

**COMPUTERIZED CONFERENCING
& COMMUNICATIONS CENTER**

at

**NEW JERSEY
INSTITUTE OF TECHNOLOGY**

RESOURCES REFERENCE GUIDE 1

**A PRELIMINARY REFERENCE GUIDE FOR
THE RESOURCES NETWORK DATABASE SYSTEM**

RESEARCH REPORT NUMBER NINETEEN

Draft: September 1982

**c/o Computer & Information Science Department
New Jersey Institute of Technology
323 High Street, Newark, N. J. 07102**

Computerized Conferencing and Communications Center

at

New Jersey Institute of Technology

RESOURCES REFERENCE GUIDE 1

A PRELIMINARY REFERENCE GUIDE FOR
THE RESOURCES NETWORK DATABASE SYSTEM

RESEARCH REPORT NUMBER NINETEEN

Draft: September 1982

RE-SOURCE, n. [Fr. RESSOURCE, from Old Fr. RESOURDRE, to arise anew; RE-, again, and SOURDRE, to spring up as water, from L, SURGERE, to arise, contr. for SURRIGERE; SUB, under, and REGERE, to direct.]

CREDITS

The RESOURCES system was originally developed by Peter and Trudy Johnson-Lenz and James Whitescarver for the Neighborhood Information Sharing Exchange project, managed by Community Methods, Inc., Washington, D.C., Miles Martin, President.

Jim Whitescarver and John Foster, New Jersey Institute of Technology, have been responsible for the implementation of RESOURCES. They have been assisted in the system validation and documentation by William Hatfield and Peter and Trudy Johnson-Lenz.

The RESOURCES network database system has been designed to operate within the Electronic Information Exchange System (EIES), located at the New Jersey Institute of Technology, Murray Turoff, designer.

Support for Jim Whitescarver during the design phase and for augmentation of the INTERACT programming language to support RESOURCES, and support for John Foster was provided by the Computerized Conferencing and Communications Center of the New Jersey Institute of Technology, Murray Turoff, Director, under grant MCS 78-00519 from the National Science Foundation. Additional funding has been provided by the County of Essex, New Jersey and the Chloromone Corporation.

NOTICE TO USERS

IT IS EMPHASIZED THAT THIS PRELIMINARY DOCUMENTATION FOR THE RESOURCES NETWORK DATABASE SYSTEM CONFORMS TO THE SYSTEM SPECIFICATIONS.

RESOURCES REFERENCE GUIDE 1 is for limited distribution and review. It is not a user manual. Your interaction with the system may not always be exactly as described. All major features except keyword relations are supported in part, if not fully, at this time. Operation of some facilities are some what slow pending funding for optimization.

Groups and individuals wishing to use RESOURCES on the Electronic Information Exchange System should make arrangements through the Computerized Conferencing and Communications Center at the New Jersey Institute of Technology to discuss their application and the establishing of user accounts, disk storage charges and monthly account charges, software maintenance, and consultation in the effective use of the system.

(A current list of operational commands within RESOURCES is available by entering a single question mark (?) at COMMAND? while in any RESOURCES database.)

**** March 14, 1983 ****

Table of Contents

1 OVERVIEW OF THE RESOURCES SYSTEM.....	0
2 BASIC SYSTEM CONCEPTS.....	2
2.1 DATABASES AND RESOURCES.....	4
2.2 ACCESS: USERS, LEVELS OF ACCESS.....	5
2.3 THESAURUS: KEYWORDS AND THEIR INTERRELATIONSHIPS.....	6
2.4 RESOURCE PARTS.....	7
2.4.1 SYSTEM PARTS.....	8
2.4.2 DATE PARTS.....	8
2.4.3 ID PARTS.....	8
2.4.4 ASSIGNING THE PRIMARY ID PART.....	9
2.4.5 TEXT PARTS.....	9
2.4.6 KEYWORD PARTS.....	9
2.4.7 NUMBER PARTS.....	10
2.4.8 TYPE PARTS.....	11
2.4.9 LABEL PARTS.....	11
2.4.10 OPERATIONS ON PARTS.....	11
2.4.11 THE DATABASE FORMAT.....	12
2.4.12 INDEXED PARTS.....	13
2.5 GETTING AND FINDING RESOURCES.....	13
2.5.1 LISTS AND RANGES.....	14
2.5.2 PART SPECIFICATIONS.....	14
2.5.3 PART-SPECIFIC PROMPTS.....	16
2.5.4 PATTERNS.....	16
2.5.5 RECURRENT FINDS WITH PATTERNS.....	18
2.5.6 TEMPLATES.....	18
2.5.7 DELIVERY OF MOST RECENT RESOURCES.....	19
2.5.8 LAYOUTS.....	19
2.6 TALLIES, COMMENTS AND MARKERS.....	20
2.7 ACTIVITY TRACE: USER ACTIVITY AND BILLING INFORMATION.....	21
2.8 MONITOR OPTIONS.....	21
3 THE RESOURCES USER COMMAND INTERFACE.....	23
3.1 GENERAL RESOURCES COMMAND FEATURES.....	24
3.1.1 COMMAND MENU AND DESCRIPTIONS.....	24
3.1.2 RETURNING TO EIES.....	25
3.1.3 ANSWERING AHEAD CONVENTIONS.....	25
3.1.4 ACTIVITY TRACE BY COMMAND.....	25
3.2 DATA ENTRY/MODIFICATION COMMANDS.....	26
3.2.1 ADDING A RESOURCE TO THE DATABASE (AR).....	26
3.2.2 MODIFYING AN EXISTING RESOURCE (MR).....	27
3.2.3 REMOVING A RESOURCE FROM THE DATABASE (RR).....	28
3.2.4 FILE INVERSION: UPDATE DATABASE (UD).....	28
3.2.5 UPDATING RESOURCES (UR).....	29
3.3 THE RESOURCE EDITOR.....	30
3.3.1 RESOURCE EDITOR COMMANDS.....	31
3.3.1.1 ABBREVIATIONS: THE FIRST-FOUND RULE.....	33
3.3.1.2 SPECIAL USER SUPPORT FOR ENTRY OF KEYWORDS.....	33

3.3.1.3	PART-SPECIFIC PROMPTS.....	33
3.3.1.4	EXAMPLE INTERACTION WITH THE RESOURCE EDITOR.....	34
3.3.1.5	CHECKS PERFORMED BY THE RESOURCE EDITOR.....	35
3.3.1.8	SPECIAL ANSWER-AHEAD FEATURE FOR NUMBERS.....	36
3.3.1.9	PROMPTED MODE RESOURCE ENTRY.....	36
3.3.1.10	MODIFYING RESOURCES DURING PROMPTED MODE.....	38
3.3.1.11	DEFAULT ENTRIES IN PROMPTED MODE.....	38
3.3.2	MICRO MODE FOR THE RESOURCE EDITOR (ER).....	38
3.3.2.1	MICRO MODE HANDSHAKING.....	39
3.3.2.2	PARITY CHECKING.....	39
3.3.2.3	CHECKING AND FORMATTING IN MICRO MODE.....	39
3.3.2.4	END OF RESOURCE SIGNAL.....	40
3.3.2.5	END OF TRANSMISSION SIGNAL.....	41
3.3.2.6	EXAMPLE INTERACTION WITH MICRO MODE.....	41
3.3.2.7	MICRO MODE CHECKING FOR UNIQUENESS OF ID PARTS...	42
3.3.2.8	TRACE OF MICRO MODE ENTRIES.....	43
3.3.3	STANDARD RESOURCE FORMATS FOR INPUT AND OUTPUT.....	43
3.3.3.1	STANDARD RESOURCE OUTPUT FORMAT.....	44
3.3.3.2	STANDARD SYSTEM ID LINE FORMAT.....	44
3.3.3.3	STANDARD RESOURCE INPUT FORMAT.....	45
3.3.3.4	STANDARD RESOURCE PROMPTED FORMAT.....	45
3.3.3.5	OUTPUT IN STANDARD INPUT FORMAT.....	45
3.3.3.6	STANDARD GET/FIND FEEDBACK MESSAGES AND OUTPUT...	46
3.3.3.7	SUMMARY OF USE OF STANDARD PROMPTS.....	47
3.4	THESAURUS MANAGEMENT COMMANDS.....	47
3.4.1	ADDING A KEYWORD TO THE THESAURUS (AK).....	47
3.4.2	REMOVING KEYWORDS FROM THE THESAURUS (RK).....	48
3.4.3	MODIFYING THE RELATIONSHIPS AMONG KEYWORDS (MK).....	49
3.4.4	DISPLAYING KEYWORDS IN THE THESAURUS (DK).....	50
3.4.5	LISTING THE THESAURUS FOR MICRO MODE (XK).....	51
3.4.6	GETTING THE THESAURUS (GK).....	52
3.5	GET/FIND RESOURCE COMMANDS.....	53
3.5.1	PRINTING RESOURCES WAITING (PR).....	54
3.5.2	GETTING RESOURCES BY A SINGLE PART (GR).....	54
3.5.3	DISPLAYING RESOURCES (DR).....	55
3.5.4	UNKNOWN CONTENTS.....	56
3.5.5	PART-SPECIFIC PROMPTS.....	56
3.5.6	LISTS AND RANGES.....	56
3.5.7	USER SUPPORT FOR GETTING RESOURCES.....	56
3.5.8	TRACE OF GETS.....	58
3.5.9	FINDING RESOURCES (FR).....	58
3.5.9.1	BOOLEAN COMBINATIONS OF PARTS.....	59
3.5.9.2	USER SUPPORT FOR MULTIPLE KEYWORD FINDS.....	59
3.5.9.3	FINDING RESOURCES WITH NON-KEYWORD PARTS.....	61
3.5.9.5	USER SUPPORT FOR FINDING RESOURCES.....	62
3.5.9.6	ACCESS TO COMMENTS AND TALLY FEEDBACK.....	62
3.5.9.7	TRACE OF FINDS.....	63
3.5.9.8	COMPARISON OF GR AND FR FEATURES.....	63
3.5.10	PATTERN COMMANDS.....	63
3.5.10.1	ADDING PATTERNS (AP).....	64
3.5.10.2	MODIFYING PATTERNS (MP).....	65
3.5.10.3	REMOVING PATTERNS (RP).....	66
3.5.10.4	DISPLAYING PATTERNS (DP).....	66
3.5.10.5	FINDING BY PATTERNS (FP).....	66
3.5.10.6	ACCESS TO COMMENT AND TALLY FEEDBACK.....	67
3.5.10.7	TRACE OF PATTERN FINDS.....	67
3.5.10.8	PRINTING OUT FOUND RESOURCES IN INPUT FORMAT (XP)	68
3.5.10.9	PRINTING OUT PARTS OF RESOURCES.....	68

