \end{cases} \quad (3)$$

where $\beta = \frac{1}{2}(\alpha + \gamma)$.

The extension of this equation as used in our work has the form:

$$S'(x, \alpha, \beta, \gamma) = \begin{cases} 0 & \text{for } x \leq \alpha \\ \frac{(x-\alpha)^2}{(\alpha-\beta)(\alpha-\gamma)} & \text{for } \alpha < x \leq \beta \\ \frac{1 - (x-\gamma)^2}{(\gamma-\beta)(\gamma-\alpha)} & \text{for } \beta < x \leq \gamma \\ 1 & \text{for } \gamma < x \end{cases} \quad (4)$$

Table 6. Maxima of Adhesion Coefficients for Equations (1) and (2) for Each Category

Category Number	Adhesion Coefficient (1)	Adhesion Coefficient (2)
01	258.70	79824.64
02	225.86	67544.95
03	165.78	34569.92
04	226.91	26037.08
05	240.52	85350.36
06	195.30	4602.03
07	134.21	7536.83
08	130.48	6525.12
09	150.59	11123.43
13	202.68	14284.86
14	210.22	63210.38
15	236.09	71364.53
16	156.97	938.77
17	227.56	18172.76
20	198.11	69303.18
21	189.96	59934.19
22	199.14	64700.51
25	212.46	44264.59
29	229.34	56289.64
30	164.05	23034.05
32	189.09	55405.16
33	318.07	110027.67
36	233.31	70349.80
40	199.99	57947.89
42	179.97	51790.00
43	233.53	67066.26
44	211.62	60364.38
45	206.87	31931.35
50	226.88	73643.36
51	204.78	33870.45
52	214.00	38545.47
53	85.30	1159.99
55	291.32	111630.28
56	329.47	116238.03
57	123.73	1454.60
58	237.23	74008.31
64	224.70	71587.31
65	260.63	73660.90
70	220.52	61019.40
99	197.87	20209.47

where $\beta \in [\alpha, \gamma]$ is the parameter that determines the shape of the curve; see Figure 24. For $\beta = \frac{1}{2}(\alpha + \gamma)$, equation (4) reduces to the standard form (3).

Function S' is a membership function for any category i with parameters

$$\alpha_i = 0,$$

γ_i = maximum of the 'adhesion coefficients' for this category,

β_i = an experimentally-determined parameter.

$$S'(x,a,b,g) = \begin{cases} 0 & x < a \\ \alpha(x-a)^2 & a \leq x < b \\ 1 - \beta(x-g)^2 & b \leq x < g \\ 1 & g \leq x \end{cases}$$

where $\alpha = \frac{1}{(a-b)(a-g)}$ and $\beta = \frac{1}{(g-b)(g-a)}$.

If $b = \frac{1}{2}(a+g)$, then

$$S'(x,a,b,g) = S(x,a,b,g) = \begin{cases} 0 & x < a \\ 2(x-a)^2 & a \leq x < b \\ 1 - 2(x-g)^2 & b \leq x < g \\ 1 & g \leq x \end{cases}$$

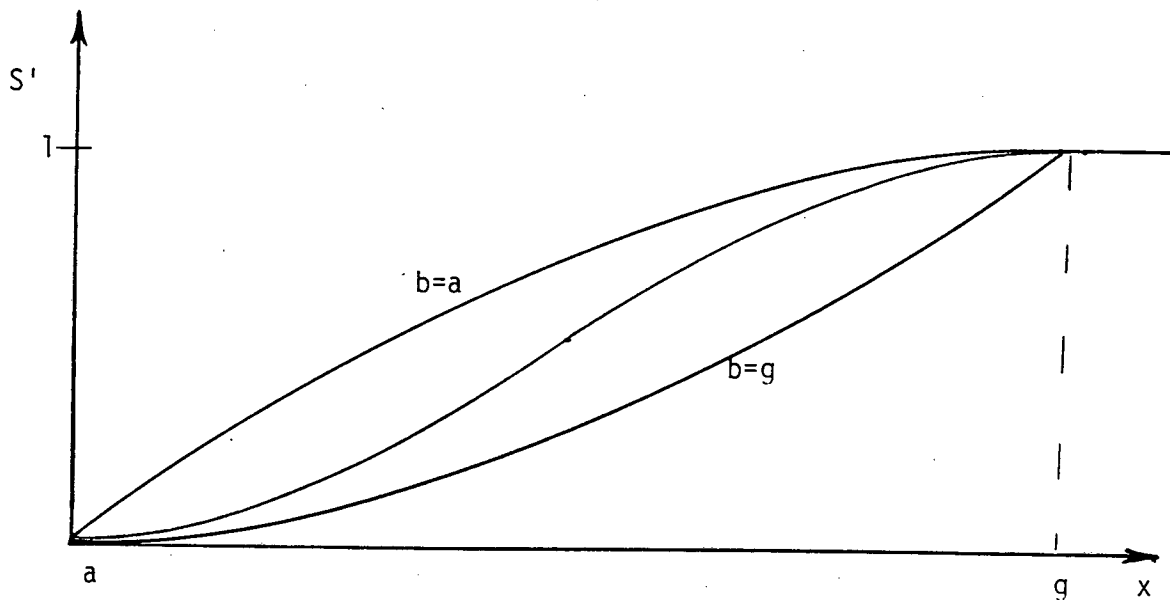


Figure 24. Plot of x versus $S'(x, \alpha, \beta, \gamma)$ for different values of β

For purposes of experimentation, we set and change β_i manually, but it will be adjusted automatically in the production system.

Parameter x is characteristic of an abstract and category. In general, it has the form

$$x_j = f_i(a_j) \quad (5)$$

where a_j is "adhesion coefficient" (1) or (2), and f_i is a function that takes into consideration such information characteristic of the abstract and dictionary as number of words, frequency of use, and so on. The membership values of an abstract to category fuzzy sets are more comparable than are "adhesion coefficients".

We classify abstracts into one of three fuzzy sets:

1. The categories the algorithm strongly predicts for a given abstract, called the 'YES' categories for that abstract.
2. The categories the algorithm weakly predicts or rejects for a given abstract, called the 'POS' (possible) categories for that abstract.
3. The categories the algorithm strongly rejects for a given abstract, called the 'NO' categories for that abstract.

The best results we can hope to obtain will be if the fuzzy set YES is equal to the set of categories everyone predicts for the abstract, fuzzy set NO is equal to the set of categories no one predicts, and fuzzy set POS is equal to the set of categories some one (but not everyone) predicts.

Tests were made for different values of parameter β_i of function S' (4), and different functions f_i (5) for a sample of 78 abstracts taken from EDB tapes not used to construct the dictionary. At one point during the testing the dictionary was changed slightly by eliminating words that occurred only once or twice. The results of each test was compared with the manual assignment of categories.

5.5. Testing Results

In interpreting the results of testing, we first assume that all manual predictions of tested abstracts are correct (in the sense that they coincide with the set of categories 'everyone' predicts mentioned in the last section). Thus, we do not divide the manual categories into certainties and possibilities.

We compare manually assigned categories with automatically assigned categories by the following six parameters:

1. The number of categories to which the abstract was classified both manually and by the algorithm (labelled '+YES').
2. The number of categories to which the abstract was classified by the algorithm but not manually (labelled 'YES-').
3. The number of categories to which the abstract was classified manually, and classified as POS by the algorithm (labelled '+POS').
4. The number of categories to which the abstract was classified as POS by the algorithm, but not assigned manually (labelled 'POS-').
5. The number of categories to which the abstract was classified neither manually nor by the algorithm (labelled '+NO').
6. The number of categories to which the abstract was classified manually, but rejected (classified NO) by the algorithm (labelled 'NO-').

The ideal situation would be if columns YES-, POS-, and NO- were all equal to zero for every abstract, but this is unrealistic.

On the basis of the above six numbers, we can determine the 'goodness' of the algorithm, by applying the following scale of 'goodness':

1. Very good - if columns YES- and NO- contain zeros.

2. Good - if column NO- contains zeros.
3. Partially good - if one of columns NO- and +YES are zero.
4. Bad - if both columns NO- and +YES are zero.
5. Very bad - all other cases.

Tests done for different algorithms on the same set of abstracts can be compared by counting the number of abstracts in each of the five grades of 'goodness'.

The testing program reads through a file of abstracts. For each abstract:

- the adhesion coefficients a_k^1 and a_k^2 are evaluated for each category k ,
- the membership function $S'(x_k, \alpha_k, \beta_k, \gamma_k)$ is evaluated for each category k ,
- the set of categories is partitioned into the three fuzzy sets YES, POS, and NO,
- the results are compared to the manual assignment, and entries are made in the six classes +YES, YES-, ..., NO-.