3.5.11	SPECIAL RESOURCE OUTPUT FEATURES.....	68
3.5.11.1	MODIFYING THE TEMPLATE (MTE).....	69
3.5.11.2	GETTING THE TEMPLATE (GTE).....	69
3.5.11.3	THE TEMPLATE IN EFFECT NOTICE.....	70
3.5.11.4	PRINTING RESOURCES IN STANDARD INPUT FORMAT (XR).....	70
3.5.11.5	ADDING LAYOUTS (AL).....	71
3.5.11.6	MODIFYING LAYOUTS (ML).....	72
3.5.11.7	REMOVING LAYOUTS (RL).....	72
3.5.11.8	GETTING LAYOUTS ON FILE (GL).....	72
3.5.11.9	UPDATING LAYOUTS FOR RESOURCE PRINTING (UL).....	73
3.6	REPORT-WRITING COMMANDS.....	73
3.6.1	GETTING SORTED LISTS OF RESOURCES (SR).....	74
3.6.2	TABULATING RESOURCES BY PARTS (TR).....	75
3.6.3	TABULATING FREQUENCY OF ACCESSES TO RESOURCES (TD).....	78
3.6.4	TABULATING USER STATISTICS (TU).....	79
3.6.5	CROSS TABULATING RESOURCES BY PARTS (CTR).....	80
3.7	FEEDBACK COMMANDS.....	80
3.7.1	ENTERING COMMENTS IN THE FEEDBACK FILE (AC).....	80
3.7.2	ENTERING PEN OR ANONYMOUS COMMENTS.....	81
3.7.3	TRACE OF COMMENTS ADDED.....	82
3.7.4	MODIFYING COMMENTS (MC).....	82
3.7.5	REMOVING COMMENTS (RC).....	83
3.7.6	GETTING COMMENTS FROM THE FEEDBACK FILE (GC).....	83
3.7.7	DISPLAYING COMMENT HEADERS ONLY (DC).....	84
3.7.8	PRINTING COMMENTS WAITING (PC).....	84
3.7.9	MODIFYING COMMENT MARKERS (MM).....	85
3.7.10	COPYING INTO THE SCRATCHPAD (+RCY) AND (+RCIN).....	85
3.7.11	ADDING TALLIES TO THE FEEDBACK FILE (AT).....	85
3.7.12	DISPLAYING TALLY RATINGS (DT).....	87
3.7.13	GETTING TALLY QUESTIONS AND RATINGS (GT).....	87
3.8	FORMAT MANAGEMENT COMMANDS.....	88
3.8.1	ADDING A FORMAT TO A DATABASE (AF).....	88
3.8.2	MODIFICATION OF FORMAT (MF).....	91
3.8.3	MODIFYING QUESTIONS IN A TALLY (MT).....	92
3.8.4	GETTING A PRINTOUT OF THE FORMAT (GF).....	93
3.8.5	TRANSMITTING THE FORMAT TO A MICROCOMPUTER (XF).....	94
3.9	USER MANAGEMENT COMMANDS.....	95
3.9.1	ADDING A USER (AU).....	96
3.9.2	MODIFYING USERS IN THE DATABASE (MU).....	96
3.9.3	REMOVING USERS FROM THE DATABASE (RU).....	97
3.9.4	DISPLAYING DATABASE USER LIST (DU).....	97
3.9.5	GETTING USER INFORMATION (GU).....	97
3.9.6	GETTING A USER'S ACTIVITY TRACE (GA).....	98
3.9.7	PRUNING THE USER ACTIVITY TRACE FILES (MA).....	99
3.10	RESOURCES SYSTEM MANAGEMENT COMMANDS.....	100
3.10.1	ADDING A DATABASE (AD).....	100
3.10.2	MODIFYING DATABASE MONITOR OPTIONS (MD).....	101
3.10.3	REMOVING DATABASES (RD).....	102
3.10.4	GETTING DATABASE OPTIONS (GD).....	102
3.10.5	DISPLAYING DATABASE RELATIONS (DD).....	103
3.11	EIES SYSTEM COMMANDS (+RESOURCES, +RCY, +RCIN).....	103
3.11.1	RESOURCE EDITOR MODE COMMANDS (+DIRECT, +PROMPTED).....	104
3.11.2	SWITCH 14 FOR TOP-OF-PAGE CONTROL (+SSW14).....	105
3.11.3	SWITCH 6 FOR RECEIVING YOUR OWN ITEMS (+SSW6).....	105
3.11.4	SWITCH 20 FOR MICROCOMPUTER HANDSHAKE (+SSW20).....	105
3.12	BACKGROUND COMMAND PROCESSING (BG).....	105
4	COMMAND SUMMARY.....	108

5 CAPACITIES AND LIMITATIONS.....111

1.0 OVERVIEW OF THE RESOURCES SYSTEM

The RESOURCES network database system (RESOURCES) is a specially tailored database and communications system, written in INTERACT, to run on the Electronic Information Exchange System (EIES). With RESOURCES, databases may be created and data or information items (resources) may be entered and retrieved. However, RESOURCES goes far beyond this basic "database" function by nesting and integrating data entry, data retrieval, and the databases themselves within the larger communications context of EIES.

First, RESOURCES databases can be configured to allow retrieval of resources by any database user regardless of geographic location. Second, users can be given appropriate levels of access to allow them to enter and modify resources in the database directly, thereby decentralizing and diffusing the typical "central clearinghouse" bottleneck plaguing modern database systems. Third, RESOURCES expedites timely retrieval by providing two kinds of personal markers under control of each user that allow sequential delivery of new resources (like a conferencing system) or that manage pre-defined searches on recently-added resources automatically (like a SDI) while the user is not online. Fourth, any database user retrieving a resource is allowed by the system to enter an evaluative comment and to rate the resource item on a numeric scale specified by the database monitor. Subsequently, database users and monitors alike can access this evaluative feedback. Finally, RESOURCES is equipped with features to facilitate offline entry and editing of resources in a microcomputer for subsequent transmission to the intended database or databases on EIES. This further decentralizes the network of sources and resources, and reduces connect costs.

RESOURCES databases contain resources, and each resource contains several discrete parts (these are sometimes called "records" and "fields" in other systems). Resource items can be entered with a variety of associated kinds of parts including free text, keyword descriptors, numbers, dates and labels such as name, address, telephone, etc. Resources can be grouped into databases of similar items; Organizations, Resource People, Publications, are examples of such groupings.

More than one database can be created. Each database can have its own unique format describing the uniform list of parts for every resource in that database. Thus, separate databases can be created for different types of resources such as organizations, resource people, publications, etc. Where appropriate, databases in RESOURCES can share a common THESAURUS of keywords as well as a file of users with shared access to several databases.

RESOURCES also includes a variety of special evaluation and report-writing features that aid database monitors in tracking, managing and maintaining efficient utilization of even the most active databases.

2.0 BASIC SYSTEM CONCEPTS

The basic unit of data in the RESOURCES network database system is a RESOURCE. These resources are organized into one or more DATABASES. Each DATABASE contains similar kinds of resources, such as one database on people, another on publications, and still another on organizations. This allows different kinds and FORMATS of information, called RESOURCE PARTS, to be associated with each different kind of resource.

A DATABASE MONITOR must specify the FORMAT that defines the PARTS, their names and their characteristics. Once it is defined for a database, the FORMAT remains constant over all resources entered into that database and is used by the data entry and modification routines to assure integrity of the database.

Access to a database can be given to any EIES account or subaccount user by making them a USER OF THE DATABASE. Users are assigned a LEVEL OF ACCESS when they are ADDED to the database. This level of access specifies whether the user may only READ, may ADD and MODIFY their own resources, may EDIT any resources or may have a higher access level (such as being able to grant access to the database). This allows a diverse network of users--who may be geographically dispersed--sufficient access to get their work done while still allowing database monitors control over the database.

Associated with each resource in a database is a FILE OF TALLIES and another FILE OF COMMENTS. Each of these files contains FEEDBACK from users who have retrieved a resource and entered COMMENTS or rated the resource on a TALLY SCALE. Anyone with READ LEVEL ACCESS can use these feedback files at any time to see what others' reactions to the resources are.

In addition to being able to search the database of resources by Boolean combinations of keywords and numbers, a user may enter one or more of these combinations into the system as a SEARCH PATTERN. This PATTERN serves as an ongoing search specification used automatically by the system to periodically scan new entries and FIND those resources that match the PATTERN.

The system also keeps detailed traces of user activity showing how many times and for how long the system was used, which resources were retrieved and whether comment feedback was entered. These traces of activity can be used in a variety of ways (such as billing and project evaluation).

Each of these basic concepts is included as a part of the overall database in some fashion and is an "operand" in the algebra of operations that can be performed on the database. Sections (2.1) through (2.8) detail these operands and summarize the operations

allowed by the system.

The RESOURCES user command interface, and the command syntax for the system, are then specified in Section (3). Each command corresponds to an operand/operator pair in this algebra by being made up of a verb (operator) and noun (operand). These commands are detailed, their functional characteristics are described and examples are given.

2.1 DATABASES AND RESOURCES

The DATABASE is the largest unit of organization within the RESOURCE system. A database is then divided into resources, as discussed in (1) and (2). Also included in the database is a FORMAT (3.8, 3.10) and a file of the USERS who are members of that database. Where appropriate, the file of USERS may be shared between two or more DATABASES.

Each resource within a database is uniquely identified by a SYSTEM ID number that is sequentially assigned when the resource is entered. These system ID numbers are automatically generated and beyond the control of the user. However, the FORMAT specifying the PARTS of each RESOURCE may include one or more additional ID PARTS under control of the user. This allows users to enter and retrieve resources using unique IDs. These user controlled ID PARTS allow an alphabetic portion and a numeric portion and are discussed in (2.4).

The basic OPERATIONS that can be performed on a database are ADD, GET, REMOVE, DISPLAY, MODIFY and UPDATE. The ADD operation creates a new database. The GET operation displays status information about the database. The REMOVE operation deletes an existing database in its entirety. The DISPLAY operation prints out status information about all of the databases in RESOURCES, showing relationships between the databases. The MODIFY operation allows modification of database status. The UPDATE operation keeps the database current by updating the INVERTED INDICES used for rapid retrieval of RESOURCES. These operations are described in (3.10).

The basic operations that can be performed on a resource are ADD, REMOVE, DISPLAY, GET, ENTER, PRINT, TRANSMIT, FIND, MODIFY, and COPY. The ADD operation adds a resource to the database, REMOVE removes a resource, and MODIFY modifies an existing resource. The DISPLAY and GET operations print out either part or all of a resource. The ENTER operation is used to enter resources from a remote microcomputer and the TRANSMIT operation is used to transmit resources from a database to a microcomputer. The PRINT operation allows printing of resources not yet seen by the user, as in a conferencing system. The FIND operation is used to search for resources using complex, interactive search specifications. The COPY operation copies the resource from the database into the EIES scratchpad. These operations are described in (3.2) to (3.5).

2.2 ACCESS: USERS, LEVELS OF ACCESS

Each database has an associated file of USERS who have membership access to that database. Because two or more databases may share a file of USERS, access to one database shared by a group gives access to all databases of that group.

Each user has a LEVEL OF ACCESS, an ACTIVITY FILE (3.1.4, 3.9.6), a TEMPLATE (3.5.11), a COMMENT MARKER (3.7.9), a RESOURCE MARKER (3.5.1), a file of LAYOUTS (3.5.11) and a FILE OF PATTERNS (3.5.10).

The LEVEL OF ACCESS is a number from 0 through 60 and is assigned to a user when the user is ADDED to the database by a gatekeeper or monitor. Below is a table showing the operations and activities that are allowed for the eight levels of access defined for the system:

LEVEL 0: READ-ONLY ACCESS -- user may get any resource in the database and may get any comments and/or the results of tallies associated with any resource; user may NOT add comments or resources or respond to tallies.

LEVEL 10: COMMENT-ONLY ACCESS -- user may get any resource or comment and may add feedback comments and respond to tallies as well; however, user may NOT add resources.

LEVEL 20: ADD RESOURCE ACCESS -- user may add resources to the database and may modify resources entered by him or herself.

LEVEL 30: EDITOR ACCESS -- an editor may add and modify ANY RESOURCE in the database, even those added by others. An editor may edit ANY comment in a feedback file. An editor may add and remove keywords to/from the thesaurus and may modify the relationships among keywords.

LEVEL 40: GATEKEEPER ACCESS -- a gatekeeper may add and remove users to and from the database.

LEVEL 50: MONITOR ACCESS -- a monitor may modify database options and parameters, including specifying the format for resources and questions for the tallies.

LEVEL 55: ADMINISTRATOR ACCESS -- an administrator may add a new database and may specify that it is to

share users and/or keywords with any existing database.

LEVEL 60: IMPLEMENTOR ACCESS -- system programmer access to the database.

(NOTE: Each level of access has the powers of all levels with access numbers less than its own.)

The basic operations which can be performed on a USER are ADD, DISPLAY, GET, MODIFY, REMOVE and TABULATE. The ADD and REMOVE operations add users to or remove users from a database. The DISPLAY and GET operations print out brief or detailed listings of a user's status. The MODIFY operation modifies a user's level of access. In addition there is a TABULATE operation for computing and reporting certain statistics on user activity. These operations are discussed in (3.6) and (3.9).

2.3 THESAURUS: KEYWORDS AND THEIR INTERRELATIONSHIPS

Each database has a THESAURUS or index of KEYWORD descriptors for each KEYWORD PART (2.4.6). Each KEYWORD in a THESAURUS has an associated list of resources in the database which are keyed with that keyword and filed separately, allowing retrieval of resources by that particular part only.

Each KEYWORD has an associated list of RELATED KEYWORDS; each entry in this list points to another KEYWORD in the THESAURUS and includes information denoting the type of relationship. KEYWORDS may be related in any of five ways: BROADER THAN, NARROWER THAN, RELATED TO, USE FOR, and USE INSTEAD.

The relationships among keywords are independent of resources. A resource may be keyed with a certain KEYWORD but NOT also keyed with a BROADER term which also applies. This keeps the total number of keywords per resource down and allows freedom of assignment resource rather than of keywords, so the most appropriate key can be associated with a resource without having to associate all keys that are also logically broader. Relationships among keywords can be modified freely without attention to resources because the relationships are in the THESAURUS, not in the resources.