An example of such a test is shown in Table 7 for a test of 22 abstracts using adhesion coefficient a_k^1 and $\beta_k = \frac{1}{2}\gamma_k$. So far, eighteen tests have been run. Table 8 shows the test parameters, and Table 9 summarizes results by showing the number of abstracts that fell into each of the 'goodness' classes.

Some results and comparisons of tests are shown in Figures 25 and 26. It can be seen there how much noise is in the information, and how results depend on numbers of words and their frequency in abstracts and the dictionary. This implies directions for the next phase of our research. First of all, how can the noise be reduced?

Table 7. Example of Testing
a Prediction Algorithm

+ YES -	+ POS -	+ NO -
0 - 2	0 - 0	36 - 2
1 - 0	0 - 0	39 - 0
0 - 1	0 - 0	37 - 2
0 - 1	2 - 2	35 - 0
1 - 0	0 - 1	37 - 1
1 - 0	0 - 0	39 - 0
1 - 1	1 - 0	37 - 0
0 - 1	1 - 0	38 - 0
0 - 1	1 - 0	38 - 0
1 - 0	1 - 2	36 - 0
1 - 0	0 - 0	39 - 0
1 - 0	0 - 0	38 - 1
2 - 0	0 - 0	37 - 1
1 - 0	0 - 0	39 - 0
1 - 0	0 - 3	35 - 1
1 - 0	0 - 1	38 - 0
1 - 0	0 - 0	38 - 1
0 - 1	0 - 0	38 - 1
1 - 1	0 - 1	37 - 0
1 - 1	0 - 3	35 - 0
0 - 1	0 - 0	38 - 1
1 - 1	0 - 2	36 - 0

Table 8. Testing Parameters

Test Nbr.	Occ Cutoff	Type of Prediction	α	γ	β	form of f
2	0	non-log	0	acx/10	$2\gamma/3$	ac
3	0	non-log	0	acx	$2\gamma/30$	ac
4	0	log	0	acx	$2\gamma/3$	ac log ² ctw
5	2	non-log	0	acx	$2\gamma/30$	ac
6	3	non-log	0	acx	$2\gamma/30$	ac
7	2	log	0	acx	$2\gamma/3$	ac log ² ctw
8	1	non-log	0	acx	$2\gamma/30$	ac
9	1	log	0	acx	$2\gamma/3$	ac log ² ctw
10	3	non-log	0	acx	0	ac
11	3	non-log	0	acx	$\frac{1}{2}(\alpha+\beta)$	ac
12	3	non-log	0	acx $\frac{csw}{ctw}$	0	ac $\frac{asw}{atw}$
13	3	non-log	0	acx	$\frac{1}{2}(\alpha+\gamma)$	ac $\frac{\log ctw}{\log atw}$
14	3	non-log	0	acx	0	ac $\frac{\log ctw}{\log atw}$
15	3	non-log	0	acx $\frac{csw}{ctw}$	$\frac{1}{2}(\alpha+\gamma)$	ac $\frac{asw}{atw}$
16	3	non-log	0	acx	0	ac $\frac{csw}{asw}$
17	2	log	0	acx	$2\gamma/3$	ac
18	3	log	0	acx	$2\gamma/3$	ac log ² ctw
19	3	log	0	acx	$2\gamma/3$	ac

Notes on Table 8:

1. Occurrence Cutoff. Words that occurred no more than 'n' times, where 'n' is the number given in this column, were omitted from the dictionary.
2. Type of Prediction. 'non-log' uses adhesion coefficient a_k^1 (equation 1). 'log' uses adhesion coefficient a_k^2 (equation 2).
3. Abbreviations used for γ and f.

acx = adhesion coefficient, taken from Table 6

ax = adhesion coefficient evaluated for an abstract

csw = category separate words (number of distinct words in the category).

ctw = category total words (total number of word occurrences in the category).
 asw = abstract separate words.
 atw = abstract total words.

Table 9. Results of Testing

Test Nbr.	Number of Abstracts in Each Class					Total
	Very Good	Good	Partially Good	Bad	Very Bad	
2	10	1	8	2	10	31
3	9	1	8	2	11	31
4	11	4	6	2	8	31
5	17	5	15	6	23	66
6	19	6	16	6	19	66
7	2	48	7	4	5	66
8	18	9	11	3	25	66
9	3	48	7	2	6	66
10	20	12	9	9	16	66
11	20	7	15	6	18	66
12	20	22	6	8	10	66
13	16	4	18	4	24	66
14	21	13	8	8	16	66
15	17	4	16	6	23	66
16	2	47	7	4	6	66
17	17	5	12	7	25	66
18	2	53	6	0	5	66
19	18	4	11	6	27	66

5.6. The Near Future

To reduce information noise, it is necessary to know something more about the dictionary we use. First of all,

- How big is the intersection of dictionary words for each pair of categories?
- Can we make this intersection empty by deleting some words, and what can we predict on the basis of such a dictionary?
- How many word occurrences can be eliminated? Can it be equal for each category, or better proportional to the number of abstracts for the categories used to make the dictionary?

Many other questions arise after each new test is run.

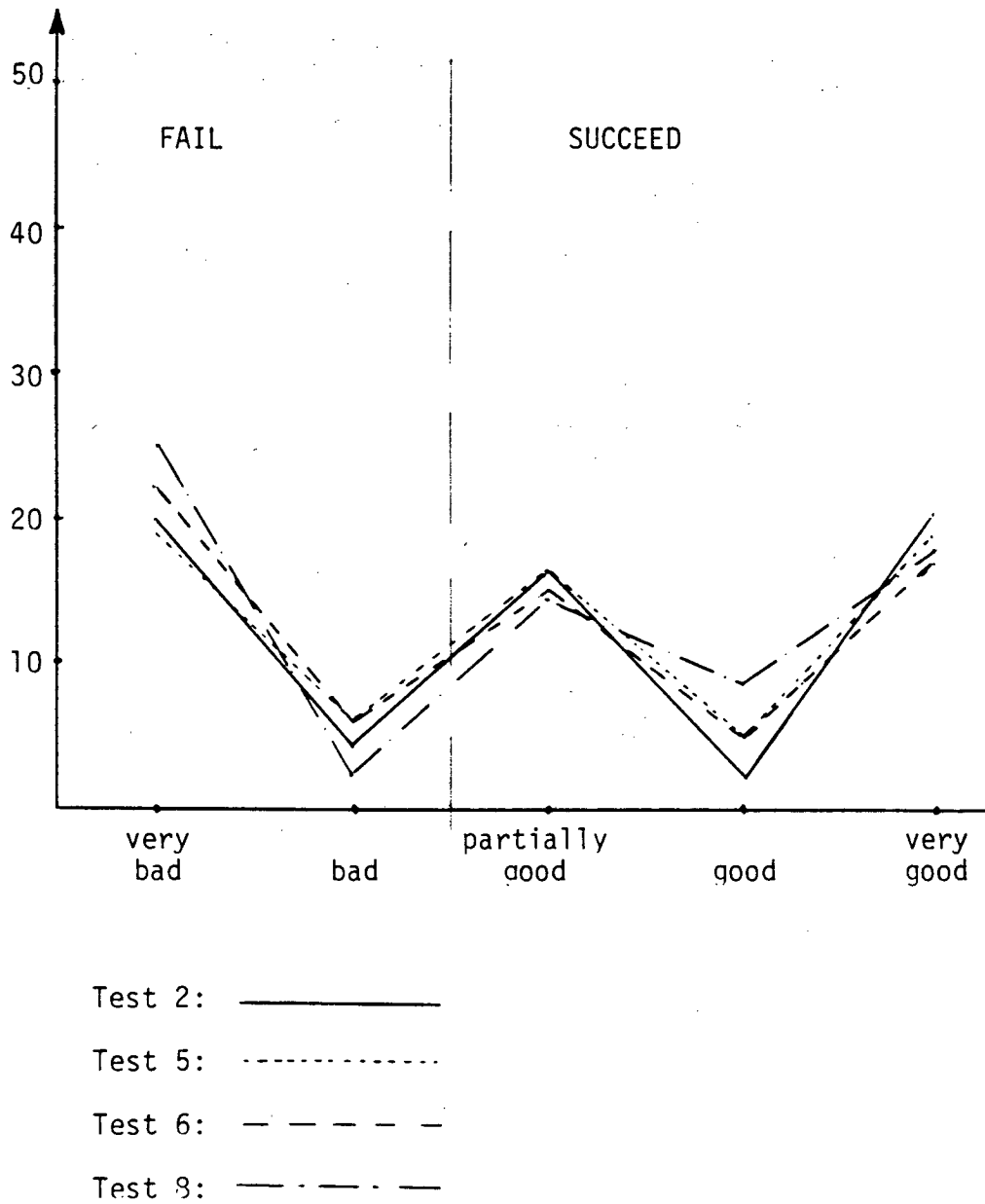


Figure 25. Plot of Test Results

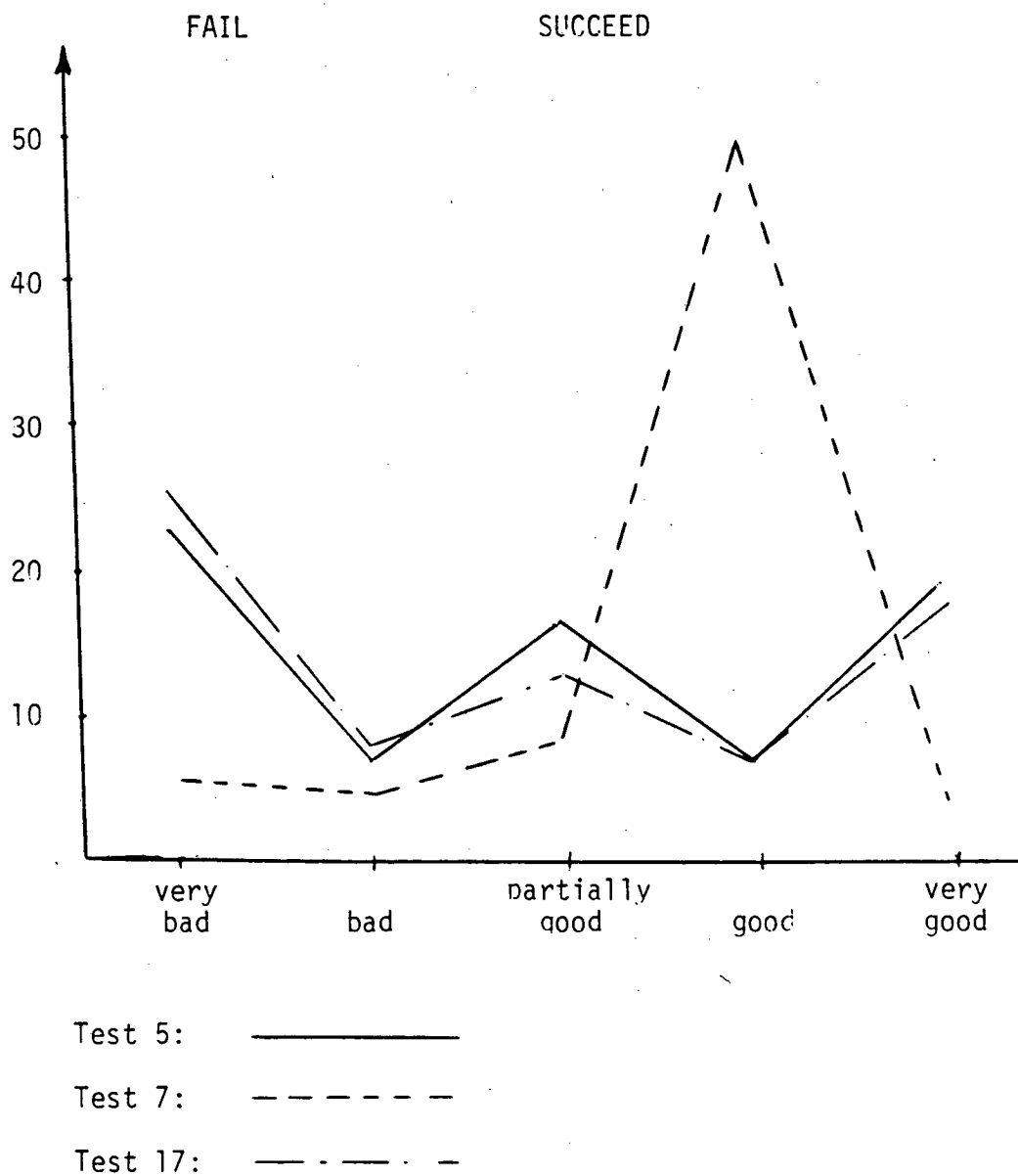


Figure 26. Plot of Test Results

6. FUTURE PLANS

Project goals for the next year or so are fairly well defined. They involve system enhancements leading toward a production capability, program revisions in response to user testing, hardware investigations, and indexing research.