The USE FOR and USE INSTEAD relationships allow frequently used synonyms to be included in the thesaurus but allows only one (the preferred USE FOR keyword) to be associated with a resource. Attempts to associate a USE INSTEAD keyword with a resource will be corrected by the system prompting to "Use [the preferred keyword] instead."

Each relationship among keywords is two-way. If KEYWORD1 is BROADER than KEYWORD2, the system will automatically assume that KEYWORD2 is NARROWER than KEYWORD1. If KEYWORD1 is specified as USE

FOR KEYWORD2, then the system assumes that KEYWORD2 is automatically specified as USE INSTEAD: KEYWORD1. However, this is as far as the system goes in enforcing the logic of relationships among keys. It DOES NOT CHECK to see if a given keyword is both BROADER and NARROWER than some other keyword. This allows the editor freedom to associate two keywords in apparently contradictory ways, allowing complex, multi-perspective taxonomies. For example, from certain perspectives NEIGHBORHOOD is NARROWER than COMMUNITY while from others it is BROADER than COMMUNITY. The system will allow both relationships to be entered into the THESAURUS. Thus, generally, a THESAURUS is a store of relational information for tailoring and refining a keyword search.

The operations that can be performed on keywords in the THESAURUS are ADD, REMOVE, DISPLAY, GET, MODIFY, TABULATE, and TRANSMIT. ADD and REMOVE adds or removes a keyword. The DISPLAY operation generates a compressed list of all the keywords. The GET operation allows optional printout of a portion of the THESAURUS defined by an alphabetical RANGE of keywords. The GET operation prints out the KEYWORD, the IDs of RESOURCES associated with the KEYWORD and the relationships between that keyword and others in the THESAURUS. The MODIFY operation allows an editor to add and remove keyword relationships. The TABULATE operation tabulates the frequency KEYWORDS describe resources. The TRANSMIT operation transmits all or part of the thesaurus to a microcomputer. These operations are described in (3.4).

2.4 RESOURCE PARTS

Each resource is made up of some number of PARTS. Each PART contains some portion of the information in the resource. There are EIGHT KINDS OF PARTS that can be included in each resource: DATE PARTS, ID PARTS, KEYWORD PARTS, LABEL PARTS, NUMBER PARTS, SYSTEM PARTS, TEXT PARTS and TYPE PARTS.

SYSTEM PARTS are not under control of the user, are already defined and named by RESOURCES and include the SYSTEM ID. The eight named categories using this kind of part are discussed collectively in (2.4.1).

The REMAINING SEVEN kinds of parts are under user control, are defined and named by the monitor, and receive and store information entered by users, allowing for its orderly classification and retrieval. ANY ONE OF THESE PARTS (or SYSTEM ID) may be assigned to be the PRIMARY ID for resources in a database. These seven kinds of parts and primary ID assignment are discussed in (2.4.2) through (2.4.9).

Specifying how many parts will be used, what kind, what they are named and what their limits are is done by the monitor as part of specifying the FORMAT of the database. FORMAT is discussed in,

e.g., (2.4.11), (3.8) and (3.10).

2.4.1 SYSTEM PARTS

All PARTS except the SYSTEM PARTS are under the direct control of the user, and are described in (2.4.2) to (2.4.9). SYSTEM PARTS described in this section are always automatically entered by the computer. While a user is free to use any SYSTEM PARTS in GET and FIND operations, the user cannot enter or modify these parts because they are under the direct control of the system. Named SYSTEM PARTS include: the SYSTEM ID (described in Section 2.1), the SYSTEM DATE (the date when the resource was added to the system), the SYSTEM MDATE (the date when the resource was last modified), the SYSTEM ADATE (the date when the resource was last accessed), the SYSTEM SOURCE (the EIES account number and hence the user that entered the resource), the SYSTEM COMMENTS (the number of comments entered as feedback on the resource), the SYSTEM TALLIES (the number of tally ratings entered as feedback on the resource), and the SYSTEM ACCESSES (the number of times the resource has been accessed). The SYSTEM ID, SYSTEM DATE, and SYSTEM SOURCE are included in the header line of each resource. Header lines or "headers" are printed out at the beginning of a resource. The SYSTEM COMMENT count and SYSTEM TALLIES are printed out by the GR (Get Resources) command (3.5.2). The SYSTEM ACCESSES are printed out by the TR (Tabulate Resources) command (3.6.2).

2.4.2 DATE PARTS

Any number of DATE PARTS can be included in the FORMAT for a database. Each DATE PART allows only the entry of the form MM/DD/YY, where MM is the month, DD is the day and YY the last two digits of the year. Date parts may be no longer than eight characters, the minimum accommodation for the specified form. Date parts may be used in GET and FIND operations. They may also be combined with other parts to build complex PATTERNS for FINDING resources in the database (3.5).

2.4.3 ID PARTS

Any number of ID PARTS can be included in the FORMAT for a DATABASE. Each ID part has two sub-parts: an alphabetic portion and a numeric portion.

These two sub-parts are separated from each other by the special character #. The numeric portion following this # character may contain only the digits 0 through 9. The alphabetic portion before the # character may contain any characters, including special characters such as *, \$, and !, except the # character which always signals the beginning of the numeric portion. It may also

contain numeric digits, but it must begin with an alphabetic or special character. The only other restriction on such IDs is that the entire ID, alpha portion plus number portion, must uniquely define a single resource within any particular database. This assures that the ID is thus a unique identifier which can be used to retrieve one and only one resource.

The two-part structure of these ID PARTS allows a wide variety of identification schemes to be built into a database, including: (1) use of the numeric portion only for simple numeric identification (the ID begins with the special # character), (2) use of the alpha portion only (the # character is excluded), (3) sub-database identification where the alpha portion identifies the sub-database and the numeric portion identifies the resources within the sub-database, (4) multiple source or networked databases where each source or node in the network wishes to use its own numbering system for retrieval of resources it adds while still retaining separate identification of those resources (here the alpha portion can be used as the node identifier and the numeric portion can contain the node-specific numbers, and (5) adding resources in any order without regard to the sequence of ID numbers.

The system will always capitalize the alphabetic portion of any ID PART to avoid confusions between upper and lower case versions of the same ID.

2.4.4 ASSIGNING THE PRIMARY ID PART

The PRIMARY ID is the part selected and assigned by the database monitor for RESOURCES to use as the primary identifier of resources in that database. The monitor assigns ONLY ONE PART the special role of PRIMARY ID. The system ID (2.4.1) can be selected for this role, if desired. When the system lists resources in a database, they are listed by their primary ID rather than any other ID or combination of IDs.

2.4.5 TEXT PARTS

TEXT PARTS contain up to 200 lines of narrative text. Text is entered into the system using the full EIES scratchpad and may therefore contain any indirect edits for formatting the text. The database monitor sets an upper line limit on the size of the TEXT PART as part of FORMAT specification (2.4.11); see also (3.8) and (3.10). TEXT PARTS, like any other text item, can contain reviews, abstracts, descriptions, meeting notes, etc.

2.4.6 KEYWORD PARTS

Each resource can have associated with it any number of keyword descriptors. All such keywords are organized together into the KEYWORD PART(S) of a resource. The KEYWORD PART can be given any name the monitor wishes, such as "Keys," "Keywords," "Descriptors," etc.

Normally, keywords must be in the THESAURUS (2.3) before they they can be associated with a resource; however, the monitor can set the FREE KEYWORDS OPTION to allow any keywords to be used. The monitor also specifies the maximum number of keywords that can be associated with a resource part in the FORMAT (2.4.11).

Once entered into the database, keywords can be used in GET and FIND operations alone, in Boolean combinations, or in combinations with any other parts that are indexed (2.4.12). KEYWORD PARTS can contain any characters, including blanks, commas, special characters, and numbers. However, each keyword must begin with a letter to be entered as contents of a KEYWORD PART.

Each KEYWORD PART in the database can share its thesaurus with the thesaurus of another KEYWORD PART in the database or the thesaurus of a KEYWORD PART in another database. This allows for common links between parts and databases.

2.4.7 NUMBER PARTS

A NUMBER PART allows only the entry of a number in a range specified by the monitor in the FORMAT (2.4.11). Each resource can have NUMBER PARTS associated with it and all resources in a database must have exactly the same number of NUMBER PARTS. Whereas a keyword can be associated with a resource or not, a NUMBER PART must have a CONTENT WITH A SPECIFIC VALUE FOR EACH RESOURCE in the database.

However, in some cases the proper content for a NUMBER PART for a resource may not be known. In such cases, the system can be instructed to allow indication of this by recording UNKNOWN CONTENTS (3.5.4). The internal representation of unknown contents is 80000000 (hexadecimal) but is never used in any form of input or output and thus a user need not be concerned with it. On the other hand, the external representation to the user during data entry and retrieval is simply an empty or blank part denoting UNKNOWN CONTENTS. NUMBER PARTS can be used in GET and FIND operations, either singly or in Boolean combinations with other NUMBER PARTS and any other indexed part (2.4.12). Lists or ranges of contents (2.5.1) must be specified in such GET or FIND operations (age=20 or

age=19...39).

The contents of NUMBER PARTS are limited to combinations of the digits from 0 through 9, a decimal point and a minus (-) sign. The number of decimal digits following the decimal point can be specified by the monitor when adding the format. Dollar signs and other special characters are specifically excluded from being NUMBER PART CONTENTS.

2.4.8 TYPE PARTS

TYPE PARTS allow words to be used as unambiguously as numbers to describe a resource. In this sense, TYPE PARTS are similar to NUMBER parts. But while numeric "codes" can be established to record categorical information about a resource (such as "sex" or "type of community," the use of arbitrary numeric codes ("male=1, female=2") have no real meaning and are not ideal. Therefore, RESOURCES includes TYPE PARTS that allow the user to enter a TYPE NAME to a resource directly (e.g., "male" and "female"). Thus the contents of TYPE PARTS do not have to be numbers, but can be the TYPE NAMES themselves. Numeric codes are in fact used to represent TYPE PART contents internally, but these numbers are not used in any input or output operations. Just as all resources in a database must have the same number of NUMBER PARTS, they must have the same number of TYPE PARTS; and, like NUMBER PARTS, they allow the specification of UNKNOWN CONTENTS (3.5.4) and can be used in GET and FIND operations.

2.4.9 LABEL PARTS

A LABEL PART consists of a string of any printable characters up to a specific maximum length set by the monitor in the FORMAT (4.2.11). The only character that cannot be the contents of a LABEL is a semi-colon; semi-colons are used for answer-aheads. All resources in a database must have exactly the same number of LABEL PARTS. As with TYPE PARTS, LABEL PARTS are set to blanks if they have UNKNOWN CONTENTS (3.5.4). LABEL PARTS are generally used for information such as name, address, author, title, etc. The major difference between TYPES and LABELS is that the allowable names for types are defined by the monitor while virtually any string can be entered as a label. Because labels are usually unique to a resource they are not used the same way keywords, numbers or types are, where many resources have the same CONTENTS. They can be used in GET and FIND operations, however, if they have been INDEXED (2.4.12).

2.4.10 OPERATIONS ON PARTS

The operations that can be performed on these PARTS are ADD, REMOVE, and MODIFY. Whereas ADDING, REMOVING, and MODIFYING a resource in the database is done with commands as specified in data entry (3.2), the ADDING, REMOVING, and MODIFICATION of individual parts within a resource is done through the RESOURCE EDITOR (3.3) that uses special editing commands for creating and modifying the contents of specific resources.

2.4.11 THE DATABASE FORMAT

The database format, or FORMAT, defines a blank form of specified PARTS, into and under the control of which the various bits of information that describe a resource may be entered.

The FORMAT thus defines, for example, how many parts there are in a resource, how large the parts are, the kinds of parts that are used, the order in which they occur, and so on. This FORMAT information is used to manage and check new resource information during data entry. There is only one FORMAT for each database. The FORMAT is the responsibility of the database monitor.

The FORMAT contains a list of parts in the order they were first specified by the monitor. KINDS OF PARTS--LABELS, NUMBERS, TYPES, etc.--may be specified in any combination or order required. As each KIND OF PART is specified, it is given a PART NAME. Part names must begin with a letter or alphabetic character. Once the part name is specified in the FORMAT, it can be abbreviated to any number of characters that uniquely identifies that part name; but, if the abbreviation is not unique, the abbreviation is associated with the first part name matched from the FORMAT's list. Monitors will be advised to consider candidate lists of part names that recognize this ABBREVIATION feature (3.3.1.1) of the system by choosing logical and unique leading letters for part names.

In addition to part names, several other parameters become associated with the kinds of parts chosen for the resource. LABEL PARTS have a maximum length specified by the monitor. NUMBER PARTS have minimum and maximum contents, or any contents if minima and maxima are set to zero. Each TYPE PART must have its exact and limited list of allowable contents, or type names, specified. KEYWORD PARTS have upper limits for the number of different keywords associated with a single resource. TEXT PARTS need a maximum number of scratchpad lines specified. DATE-, LABEL-, NUMBER- and TYPE PARTS can be specified as indexed or unindexed (2.4.12). The TALLY features are also specified in the FORMAT, where tally questions are stored and vote response scales are defined (3.7)

The operations that can be performed on the format are ADD,

GET, MODIFY and TRANSMIT. ADD is used to specify the format for the database BEFORE resources are entered. GET prints the current format in effect for the database. MODIFY allows the monitor to modify the PART CHARACTERISTICS of the FORMAT. TRANSMIT (XMIT) allows the format to be exchanged with microcomputers (3.8.5). These operations on FORMAT are discussed throughout (3.8).

2.4.12 INDEXED PARTS

An INDEXED PART is a DATE PART, ID PART, KEYWORD PART, LABEL PART, NUMBER PART or TYPE PART for which there is both a relationship from its resource to an index and a relationship from that index to the resource that is compiled and maintained by the system. The ability to use this relationship for any part in the format--except TEXT PARTS--is specified when the format is added (3.8.1).

The use of such "inverted file" procedures speeds searches on the kinds of parts more frequently used for displays, gets, finds or finds-by-pattern. RESOURCES automatically maintains inverted indices for the keywords used in KEYWORD PARTS and for ID contents of ID PARTS; they are said to be INDEXED. DATE-, LABEL-, NUMBER- and TYPE PARTS may be optionally specified as indexed by the monitor if the purpose of the database requires they be frequently searched. Monitors will recognize significant savings in disk storage space by keeping indexed parts to a minimum.

The inverted indices created by indexing replicate the information in a resource in a manner that supports rapid searching and retrieval of resources. Indexing stores information more than one place on a disk; this is the reason for the trade-off between rapid searching and disk storage. Using keywords as an example, it is easier to keep an index of keywords and, for each keyword, an index of all the resources associated with that keyword. Resources then contain information "pointing to" keywords and the keyword index contains an "inverted" index "pointing back" to the resources.

After initial FORMAT specification, unindexed parts for which indexing is possible will be indexed if a user attempts to search on them. The user will be given the option of having the necessary index compiled overnight if the search is still desired.

Further discussion can be found in (2.5) and (3.5).

2.5 GETTING AND FINDING RESOURCES

RESOURCES may be retrieved (gotten) from a DATABASE with the GR (Get Resources) command by specifying the name of the PART the system is to use and the CONTENTS for which it is to look. The database must be UPDATED (3.2.4) to INDEX any new entries before they may be retrieved by any parts other than SYSTEM parts. For example:

```
GR,SYSTEM ID=983
```

will get the unique RESOURCE with a SYSTEM ID of 983 (the 983rd resource entered into the particular database). Some other examples include

```
GR,NAME=Johnson-Lenz, Peter and Trudy
GR,DATE=12/13/80...12/15/80
GR,KEYWORD=ENERGY
```

Thus, a PART specification includes two things: (1) the name of the PART (which may be abbreviated), and (2) the CONTENTS of the part (which may be a single CONTENT, a range of CONTENTS, or a list of CONTENTS and ranges of CONTENTS).

2.5.1 LISTS AND RANGES

CONTENTS in a list are separated from one another by semi-colons. RANGES are specified by the extremes separated by three periods (as in xxx...yyy). When either the beginning or end of a range is unknown, it may be left out and the system will automatically assume the smallest or largest CONTENTS. For example, SYSTEM ID=...13 refers to resources with SYSTEM ID from 1 to 13. Similarly, DATE=12/13/80... refers to all resources dated 12/13/80 up to the present. The form ID=... refers to ALL resources because the entire range has been implied in the specification. Below are further examples of PART specifications used with the GR command involving lists and ranges.

```
GR,SYSTEM ID=913...1004
GR,SYSTEM ID=123,159,177,219
GR,SYSTEM ID=213...220,230,240...256
GR,DATE=12/12/80...12/14/80,12/17/80,12/20/80...
```

2.5.2 PART SPECIFICATIONS

A PART specification can involve DATE, ID, KEYWORD, LABEL, NUMBER, and TYPE PARTS. SYSTEM PARTS can also be specified as if they were regular FORMATTED parts. Below, the specifics of each of these KINDS of PART specifications are detailed. The general form of a PART specification is PART NAME=CONTENTS. Two alternate forms are also acceptable: PART NAME:CONTENTS and PART NAME;CONTENTS.

A DATE part consists of the PART NAME (such as DATE, SYSTEM DATE, DATE PUBLISHED, or whatever the particular DATE PART is called) followed by an equal sign, semi-colon, or colon followed by a specific date, range of dates, or list of dates and ranges. The form DATE= can be used to specify all UNKNOWN dates.

An ID part consists of the PART NAME (such as PRIMARY ID, ID, etc.) followed by an equal sign, semi-colon, or colon followed by a specific ID, range of IDs, or list of IDs and ranges. Since UNKNOWN IDs are unacceptable to the system (all IDs must be unique), the form ID= is not allowed. The system always capitalizes the alphabetic portion of an ID part CONTENTS to avoid confusions between upper and lower case versions of the same ID.

A KEYWORD PART consists of the KEYWORD PART NAME followed by an equal sign, semi-colon, or colon followed by a keyword or a range of keywords. Keywords cannot be specified in a list the same way most other PART CONTENTS can be, because they can contain commas and a list of keywords can identify a single resource. If a list of keywords is to be specified, each keyword must be separated by a semi-colon. Because KEYWORDS have identification numbers (3.4.5), they can be specified by these numbers as well as by name. KEYWORDS may be abbreviated when used in range specifications. Otherwise, an exact match for the keyword entered by the user is intended.

A LABEL PART consists of the PART NAME (such as NAME, ADDRESS, etc.) followed by an equal sign, semi-colon, or colon followed by a string of characters (the actual name, address, etc.) or a range of contents. Because LABELS can contain internal commas, they CANNOT be specified in a list. If a list of LABELS is to be specified, each part contents must be separated by a semi-colon. If label CONTENTS are abbreviated, all resources with CONTENTS matching the abbreviation will be specified. In finding a match with the contents of a LABEL, the system considers upper and lower case letters to be equivalent: no distinction is made between upper and lower case letters during get and find operations. During indexed look-up, the system uses the first two characters as the most significant search specification, then uses the rest of the string. Therefore, at

least two characters should be entered when looking for resources by label contents. The form NAME= or ADDRESS= can be used to specify all resources in which the contents are UNKNOWN.

A NUMBER PART consists of the PART NAME followed by an equal sign, semi-colon, or colon followed by a NUMBER, range of NUMBERS, or list of NUMBERS and RANGES. The form NUMBER= can be used to specify all resources in which the contents are UNKNOWN.

A TYPE PART consists of the PART NAME followed by an equal sign, semi-colon, or colon followed by a TYPE NAME or list of TYPE NAMES. Since the categories represented by TYPE NAMES do not represent any ordering, the range specification cannot be used with TYPE PARTS. The special form TYPE= can be used to specify all resources in which the contents are UNKNOWN.

2.5.3 PART-SPECIFIC PROMPTS

In most cases, when the system is asking the user for a PART specification for getting resources and the user enters only the PART name without any equal sign, colon, or semi-colon after the name, and without any CONTENTS following the name, the system will prompt the user for the contents using a PART-SPECIFIC prompt based on the name and KIND of PART entered by the user. Below is a table showing the PART-SPECIFIC prompts that the system will use in requesting specification of the CONTENTS from the user. Note that the prompt encourages the user to enter either a single CONTENT, or a list of contents (except for KEYWORDS and LABELS) and ranges of contents. Note also that the system does not allow ranges to be specified for TYPE PARTS, because they have no meaning.

KIND OF PART	PROMPT
-----	-----
DATE	[DATE part name] (mm/dd/yy,mm/dd/yy...mm/dd/yy)?
ID	[ID part name] (abc#123,ijk#456...xyz#789)?
KEYWORD	[KEYWORD part name] (keyword;keyword...keyword)?
LABEL	[LABEL part name] (label;label...label)?
NUMBER	[NUMBER part name] (#,#...#)?
TYPE	[TYPE part name] (type,type)?

2.5.4 PATTERNS

A DATABASE may also be scanned to FIND RESOURCES that match a certain complex pattern of parts with either the FR (Find Resources) command (3.5.9), or the FP (Find resources by Pattern) command (3.5.10).

A complex pattern of PARTS used for finding resources is called a PATTERN and consists of a series of PART specifications (2.5.2) that collectively make up the PATTERN. The PART specifications are separated from one another by semi-colons (to differentiate between lists within PART specifications and lists of PART specifications). In addition, PARTS after the first part of a PATTERN begin with a Boolean operator (& for AND, / for OR, and - for NOT) that specifies how the lists of RESOURCES defined by each part are to be combined into the final, complete PATTERN. Thus a PATTERN is a list of PART specifications separated by semi-colons and followed by Boolean operators (&, /, or -), or in general:

PATTERN = PART;(operator)PART;(operator)PART;(operator)PART

The list of Boolean operators includes:

AND represented by the character &
(ampersand)

OR represented by the character / (slash)

NOT represented by the character - (minus
sign)

Here are some examples of acceptable PATTERNS:

KEYWORD=ENERGY;/KEYWORD=SOLAR (resources keyed ENERGY
or SOLAR)

KEY:ENERGY;&KEY:CONSERVATION (resources keyed ENERGY
and CONSERVATION)

KEYWORD=COMMUNICA...COMMUNITY (resources keyed with
any keyword in the range COMMUNICA to COMMUNITY --
the system will display all keywords in the range and
ask which one the user wishes to use)

KEY:COMM (resources keyed with any keyword beginning
with the four letters COMM -- the system will display
all keywords which match the letters COMM and ask
which one the user wishes to use)

K:ENERGY;&POPULATION:...10000 (resources keyed ENERGY
and with POPULATION up to 10000)

DATE;12/12/80...;-ID=234...250 (resources dated
12/12/80 and later, but NOT those with IDs in the
range 234 to 250)

AGE=21...65 (resources with AGE in the range from 21
to 65)

SEX=MALE;&NAME=a...c (all males whose names begin
with the letters a through c).

NAME=Johnson (all resources with name beginning with Johnson)

Although not shown in the above examples, blanks may be included before and after Boolean operators and the =, :, or ; delimiter. Thus DATE;12/12/80...; - ID = 234...250 will also work.

2.5.5 RECURRENT FINDS WITH PATTERNS

The system allows a user to store frequently used PATTERNS for later use. Each such stored PATTERN is assigned a unique (to the user) identification name by the user, such as PATTERN1 or THE USUAL. Then later on the user may refer to the pattern by this unique name; thus if PATTERN1 were defined as K=ENERGY;&K=CONSERVATION for a user, he or she could find resources with that pattern by using the FP (Find resources by Pattern) command instead of the normal FR (Find Resources) command. Each PATTERN has an associated MARKER that shows the last resource processed by that PATTERN. This allows scanning of the most recent portions of the database only. See Section (3.5) for discussion of how to find resources interactively and (3.5.10) for details about commands used to add, modify, remove, and find resources using PATTERNS and PATTERN MARKERS.

When a list of resources that matches a pattern is found it is always sorted from MOST RECENT to LEAST. Since the PATTERN find includes only resources more recent than the user's PATTERN MARKER, previously found resources will not be printed again. This makes it easy to keep up to date in a specific category without having to keep track manually of which resources have been seen and which have not.

Thus the basic operations of ADD, REMOVE, DISPLAY, GET, and MODIFY are available through the command interface for stored PATTERNS. These are detailed in "Data Entry/Modification Commands" (3.2).

2.5.6 TEMPLATES

Normally, the system prints out all the PARTS of resources when the Get and Find commands are used. However, it is often useful to selectively print out only some of the PARTS of resources--either for preparation of special lists of resources, or to limit the amount of printout received. To make this possible, the system includes the concept of the OUTPUT TEMPLATE, which is a list of PARTS the user wishes to print out. The TEMPLATE also determines the ORDER in which PARTS are printed out for a resource.