A number of enhancements are required to existing support programs to make them easier to use and more robust, and several new programs will be written to provide additional services. **TICEDT** will be tested by the indexers, with the intent of finding out what works easily and what doesn't. The program will then be overhauled to make it easier to use.

One idea we are considering is the following. Currently, the indexer is aware of two distinct program modes: control, and editing. As new pieces are added (such as **THSGRF**, and access to the inverted index to the thesaurus), the situation becomes even more confusing and difficult to deal with. Another problem that has appeared is the difficulty in creating index terms from the abstract, due to human memory capabilities. The indexer enters the editor to examine the abstract for indexing terms. He then must leave the editor for the Record Editing Menu, and enter the editor again to process the keywords. After a little processing, it becomes necessary to consult the abstract again, so this process is repeated twice (to the abstract, and then back to the keywords).

Discussions between the users at TIC and the authors at LBL have led to an idea that we feel can solve both of these problems. We visualize a set of *frames* available to the indexer, that correspond to different aspects of the indexing; if possible, we wish to be able to have two frames visible at the same time on the screen (using split screening). So far we have identified five such frames; more may be added.

1. Edit. The entire record will be displayed except the key word fields (801-899), in an easy-to-use format. The indexer will be able to add, delete, or modify the contents of any field; the program will keep track of which field is being edited, so there will be no need to alternate between a menu and the editor.
2. Descriptors. A structure of M's and Q's, and splits, will be constructed much as described earlier.
3. Primary thesaurus display. This will be essentially the display of main terms, broader terms, narrower terms, and related terms described above for **THSGRF**.
4. Secondary thesaurus display. This will be essentially the display of definitions, scope notes, use and use for, and see and seen from attributes.
5. Inverted index. This will be a new display, of main terms that include a designated word.

By actually maintaining the contents of each frame intact between calls, this procedure should be quite efficient and effective. It also permits calls to have 'arguments'. By placing the cursor (or a light pen, etc.) on a keyword in Frame 2, and requesting frame 3, the thesaurus around that keyword can be displayed easily. Similarly, by pointing the cursor at a word anywhere, Frame 5 can be entered at the requested word. Thus, there would be two modes for frame transition: back to the immediately preceding display with no change, or to a new display.

Hardware investigations will probably go in two directions. First, split screening is inconvenient on the standard-size terminal, with 24 lines of 80 characters. A terminal with 48 or more lines of 80 characters, or 24 lines of 132 characters would be better. Second, we wish to investigate the use of microprocessors, with the possibility of moving **TICEDT** to the terminal, rather than using the main DEC 10 computer. The program is quite large, and growing, and uses a considerable amount of resources. It might well be more efficient to off-load to a microprocessor.

The indexing research reported on in Chapter 5 will be continued, with the intent of finding a workable algorithm for **SORT** to use. As part of this work, we will investigate the possibility of word stemming, not only in **SORT**, but also in **TICEDT** and **THSGRF**. The movement from frame to frame described above certainly will require some form of stemming, even if only to deal with plurals.

Long range plans go into stages 2 and 3 described in Chapter 3. We believe that many of Klingbeil's ideas [22] have potential application to this project, so this will also be explored.

REFERENCES

1. DOE Technical Information Center: Its Functions and Services, TIC-4600-R2, DOE Technical Information Center, Oak Ridge, TN (January 1978).
2. DOE/RECON: User's Manual, TID-4587, DOE Technical Information Center, Oak Ridge, TN (1981).
3. Julia Redford, Technical Information Center, personal communication (1982).
4. B. A. Cerny and J. D. Lawrence, A Preliminary Report on an On-Line Indexing-Editing System for DOE/Technical Information Center, LBL Report LBID-446, Lawrence Berkeley Laboratory, Berkeley, CA (November 1981).
5. L. C. Smith, Artificial intelligence in information retrieval systems, *Information Processing and Management* 12 3 (1976), 189-222.
6. J. D. O'Connell, E. C. Fubini, K. G. McKay, J. Hillier and J. H. Hollonon, Electronically expanding the citizen's world, *IEEE Spectrum* 6 (1969), 30-40.
7. G. Salton, *The SMART Retrieval System*, Prentice Hall, Englewood Cliffs, NJ (1971).
8. C. J. van Rijsbergen, *Information Retrieval*, Butterworths, London (1979).
9. K. Sparck-Jones and R. G. Bates, Research on Automatic Indexing 1974-1976, Report, Computer Laboratory, University of Cambridge (1976).
10. H. F. Stiles, The association factor in information retrieval, *Journal of the ACM* 8 (1961), 271-291.
11. DOE Energy Information Data Base: Energy Categories, TID-4584, DOE Technical Information Center, Oak Ridge, TN (1980).
12. DOE Energy Information Data Base: Subject Thesaurus, TID-7000, DOE Technical Information Center, Oak Ridge, TN (1979).
13. W. S. Cooper, Is inter-indexer consistency a hobgoblin? *American Documentation* 2 (1969), 268-278.
14. J. A. Digger, *A Study of the Intellectual Elements of Indexing for Information Retrieval*, thesis submitted for fellowship of the Library Association (1973).
15. Guide to Abstracting and Indexing at the Technical Information Center, TID-4583, National Technical Information System (1978).
16. S. Amarel, Problem solving and discussion-making by computer: an overview, In: *Cognition: A Multiple View*, P. L. Garvia (ed), Spartan Books, NY (1970), 279-329.
17. L. Evans, Evaluation of the ISpra Automatic Indexing Programs SLC-II. Final Report, INSPEC (July 1978).
18. R. W. Graves and D. P. Helander, A feasibility study of automatic indexing and information retrieval, *IEEE Transactions on Engineering Writing and Speech*, EWS-13 (2), (1970), 58-59.
19. J. B. Lovins, Development of a Stemming Algorithm, *Mech. Trans. Comput. Linguis.* 11, (1968), 22-31.
20. D. B. McCarn and J. Leiter, On-line services in medicine and beyond, *Science* 181 (1973), 318-324.
21. H. P. Luhn, A statistical approach to mechanized encoding and searching of literary information, *IBM Journal of Research and Development* 1 (1957), 309-317.
22. P. H. Klingbeil, Machine-aided indexing of technical literature, *Information Storage and Retrieval* 9 (1973), 79-84.
23. D. E. Walker, The organization and use of information: contributions of information sciences, computational linguistics and artificial intelligence, *JASIS* 32 (5), (1981), 374-363.
24. N. V. Findler (ed), *Associative Networks — The Representation and Use of Knowledge in Computers*, Academic Press, NY (1979).

25. J. Figen, A Summary of the RATFOR Language, an Extended Portable Dialect Called REP, Its Style and Flavor, and Details of Its Implementation on the PDP-10, Report LBL-13405, Lawrence Berkeley Laboratory, Berkeley, CA (September 1981).
26. DOE Energy Information Data Base: DECOL Manual, TID-4602, DOE Technical Information Center, Oak Ridge, TN (1980).
27. INIS Subject Categories and Scope Descriptions, Report No. IAEA-INIS-3 (rev. 5), International Atomic Energy Agency, Vienna (Nov. 1978).
28. D. Soergel, *Indexing Languages and Thesauri: Construction and Maintenance*, Wiley, Los Angeles, 1974.
29. B. J. Field, Towards Automatic Indexing: Automatic Assignment of Controlled-Language Indexing and Classification from Free Indexing, *Journal of Documentation* 31 (4) (1975), 246-265.
30. V. Kh. Kaipov and A. A. Selingin, Classification in Fuzzy Environment, in M. M. Gupta and E. Sanchez (ed), *Approximate Reasoning in Decision Analysis* (1982).
31. Mizumoto and K. Tanaka, Algebraic properties of fuzzy numbers, *Proceedings of the International Conference on Cybernetics and Society*, Washington, D. C. (1976), 559-563.
32. L. A. Zadeh, Fuzzy sets as a basis for a theory of possibility, *Fuzzy Sets and Systems 1* (1978), 3-28.
33. L. A. Zadeh, Fuzzy sets, *Information and Control* 8, (1965), 338-353.

This report was done with support from the Department of Energy. Any conclusions or opinions expressed in this report represent solely those of the author(s) and not necessarily those of The Regents of the University of California, the Lawrence Berkeley Laboratory or the Department of Energy.

Reference to a company or product name does not imply approval or recommendation of the product by the University of California or the U.S. Department of Energy to the exclusion of others that may be suitable.

TECHNICAL INFORMATION DEPARTMENT
LAWRENCE BERKELEY LABORATORY
UNIVERSITY OF CALIFORNIA
BERKELEY, CALIFORNIA 94720