The TEMPLATE is an output feature (3.5.11) that governs the order and selection of all PARTS output by GET or FIND commands, as well as any other commands that normally print out complete resources in STANDARD OUTPUT FORMAT.

The TEMPLATE is normally set to print out all PARTS of resources. This is called the DEFAULT TEMPLATE. Rather than bothering to specify all PARTS in such a complete TEMPLATE, the system understands that a completely empty or null template specification means that it is to print out all PARTS. Any specific list of PARTS in the TEMPLATE then means that only those specific PARTS are to be printed in the specified order. A new user need not be concerned about the TEMPLATE because the DEFAULT TEMPLATE is normally in effect.

Two commands have been provided for management of the user's TEMPLATE. The GTE (Get Template) command (3.5.11.2) lists the current TEMPLATE. The MTE (Modify Template) command (3.5.11.1) allows re-specification of the PARTS in the TEMPLATE or resetting the TEMPLATE to the default full printout.

When a non-default TEMPLATE is in effect, some PARTS of resources may not be printed out. Therefore, to make sure the user is aware of this, a special warning message is issued by the system whenever a user invokes a command that uses the TEMPLATE and the TEMPLATE does not specify printout of all PARTS. This message says "NOTICE: YOUR TEMPLATE CURRENTLY SPECIFIES PARTIAL RESOURCE PRINTOUT." This template in effect notice is discussed further in (3.5.11.3).

TEMPLATES do not override LAYOUTS. If a layout is active for a user, it will be used. If there is no layout active for that user, then the template will be used for resource printing. Section (3.5.11) contains more information on templates and layouts.

2.5.7 DELIVERY OF MOST RECENT RESOURCES

The command PR (Print Resources waiting) (3.5.1) may be used to retrieve all new resources since you last used the PR command. Your RESOURCE MARKER is updated to show which resource you last received. You can change your resource marker using the MM (Modify Marker) command (3.7.9). Working like a conferencing system, users can pick up new resources from the database in the order they were added.

2.5.8 LAYOUTS

LAYOUTS is an output feature (3.5.11) that allows you freedom in formatting resource output, e.g., to compose forms, mailing lists and reports; you can specify where the part is to appear in a page or form. Text and keyword parts must be the only item on a line, but other format parts may appear in the middle of text and inside of indirect editing commands.

The LAYOUTS are set up just like a text item. You can have local and global LAYOUTS in the database. Operations performed on layouts include ADD, MODIFY, REMOVE, GET and UPDATE.

2.6 TALLIES, COMMENTS AND MARKERS

When a user GETS or FINDS a resource in the database, an opportunity is provided to add a comment on the resource. Comments can include evaluative material or a critique of the resource that can then be used by others interested in the same resource. In addition, the monitor can assign any number of questions and numeric rating scales to a DATABASE as part of the FORMAT. Users can enter ratings on these scales in response to the questions for resources they have received. These TALLIES of ratings can then be used as a concise form of evaluation data on resources. Tallies, as well as comments and markers, are discussed in Feedback Commands (3.7).

Both COMMENTS and TALLIES are associated with specific RESOURCES in the database. When a COMMENT is added, the system asks for identification of the resource. Any unique part specification may be used. If desired, COMMENTS may be associated with NO resource and, hence, general.

COMMENTS are sequentially numbered within a DATABASE to facilitate review of this feedback by a monitor or other user wishing to review ALL comments added to a DATABASE. Thus all comments for a database may be gotten or only those comments for a particular resource may be gotten, depending on the user's interests. To facilitate review of these comments, each user has a MARKER for each DATABASE. Each such MARKER identifies the most recent comment read so that subsequent delivery of comments can include only new ones.

The following operations can be performed on COMMENTS: ADD a comment, REMOVE a comment, DISPLAY the header of a comment, GET an entire comment, MODIFY a comment, COPY a comment into the scratchpad and PRINT all waiting comments in a DATABASE.

The following operations can be performed on a TALLY: ADD a response to a tally, MODIFY a previous response to a tally, DISPLAY the results of a tally for a resource and GET the results of a tally for a resource. TALLIES are associated with each resource and established by the monitor as part of the FORMAT specification. Tally questions and tally scales are created, modified and removed by the database monitor using the MODIFY TALLY command. However, as with comments, the user may get the results of tallies for specific resources or may simply get the results of all tallies.

The only operation that can be performed on a MARKER by a user is the MODIFICATION of that marker, that is the moving of its "position" forward or backward (3.7.9). MARKERS are automatically created by the system; they are removed when either the user or the database is removed.

2.7 ACTIVITY TRACE: USER ACTIVITY AND BILLING INFORMATION

The system keeps records of user activity. For each RESOURCE and COMMENT a trace record is kept of how many times it has been accessed. A record of activity is kept for each USER showing how many resources and comments have been added, gotten or found. These counts are maintained automatically by the system.

However, in addition to these counts, a more detailed trace of a user's activity is kept by the system if the ACTIVITY TRACE OPTION is turned on by the database monitor. When this option is turned on, the system will record the date and time, command used, and PRIMARY ID of the resource or comment any time a command is used which references a resource or comment. Any user may access this trace of his or her own activity. A gatekeeper may access the trace for any user. These traces may be used for billing purposes where appropriate. The activity trace can be turned off or on at any time by the database monitor by modifying the database options.

The only operations that can be performed on a USER'S trace file are GET and MODIFY. User management commands are discussed in (3.9).

CAUTION: Activity trace files can only hold about 650 records. Database monitors should keep up with very active users and prune their activity files (3.9.7). The system will exclude a user from any database where the trace files for that user are full.

2.8 MONITOR OPTIONS

Each database has associated with it several MONITOR OPTIONS which can be modified by the monitor. These options govern certain functional characteristics of the software. Each of these options has a DEFAULT setting that will be used if the monitor makes no decision on options.

The first option is the PRIVATE DATABASE option. It can be set to either PRIVATE or PUBLIC. The default setting is PRIVATE. When set to PRIVATE, only users specifically given access to the database by the gatekeeper will be able to get in. When set to PUBLIC any user of a regular EIES account or subaccount will be able to use the database. However, when set to PUBLIC, the monitor must specify a DEFAULT ACCESS LEVEL to be automatically assigned to users coming in publicly.

The second option is the PEN and ANONYMOUS COMMENTS option. It can be set to either allow or disallow PEN and ANONYMOUS COMMENTS. The default setting is to allow them. NOTE that the PEN names are NOT unique. The system does no checking. If the user answers N to OK TO ADD (Y/N)? the system asks PEN NAME? and the user enters a PEN NAME (or ANONYMOUS). This allows subaccount users to author PEN and ANONYMOUS comments. See Section (3.7.2) for details of use.

The third option is the TRACE option. It can be set to either trace all users' activity or not. If it is set to trace all users' activity, the system will record an entry in the user's activity trace file (printed out by the GA command) each time the user executes a command, and each time a resource or comment is entered into or retrieved from the system. The entry in the file includes the command used, the date and time of the activity and the PRIMARY ID of the resource or the COMMENT ID of the comment involved. Even if the trace option is not turned on, the system automatically increments counters for each member, resource and comment every time an item is added, gotten, found or otherwise retrieved.

The fourth option is the PROMPTED MODE option. Users will note it can be set to make the RESOURCE EDITOR (3.3) go automatically into PROMPTED MODE (3.3.3) when the AR command is used; or it may be set so that the RESOURCE EDITOR goes into direct mode when the AR command is used. The default is PROMPTED MODE. The user may override this setting with the +DIRECT or +PROMPTED commands (3.11.1).

The fifth option is the UPDATE option. It can be set to have the database automatically updated immediately after a resource has been entered or modified, or to delay such updating any number of days from overnight to weekly or monthly. The default is overnight. See also (3.2.4).

The sixth option allows the display of EMPTY RESOURCE PARTS

during resource gets and displays. The default is to display the empty parts. If the option is set, resource parts that are blank will NOT be included in the printout when the default template or a layout is used. This is to assure that only parts with known contents are included in the printout and that blank parts are suppressed.

These MONITOR OPTIONS can be modified only by a monitor (access level 50) with the MD (Modify Database) command (3.10). Any user may display the current settings of these options with the GD (Get Database) command.

3.0 THE RESOURCES USER COMMAND INTERFACE

The command interface to RESOURCES includes all the commands that are defined and outlined in this section. These commands have different levels since some require more authority to use than others. These access levels can be broken down into several categories such as DATA ENTRY, DATABASE MONITORING, GATEKEEPER, READ ONLY and ADMINISTRATOR commands. These access level definitions are presented in the section on ACCESS (section 2.2).

3.1 GENERAL RESOURCES COMMAND FEATURES

To get access to a database in the RESOURCES system, an EIES user enters the +RESOURCES command from anywhere in EIES. The system then responds by asking the user to enter the name of the database. If the user has access to the database, it will print a welcome, display the number of resources and comments in the database, and then ask COMMAND? This is shown in the example, below:

```
INITIAL CHOICE?+resources
DATABASE?example
```

```
Welcome to the EXAMPLE database.
As of 12/12/80 EXAMPLE contains 439 resources and 295 comments.
```

```
COMMAND?
```

If a single question mark (?) is entered at DATABASE? a list of all the public databases in the RESOURCE system will be displayed.

3.1.1 COMMAND MENU AND DESCRIPTIONS

When the system asks COMMAND?, you may begin interacting with the database in any of the ways allowed under your ACCESS LEVEL (see Section 2.2) by entering a two- or three-letter command from the list of commands shown in Section 4. Each of these commands is made up of two (or more) letters -- one letter taken from the list of verbs (operations) and the rest from the list of nouns (operands) shown at the beginning of Section 4. If a user presses RETURN at COMMAND? the system will print out this menu with the verb and noun columns. If a user enters a single question mark (?), the system

will print out a list of commands which can be executed at his or her level of access. The table of commands shown in Section 4 lists the minimum level of access required to execute each command listed. These levels are used by the system both to prune the list of operational commands printed in response to ? and to determine if a command entered by a user can be executed by him/her.

If you need an explanation of how a command works, you can type HELP or ?? at the COMMAND? prompt. The system will then ask you for EXPLANATION FOR WHICH COMMAND?. Here you can enter any valid command and the system will retrieve the section of this manual that is indexed by the phrase you type in.

3.1.2 RETURNING TO EIES

Following execution of user entered commands, the system will again ask for COMMAND? and continue to do so until a user enters ++ to return to EIES. When ++ or -- is entered, the system responds with a reminder message that the user is leaving the database, as shown in the example below:

```
COMMAND?++
```

```
Leaving the EXAMPLE database.
```

```
NO MESSAGES WAITING
```

```
INITIAL CHOICE?
```

In the following sections the full array of commands is specified in greater detail.

3.1.3 ANSWERING AHEAD CONVENTIONS

The RESOURCES system supports answering ahead by separating successive commands or other inputs from each other by semi-colons. The comma is not generally used as an answer-ahead character since it is allowed in LABEL and KEYWORD part CONTENTS. Those few cases where commas are allowed as an answer-ahead character are explicitly specified.

3.1.4 ACTIVITY TRACE BY COMMAND

If the activity trace option has been turned on by the database monitor, the system will make an entry in the user's activity trace file (printed out with the GA command -- see Section (3.9.6) for details) for each command used. In addition, if the command used adds or gets more than a single item, an entry will be made for each resource and or comment added or gotten.

3.2 DATA ENTRY/MODIFICATION COMMANDS

Three commands may be used by users with access level 20 or above to add, modify, and remove resources from the database. The AR (Add Resource) command is used to add a resource new to the database. The MR (Modify Resource) command is used to modify an existing resource. The RR (Remove Resource) command is used to remove an existing resource from the database.

3.2.1 ADDING A RESOURCE TO THE DATABASE (AR)

When the AR command is entered, the system checks to see if the user has at least level 20 access; if so, it responds by taking the user into the RESOURCE EDITOR (3.3) for entry and editing of the resource before it is filed.

After the user has completed entering and editing the resource, the RESOURCE EDITOR is terminated by entering a +. The system will then ask a double-check question, OK TO ADD TO [DATABASE NAME] (Y/N)?

At this point, if the user enters N the resource will NOT be added and the system will ask DO YOU WISH TO MODIFY THIS RESOURCE (Y/N)?. If the user again answers N, the system will respond with RESOURCE NOT ADDED and clear the resource editor. Then, it will ask ADD ANOTHER RESOURCE (Y/N)? If the user enters N again, the system will go back to COMMAND? level. If Y or (carriage return) is entered, the system will go back to the RESOURCE EDITOR with a clean workspace. If Y is the response to the DO YOU WISH TO MODIFY THIS RESOURCE (Y/N)? prompt, then the user will be taken back into the RESOURCE EDITOR with the resource still loaded for DIRECT mode editing.

However, if the user enters a Y or presses RETURN in response to the initial OK TO ADD TO [DATABASE] (Y/N)? question, the resource will be assigned the next sequential SYSTEM ID and will be added to

the database. Then the system will ask ADD ANOTHER RESOURCE (Y/N)? If the user enters Y or presses (carriage return), they will be returned to the RESOURCE EDITOR and given the opportunity to add another resource. If they enter N, they will be taken to COMMAND?

The AR command checks to see if the trace option is on and, if so, makes an entry in the user's trace file. Even if the trace option is not on, the count of resources in the database and the count of resources added by that user will be incremented by one. See Section (3.3.1.4) for more details on the AR command, including examples.

3.2.2 MODIFYING AN EXISTING RESOURCE (MR)

When the MR command is entered, the system responds by asking RESOURCE TO BE MODIFIED? The user answers this question by entering a PART specification (2.5.2) that uniquely defines the resource to be modified. All the standard user-support features specified for the GR command (3.5.2) are also operational at this input. This includes part-specific prompts, feedback of part names in response to ?, and feedback of keywords within a range or matching an abbreviation. If the user is either the author of the resource or has EDITOR (level 30) access, the system prints out the one-line ID or header of the resource and goes into the RESOURCE EDITOR with the resource already loaded into the editor workspace. The RESOURCE EDITOR commands (3.3.1) may then be used to modify the resource. When modification is done, the user enters a + and the system files the resource after printing out the SYSTEM ID line of the resource and asking the double-check question OK TO MODIFY RESOURCE (Y/N)?

Below is an example of modifying a resource

```
COMMAND?mr
RESOURCE TO BE MODIFIED?name=Peter+Trudy Johnson-Lenz
```

```
EXAMPLE 432 6/19/80 12:12 PM SOURCE: 118
```

```
BEGIN MODIFYING RESOURCE:
```

```
?:add
ADDRESS: 695 Fifth St, Lake Oswego, Oregon 97034
?add:695 Fifth Street, Lake Oswego, Oregon 97034
?+
```

```
OK TO MODIFY RESOURCE (Y/N)?y
RESOURCE MODIFIED AS:
```

```
EXAMPLE 432 6/19/80 12:12 PM SOURCE: 118
MODIFIED: 7/22/80 4:35 PM
```

```
COMMAND?
```

NOTE: the SYSTEM ID line is modified by the MR command to show the date/time the resource was LAST modified. Modification of a previously modified resource merely updates this additional date/time.

3.2.3 REMOVING A RESOURCE FROM THE DATABASE (RR)

When the user enters the RR command, the system responds by asking RESOURCE TO BE REMOVED? The user answers by entering a PART specification (2.5.2) that uniquely defines the resource to be removed. All the standard user-support features specified for the GR command (3.5.2) are also operational at this input, including part-specific prompts, feedback of part names in response to ? and feedback of keywords within a range or matching an abbreviation. If the user is the author of the resource or if the user has EDITOR (level 30) access, the system will print out the SYSTEM ID line of the resource, ask OK TO REMOVE RESOURCE (Y/N)? and then remove the resource in its entirety from the database.

When a resource is removed from the database, all comments associated with it and all references to it in the THESAURUS are removed; and the count of total resources in the database is decremented by one.

Below is an example of removing a resource from a database:

```
COMMAND?rr
RESOURCE TO BE REMOVED?name=Peter+Trudy Johnson-Lenz
```

```
EXAMPLE 432 6/19/80 12:12 PM SOURCE: 118
```

```
ID: P+T#999
NAME: Peter+Trudy Johnson-Lenz
DATE: 5/18/80
```

```
OK TO REMOVE RESOURCE (Y/N)?y
RESOURCE REMOVED.
```

```
COMMAND?
```

If the part specification matches more than one resource in the database, RR will loop and the user will first be shown the display parts of the resource to be removed, then the system will ask OK TO REMOVE (Y/N)?. Responding with No will leave the resource and continue the removal loop, Yes, or (carriage return) will remove the resource in question, and + will terminate the loop and return the user to COMMAND?.

3.2.4 FILE INVERSION: UPDATE DATABASE (UD)

Any time the three data entry/modification commands listed in (3.2) are used, THE DATABASE MUST BE UPDATED before the newly-added or -modified resources can be accessed with other commands in the system. This is because the system maintains all INDEXED parts (2.4.12) in both normal and inverted list forms on disk to facilitate rapid retrieval. The time overhead makes updating the inverted files during resource modification prohibitively slow. Therefore, updates are not done until resource entry and modification is complete.

The actual UPDATING is done by a BACKGROUND TASK that periodically checks and updates the database. The frequency with which this task runs is determined by the UPDATE MONITOR OPTION (3.2.4). If that option is set to zero, the task will begin executing immediately following entry/modification of a resource. If it is set to 1, the task will begin executing late that same night. A setting of 2 means the task will begin the next night, etc.

If the option is set to 1 or more, a user with level 20 access may use the UD command directly, triggering immediate execution of the task and updating the database as rapidly as possible. Once the UD command is executed by a user, the user is free to log off EIES. The file inversion will be done by the background task (generally overnight) so that the newly entered and modified data will be accessible the next day.

When entered at COMMAND? the UD command will ask the user as a double-check OK TO UPDATE (Y/N)? Once the background task has been submitted in this way it will perform the update and then send a brief EIES message to the database monitor under whose ID the UPDATE option was last set.

3.2.5 UPDATING RESOURCES (UR)

The UR command (Update Resource) allows users to update already existing resources in the database. The UR command is like the MR (Modify Resource) command; however, there are two significant differences:

First, Update Resource allows users to resume data entry on a partially-completed resource. UR finds the last part that was filled in and prompts for next part(s). This can be useful for databases where Add Resources work must begin while data are being collected and the full database format is not known.

Second, the resource is not copied into the resource editor. And if it has already been indexed, each part entered will be indexed as it is added. This takes longer for each part entered than AR or MR. The functions `::` and `:-` are available to list all the resource parts (3.3.1.4).

When a part is entered, it immediately over writes the the previous contents. There is no double check question so if a mistake is made the contents will have to be overwritten again. Below is an example of Update Resource:

```
COMMAND?ur
RESOURCE TO UPDATE?system=432
```

```
TYPE?subr
ENTER KEYWORDS -- END WITH +
KEYWORDS?networking;computers;+
ENTER DESCRIPTION -- LIMIT 20 LINES -- END WITH +
1?.text
2? P+T are independents consultants who work out of their
3?home office in Lake Oswego where they "teleconsult" and live
4?the life "electronic cottage" workers predicted by
5?Alvin Toffler.
6?+
```

```
COMMAND?
```

This example shows a resource added and left incomplete, with parts on the end blank. UR picked up at that point, continuing to prompt for data to complete the resource.

The next example shows how UR looks when the resource to be updated has data in its last position:

```
COMMAND?ur
RESOURCE TO UPDATE?system=432
```

```
EXAMPLE 432 6/19/80 12:12 PM SOURCE: 118
MODIFIED: 6/22/80 1:43 AM
```

```
PART?name=Peter and Trudy Johnson-Lenz
PART?+
```

```
COMMAND?
```

3.3 THE RESOURCE EDITOR

Once invoked by either the AR or MR command, the RESOURCE EDITOR allows direct entry or modification of the many parts of a resource. If the use of AR or MR is interrupted for any reason, the resource editor will retain its contents, remaining full. Whenever

this has happened, the system will prompt RESOURCE EDITOR NOT EMPTY, ERASE (Y/N)? before the user may enter it again.

When the resource editor is invoked by the MR command, it begins with the workspace filled with the resource to be modified.

3.3.1 RESOURCE EDITOR COMMANDS

When invoked, the RESOURCE EDITOR responds with the =? prompt. In response to this prompt a user may enter any of a wide variety of RESOURCE EDITOR commands as detailed below. Note that individual parts may be referred to within these commands by either the name of the part or its abbreviation. Abbreviations must be an exact match on one of the part names starting with the first character of the name and matching through consecutive characters until a complete match with the abbreviation is entered. In cases where an abbreviation may match more than one part, that part specified FIRST in the FORMAT is considered to be the part referenced by the abbreviation.

RESOURCE EDITOR COMMANDS

+ to end editing and file resource away (system asks double-check question); the + command is also used to terminate entry or modification of a TEXT or KEYWORD part.

- to return to COMMAND. The resource is left loaded in the resource editor.

:- to list the entire resource in INPUT format (see Section 3.1.3 for details)

:: to list the entire resource in OUTPUT format (see Section 3.1.3 for details)

? (or just RETURN) to get a brief explanation of RESOURCE EDITOR commands

:PART to list a specific part only (uses standard INPUT format like :-)

PART=CONTENT to enter or modify contents of a DATE or LABEL PART

PART= or -PART to blank a specific LABEL PART

PART=CONTENT to enter or modify content of an ID or NUMBER PART

PART=TYPE NAME to enter or modify content of a TYPE

PART

PART= or -PART to set the content of a specific ID, NUMBER, or TYPE PART to UNKNOWN CONTENTS

(NOTE: the alternate form PART; may also be used instead of the form PART= shown above. If only the PART NAME or abbreviation is entered, the system responds with the PART-specific prompt. For example, if a DATE PART NAME is entered, the system asks DATE (mm/yy/dd)?. See the section on PART-SPECIFIC PROMPTS for more detail.)

KEYWORDS (or whatever the name or abbreviation for the KEYWORDS PART is) to begin entering KEYWORDS using as many lines as needed and terminating with a +; the system responds with ENTER KEYWORDS -- END WITH + and then asks KEYWORDS? until + or (carriage return) is entered. In this KEYWORD mode, the following commands may be used:

KEY (where KEY is either a KEYWORD, abbreviation, or keyword ID # as per Section 3.2) to enter a KEYWORD (NOTE: only the name or abbreviation of the keyword need be entered to enter a KEYWORD into a resource)

-KEY (where KEY is either a KEYWORD, abbreviation, or keyword ID # as per Section 3.2) to remove a KEYWORD

? (or simply pressing return in response to the KEYWORDS? prompt) to get a compressed list of keywords in the thesaurus

The KEY and -KEY commands may be entered several per line, separated by semicolons.

PART to invoke the scratchpad for entry or modification of TEXT item; the system responds with ENTER [name of PART] -- LIMIT ## LINES -- END WITH + and then asks for lines of text until a + is entered. In this TEXT mode, ALL EIES SCRATCHPAD commands may be used, including < and > as well as all indirect edits.

+DIRECT to change from prompted to direct mode (see PROMPTED MODE, below).

+PROMPTED to change from direct to prompted mode.

(NOTE: RESOURCE EDITOR commands may be answered-ahead with semi-colons in between. Only the semi-colon will work for this since commas may be

included in LABEL and KEYWORDS. Such answering ahead cannot be done WITHIN a TEXT item where semi-colons as well as commas may be included in the text.)

3.3.1.1 ABBREVIATIONS: THE FIRST-FOUND RULE

A general rule is used to determine which abbreviations can be used when entering PART NAMES, KEYWORDS, and TYPE NAMES for TYPES. Because all of these are entered as character strings, it is useful to allow the user to enter abbreviated forms as well as complete strings. The rule used by the RESOURCE EDITOR is that any abbreviation is compared with the list of names to be scanned and the FIRST string which matches the abbreviation exactly--letter for letter, even if the string is itself longer than the abbreviation--is considered the match. Thus a single letter can be used for abbreviations, as can any number of letters. In cases where an abbreviation matches more than one string, the FIRST string found to match is used. PART NAMES, KEYWORDS and TYPE NAMES are all searched in alphabetic order. Thoughtful choice of PART NAMES, KEYWORDS, and TYPE NAMES can take wise advantage of this feature.

3.3.1.2 SPECIAL USER SUPPORT FOR ENTRY OF KEYWORDS

When the KEYWORDS command has been entered and KEYWORD entry is proceeding, or when in PROMPTED MODE (see below) the KEYWORD? prompt is given to a user, the user may enter a single question mark (?) or simply press RETURN to get a compressed list of all keywords in the thesaurus printed out to assist in selection of keywords.

If the user enters a keyword that is not in the thesaurus, and if the fixed keyword option has been turned on, the system will respond with "KEYWORD X NOT IN THESAURUS -- ENTER ? TO GET LIST OF KEYWORDS."

If the user enters a USE INSTEAD keyword (2.3) that cannot be used as a keyword, the system will respond with "USE Y INSTEAD OF X." where X was entered by the user and Y is the USE FOR keyword associated with X.

Keywords can be entered with leading blanks for readability. Such leading blanks are removed by the system.

3.3.1.3 PART-SPECIFIC PROMPTS

In DIRECT mode if a user enters only a PART NAME, the system will respond by asking for the CONTENTS appropriate for the particular part. These are the same prompts used by the EDITOR in PROMPTED mode. Below is a list of the part-specific prompts the system will ask:

KIND OF PART	PROMPT
-----	-----
DATE	[DATE part name] (mm/dd/yy)?
ID	[ID part name] (letters#numbers)?
LABEL	[LABEL part name] (up to [maximum] chars.)?
NUMBER	[NUMBER part name] (#)? (if no range specified) [NUMBER part name] ([minimum]...[maximum])? (if range specified)
TYPE	ENTER [TYPE part name]: (list of TYPE NAMES separated by slashes) [TYPE part name] (TYPE NAME)?

3.3.1.4 EXAMPLE INTERACTION WITH THE RESOURCE EDITOR

Here is an example interaction with the RESOURCE EDITOR creating a new resource. (Note the interaction is in direct mode).

```

COMMAND?ar
=?id=p#303
=?date=5/18/80
=?description
ENTER DESCRIPTION -- LIMIT 20 LINES -- END WITH +
1?.text
2? P+T are independent consultants who work out of their home
3?office in Lake Oswego where they "teleconsult" and live the
4?life of "electronic cottage" workers predicted by futurist
5?Alvin Toffler.
6?+
=?na=Peter+Trudy Johnson-Lenz
=?add=695 Fifth Street, Lake Oswego, Oregon 97034
=?ph=(503) 635-2615
=?occ=27;PEOPLE=2;POPULATION=20000;TYPE=SUBURBAN
=?key
ENTER KEYWORDS -- END WITH +
KEYWORDS?networking;computers
KEYWORDS?transformation;neighborhoods;+
=?:-
ID=P#303
DATE= 5/18/80
NAME=Peter+Trudy Johnson-Lenz
ADDRESS=695 Fifth Street, Lake Oswego, Oregon 97034

```

PHONE=(503) 635-2615
 OCCUPATION=27
 PEOPLE=2
 POPULATION=20000
 TYPE=SUBURBAN
 KEYWORDS
 NETWORKING;COMPUTERS
 TRANSFORMATION;NEIGHBORHOODS
 +
 DESCRIPTION
 .text
 P+T are independent consultants who work out of their home
 office in Lake Oswego where they "teleconsult" and live the
 life of "electronic cottage" workers predicted by futurist
 Alvin Toffler.
 +
 =?::

ID: P#303
 DATE: 5/18/80
 NAME: Peter+Trudy Johnson-Lenz
 ADDRESS: 695 Fifth Street, Lake Oswego, Oregon 97034
 PHONE: (503) 635-2615
 OCCUPATION: 27
 PEOPLE: 2
 POPULATION: 20000
 TYPE: SUBURBAN

KEYWORDS:
 NETWORKING; COMPUTERS; TRANSFORMATION; NEIGHBORHOODS

DESCRIPTION:
 P+T are independent consultants who work out of their home
 office in Lake Oswego where they "teleconsult" and live the life
 of "electronic cottage" workers predicted by futurist Alvin
 Toffler.

=?+
 OK TO ADD TO EXAMPLE (Y/N)?y
 ADDED AS:
 EXAMPLE 432 6/19/80 12:12 PM SOURCE: 118

ADD ANOTHER RESOURCE (Y/N)?n

COMMAND?

3.3.1.5 CHECKS PERFORMED BY THE RESOURCE EDITOR

The RESOURCE EDITOR performs the following checks on resources being entered into a database:

DATE PARTS are checked for proper mm/yy/dd form.

ID PARTS are checked for proper string#number form, and are also checked for uniqueness within the database. The RESOURCE EDITOR will not accept a duplicate ID.

KEYWORDS are checked to see if they are in the thesaurus of the part for which they are entered. The RESOURCE EDITOR will not allow more than the FORMAT-specified upper limit on the number of keywords in the keyword part.

LABELS are checked to see if they fit within the maximum length provided in the FORMAT. The RESOURCE EDITOR will not accept an overly long LABEL.

NUMBERS are checked to see if they fit within the range specified in the FORMAT, unless no range has been specified. All NUMBERS are checked to see if they are, in fact, numbers.

TYPES are checked to see if the TYPE NAME entered is a valid one, or an abbreviation of a valid one.

TEXT parts are limited to the number of lines specified in the FORMAT.

3.3.1.8 SPECIAL ANSWER-AHEAD FEATURE FOR NUMBERS

When entering the contents for consecutively formatted NUMBERS and TYPES, the user may answer the contents ahead without having to enter the NAMES of the consecutive NUMBERS by separating successive contents with semi-colons. For example, in the above case there are four NUMBER and/or TYPE parts: OCCUPATION, PEOPLE, POPULATION, and TYPE. Since they are specified in the FORMAT as consecutive PARTS, the above entry could have been done with the following line:

```
?occ=27;2;20000;suburban
```

This would have assigned PEOPLE the content of 2, POPULATION the content of 20000, and TYPE the content of SUBURBAN. UNKNOWN CONTENTS contents can be entered with consecutive semi-colons with nothing in between, when using this special answer-ahead feature. For example, if the following were entered:

```
?OCCUPATION=27;;;SUBURBAN
```

the contents of PEOPLE and POPULATION would be entered as UNKNOWN.

3.3.1.9 PROMPTED MODE RESOURCE ENTRY

In addition to the direct mode of resource entry shown above, the RESOURCE EDITOR can also operate in prompted mode. A monitor can specify in the format that the RESOURCE EDITOR is to go directly into prompted mode when the AR command is entered by a user, or that it is to go into direct mode shown above. In addition, the user may override this default mode at any time by using the +DIRECT and +PROMPTED mode commands. Once entered by a user the default mode for that user will remain as specified by the command used until he or she changes it later on with the complementary command. +PROMPTED or +DIRECT should be selected before entering the resource editor.

In prompted mode the EDITOR begins with an empty workspace (unless prompted mode was entered in the middle of entering or editing a resource) and then paces the user through each of the required parts in the order specified in the FORMAT. It automatically goes into the scratchpad for each of the TEXT items and into the special mode for KEYWORD entry as well. It prompts the user with the full PART NAME in the part-specific prompts as shown above, ending with a question mark (?). It prompts the user to enter a TYPE PART by first printing ENTER <PART NAME>:, then printing a line containing the TYPE NAMES enclosed in parentheses and then asking <PART NAME>?.

Below is an example showing entry of the same resource as in the example above, but in prompted mode:

```

COMMAND?ar
ID (letters#numbers)?p#303
DATE (mm/yy/dd)?5/18/80
NAME (up to 32 chars.)?Peter+Trudy Johnson-Lenz
ADDRESS (up to 32 chars.)?695 Fifth Street, Lake Oswego, Oregon
97034
PHONE (up to 20 chars.)?(503) 635-2615
OCCUPATION (1...100)?27
PEOPLE (1...20)?2
POPULATION (#)?20000
ENTER TYPE:
(URBAN/SUBURBAN/RURAL)
TYPE?suburban
ENTER KEYWORDS -- END WITH +
KEYWORDS?networking;computers;transformation
KEYWORDS?neighborhoods
KEYWORDS?+
ENTER DESCRIPTION -- LIMIT 20 LINES -- END WITH +
1?.text
2? P+T are independent consultants who work out of their home
3?office in Lake Oswego where they "teleconsult" and live the
4?life of "electronic cottage" workers predicted by futurist
5?Alvin Toffler.

```

```
6?+
OK TO ADD TO EXAMPLE (Y/N)?y
ADDED AS:
EXAMPLE 432 6/19/80 12:12 PM SOURCE: 118

ADD ANOTHER RESOURCE (Y/N)?n

COMMAND?
```

3.3.1.10 MODIFYING RESOURCES DURING PROMPTED MODE

The caret or "up-arrow" character (^) can be used to back-track through the parts of a resource when using the resource editor in PROMPTED mode. Users can go back to any parts that have been entered incorrectly without undoing any work that has already been done in the resource editor. Any parts' contents that have already been entered will remain as entered in the resource file, whether they were left blank or contents were assigned to them. Pressing the carriage return key will leave any part contents as they were and the user will be prompted for the next part. While doing this, the content of each part is NOT printed out for the user to view. The resource editor command :- is used for this.

If the user has already completed the PROMPTED MODE entry and is being asked the OK TO ADD question, but wishes to either list or modify the RESOURCE, the user should answer N. The user will then be asked DO YOU WISH TO MODIFY THIS RESOURCE (Y/N)?, and if N is the response here, then the resource will be discarded. If Y is entered, then the user will be taken back into the resource editor in DIRECT MODE with the resource loaded. Further editing can then resume with any of the resource editor commands.

When finished, + will terminate the editing and the OK TO ADD (Y/N)? question will be asked again. If N is entered here, the resource will be discarded.

3.3.1.11 DEFAULT ENTRIES IN PROMPTED MODE

If the user presses RETURN during PROMPTED MODE entry in response to a prompt for a DATE, NUMBER, or TYPE PART, the system will assign the UNKNOWN CONTENTS content to that PART. If the user presses RETURN in response to a prompt for a LABEL, the part will be blanked. If the user presses RETURN in response to an ID part, the system will simply repeat the prompt, since an ID PART cannot be left blank or unknown. It must be specified.

3.3.2 MICRO MODE FOR THE RESOURCE EDITOR (ER)

In addition to the DIRECT and PROMPTED modes in the RESOURCE EDITOR which are used by a person working directly with EIES, there is another mode, MICRO mode, which is used by a microcomputer that is transmitting files of resources to EIES. This mode is invoked through entry of the ER (Enter Resources from microcomputer) command, rather than through the AR command used to invoke the RESOURCE EDITOR normally. In response to the ER command, the system will ask PARITY CHECKING (Y/N)? If the user (or microcomputer) responds with N, no parity checking will be done. If the user (or microcomputer) responds with a Y, parity checking, as detailed below, will be used for the transmission.

3.3.2.1 MICRO MODE HANDSHAKING

The system then starts by prompting the micro with a l? prompt, plus a DC1 or control-Q character to trigger a response from the micro. EIES switch 20 must be turned on with the +SSW20 command (3.4.11) to enable this control-Q handshaking. The handshaking between the system and the micro continues in this way with the micro sending a line of information and then waiting for the system to ask for another by sending the DC1 with the next prompt.

3.3.2.2 PARITY CHECKING

If optional parity-checking mode is being used the micro computes and sends an additional parity-check character as the first character in each line which is then checked by the system to see if it is correct. This check character consists of the ASCII coded form of the one-digit hexadecimal number which represents the four least significant bits of the sum of the ASCII codes of all the other characters in the line and is hence the string representation of a hexadecimal digit from 0 through F. In parity-checking mode the system transmits an ***? (three asterisks and a question mark) prompt when a line is in error, thus signaling to the micro that the last line transmitted should be retransmitted.

3.3.2.3 CHECKING AND FORMATTING IN MICRO MODE

MICRO mode is quite unlike the full RESOURCE EDITOR. For example, it performs very few checks since it assumes that the microcomputer has already done checking of the entries before transmitting them to EIES. This is done to minimize connect time. Actual entry of the resources is done via a background task, much like that used by the UD command.

MICRO mode is similar to prompted mode in the full RESOURCE EDITOR in that MICRO mode expects the parts to be transmitted from the microcomputer in FORMAT order. Thus, the microcomputer transmits resources to EIES in PROMPTED format as defined in the previous section. However, MICRO mode differs from PROMPTED mode in that MICRO mode prompts only with a sequential line numbered prompt whenever a transmitted line is successfully received by MICRO mode. The more detailed prompts of PROMPTED mode are of little use, since all parts are transmitted from the microcomputer in FORMAT order anyway.

Each TEXT item appears in the proper location in the sequence as defined in the FORMAT, and all lines from that point on until a + is entered on a line by itself are assumed to belong to the TEXT item. The system prompts lines to be entered into a TEXT item with numbered prompts using numbers which are a continuous part of the line-sequencing in the whole resource. Thus, while it appears as if text were being entered into the SCRATCHPAD, the variety of editing features available in the EIES scratchpad are not available.

KEYWORDS are packed more than one per line, in any order; however the system assumes that all keywords for a resource appear in the transmission together as a single block of lines, ending with a +, at the proper location in the sequence of parts as defined by the FORMAT. KEYWORD parts can be represented by their names, abbreviations, or by their internal, numeric identification numbers (see Section (3.4.5) on the XK command for details) rather than as the keywords themselves to reduce the number of characters transmitted. The system prompts lines to be entered into the KEYWORD part with line numbered prompts continuing the numbering sequence used for all other parts.

All parts which are unknown are transmitted as null strings (zero characters in length). The parity-check byte for such lines is a hexadecimal 0 (zero).

3.3.2.4 END OF RESOURCE SIGNAL

When the microcomputer has successfully transmitted all the lines in a resource, it then transmits the final line of PROMPTED format which contains just a + followed by a comma and either the characters Y,Y or Y,N depending on whether this is the last resource of a series of resources being entered at once. A line with +,Y,Y signals the end of a resource, but not the end of the entire series of resources. A line with +,Y,N signals both the end of a resource AND the end of an entire series of resources. In all cases, both the Y and N can also be lower case letters (as in +,y,y or +,y,n). In response to a line with +,Y,Y MICRO mode transmits two lines to the MICRO computer, one which echoes back to the microcomputer the very first line in the resource, and the second which says simply, "ENTERED.". MICRO mode then continues by starting the sequence of lines at the beginning and prompting the micro for the first line of the next resource. (Note that MICRO mode does not actually ADD a resource to the database in that no SYSTEM ID or SYSTEM DATE is assigned or reported to the transmitting microcomputer. If the transmitting microcomputer desires such information, it must wait until the transmitted resources have been fully added to the database by the background task (generally the next day), and then use the DR command to list the display parts of those resources. If the FORMAT for the database includes an ID PART, that can be used as a convenient retrieval PART specification for the DR command.)

3.3.2.5 END OF TRANSMISSION SIGNAL

An entire transmission is terminated by a line with +,Y,N at the end of a resource. When the +,Y,N is received by the system, the two lines indicating successful receipt of the last resource are sent to the micro and then MICRO mode is terminated, and the system transmits a line saying "MICRO MODE TERMINATED.", and the user is returned to COMMAND?. Receipt of a line beginning with ++ or -- also terminates MICRO mode. A ++ causes return to EIES and a -- causes a normal EIES logoff. All characters following the ++ or -- are treated as regular EIES answer-aheads. Note, however, that use of ++ or -- to terminate MICRO mode means that the resource being entered will be lost unless ++ or -- is entered in response to the first part prompt.

3.3.2.6 EXAMPLE INTERACTION WITH MICRO MODE

Below is an example of MICRO mode. Note the hexadecimal parity-check bytes at the beginning of each line transmitted from the micro. Note also how the system responds to the transmission error in line 9.

```

COMMAND?er
PARITY CHECKING (Y/N)?y
1?AP#303
2?0Peter+Trudy Johnson-Lenz
3?4695 Fifth Street, Lake Oswego, Oregon 97034
4?6(503) 635-2615
5?F27
6?42
7?320000
8?CSUBURBAN
9?4NETWORKING;COMPUTERS;TRAN&/#.mATION;NEIGHBORHOODS
***?4NETWORKING;COMPUTERS;TRANSFORMATION;NEIGHBORHOODS
10?3+
11?5.text
12?9 P+T are independent consultants who work out of their home
13?Doffice in Lake Oswego where they "teleconsult" and live the
14?Flife of "electronic cottage" workers predicted by futurist
15?3Alvin Toffler.
16?2+
17?4+,Y,Y
P#303
ENTERED.
1?AP#329
.
.
.
23?F+,Y,N
P#329
ENTERED.
MICRO MODE TERMINATED.

COMMAND?

```

3.3.2.7 MICRO MODE CHECKING FOR UNIQUENESS OF ID PARTS

MICRO mode does not perform checking done by the full RESOURCE EDITOR such as checking the length of LABELS, the range of NUMBERS, or the validity of KEYWORDS. In fact, during transmission of resources from the microcomputer, MICRO mode does no checking at all, except to check parity (if the parity check option has been turned on), and to look for the special +,Y,Y end-of-resource signal and the +,Y,N end of MICRO mode signal. Since the microcomputer

assures that all resources are correctly formatted, no additional checking need be done within resources received in MICRO mode. However, the background task which actually ADDS such transmitted resources to the database does perform one check that is needed to assure the integrity of the database, and that is to check all ID parts for UNIQUENESS. Normally, the background task need not send a message to the EIES ID under which MICRO mode was used to enter resources, but when any duplicate IDs are found, they are reported in such a message and the resource which duplicates a previously entered ID part is rejected.

3.3.2.8 TRACE OF MICRO MODE ENTRIES

If the trace option is on, the system will make an entry for EACH resource entered with the ER command, showing the date, time, and PRIMARY ID of each resource entered. Since resources are not fully added to the database until after the background task submitted by the ER command has run, the background task rather than the ER command makes these entries. [NOTE: these entries will be entered into trace file of the user executing the ER command to whom the message returned by the task will be sent, rather than in the trace file identified by the EIES ID of the task, since the task ID is not a valid RESOURCES user ID.]

Even if the trace option is not on, the ER background task will increment the count of resources in the database and the count of resources entered by the user by one for each resource successfully entered by the task. The

3.3.3 STANDARD RESOURCE FORMATS FOR INPUT AND OUTPUT

The formats used by the :- and :: commands for listing resources with the RESOURCE EDITOR demonstrate the two primary standard output formats for resources that are used by the system. The example in (3.3.1.4) shows how each of these formats looks. The conventions or standards used by these two RESOURCE EDITOR commands are also used throughout the rest of the system to standardize the formats used for printing resources and for accepting them as input to the system. These standards make possible the exchange of resources from one database to another, between a microcomputer and EIES, and from EIES to a microcomputer, all using universally acceptable formats understood by all commands and aspects of the system.

The format used by the RESOURCE EDITOR :: command is called standard OUTPUT format and is used by nearly all commands when printing out resources.

The format used by the RESOURCE EDITOR :- command is called standard INPUT format and is used by both the AR and ER commands when entering resources into a database. INPUT format is also used by the XP and XR commands for transmitting resources to a microcomputer.

Both of these formats print out ALL PARTS in the same order as specified in the FORMAT, regardless of the order in which they were entered. They are described in greater detail in this section.

3.3.3.1 STANDARD RESOURCE OUTPUT FORMAT

The :: editor command yields what is known as standard resource OUTPUT format. This is also the same format used by all other +RESOURCES commands when printing resources. In this format all indirect edits are processed before printout and all TEXT parts are thus formatted to fit the parameters of the user's terminal. When used by other +RESOURCES commands, this format begins with the standard SYSTEM ID line as described below. This is followed by a blank line and then all PARTS in FORMAT order. Note that only one DATE, ID, LABEL, NUMBER, or TYPE part is printed per line. A blank line is added before the KEYWORDS part and before each TEXT item. KEYWORDS within the KEYWORD part are printed out in the same order in which they were entered. They are not listed alphabetically. PART NAMES are printed out followed by a colon (:) and two blanks. When more than one RESOURCE is printed at a time, two blank lines are included between each. The only difference between the :: format and that used by other +RESOURCES commands is that the SYSTEM ID line is not included in the :: output.

The +RCY copy command copies RESOURCES in standard resource OUTPUT format.

3.3.3.2 STANDARD SYSTEM ID LINE FORMAT

Each resource is given a standard SYSTEM ID line when it is added to a database. This ID line begins with the name of the DATABASE, followed by two blanks and the sequentially assigned SYSTEM ID. This is in turn followed by two more blanks and the SYSTEM DATE and TIME which is followed by two more blanks, the word SOURCE:, two more blanks and the SYSTEM SOURCE number. Below is an example of this format.

```
EXAMPLE 432 6/19/80 12:12 PM SOURCE: 118
```

If a resource has been modified at some point through use of the MR command, the SYSTEM ID line will contain a second line showing the date and time of such modification as shown below:

```
EXAMPLE 519    5/19/79  10:13 AM  SOURCE:  653
MODIFIED:    10/10/80  8:33. PM
```

3.3.3.3 STANDARD RESOURCE INPUT FORMAT

The :- editor command yields what is known as standard resource INPUT format. This format is very much like standard OUTPUT format in that parts are printed in FORMAT order. The differences are: (1) INPUT format never includes the SYSTEM ID line, (2) all indirect edits in TEXT items remain as entered, (3) all TEXT items and the KEYWORD part end with an extra line with only a +, (4) no extra blank lines are included, (5) the colons after PART NAMES are replaced by equal signs except for the NAMES of KEYWORD and TEXT PARTS where neither colons nor equal signs are included, (6) a special extra line in certain cases (only when INPUT format is output by RESOURCES commands as detailed in the section below on OUTPUT IN STANDARD INPUT FORMAT but not by the :- editor command). The exact contents of this special extra line depends on whether the resource is the last in a series or not. If it is the last in a series output by a given command, the line will contain +,Y,N (where the N answers the question ADD ANOTHER RESOURCE (Y/N)? in the negative). If it is not the last in the series, the line contains +,Y,Y. Thus, standard INPUT format is, as its name suggests, an exact copy of the format in which the RESOURCE was entered. A file containing output of resources in INPUT format can then be used as input to the RESOURCE EDITOR either singly or as a series of resources, through use of the +#[item] EIES command.

The +RCIN copy commands copies RESOURCES in standard resource INPUT format (but WITHOUT the extra line (6), above).

3.3.3.4 STANDARD RESOURCE PROMPTED FORMAT

In addition to standard INPUT and OUTPUT formats, there is a third standard RESOURCE format, called PROMPTED format. It is identical to standard INPUT format, except that the PART NAMES and equal signs are not included. This format is then, as its name suggests, an exact copy of the way a user would enter a resource into the system in PROMPTED mode (see below). This mode is used primarily for entry of resources via remote microcomputer. See the discussion of MICRO mode in (3.3.2).

3.3.3.5 OUTPUT IN STANDARD INPUT FORMAT

All RESOURCES commands normally print out resources in standard resource OUTPUT format. This includes printout from the GR, FP, FR, and SR commands. However, on occasion, a user may wish to have resources printed out in standard resource INPUT format, including the special final line (6) listed above. To support such a capability, two commands are provided that operate much like the FR and FP commands except that they print resources out in standard INPUT format. These commands are the XR (for Xmit Resources) command (3.5.11.4) that operates like the FR command (3.5.9); and the XP (for Xmit resources by Pattern) command (3.5.10.8) that operates like the FP command (3.5.10.5).

These commands support transmission of resources from EIES to a microcomputer for direct entry into the database of that microcomputer without operator intervention, if the microcomputer is programmed to accept such inputs. They also support movement of resources from one database to another.

3.3.3.6 STANDARD GET/FIND FEEDBACK MESSAGES AND OUTPUT

A variant of standard OUTPUT format is used by the GET and FIND commands. First of all, for every resource printed out in standard OUTPUT format by the GR, FP, or FR commands, the system adds a special line displaying the number of times the resource has been accessed and the number of comments and tally ratings that have been entered as feedback in response to the resource. In addition, at the end of any series of resources printed out by these commands, the system also prints out a message reminding users how to get this feedback as well as enter their own. An example of these additional outputs is shown below:

TIMES ACCESSED: 27; FEEDBACK TO DATE: 9 COMMENTS, 14 TALLY RATINGS

To get these COMMENTS use the GC command. To get the TALLIES use the GT command. Use the AC command to add your own COMMENTS.

Use the AT command to add your own ratings to the TALLIES.

SUMMARY OF USE OF STANDARD FORMATS

The following table summarizes the most important uses of the three standard RESOURCE formats:

OUTPUT FORMAT	INPUT FORMAT	PROMPTED FORMAT
------------------	-----------------	--------------------

	-----	-----	-----
AR entry (DIRECT mode)	no	yes	no
AR entry (PROMPTED mode)	no	no	yes
AR output	::	:-	no
ER entry (MICRO mode)	no	no	yes
FP output	yes	no	no
FR output	yes	no	no
GR output	yes	no	no
XP output	no	yes	no
XR output	no	yes	no

3.3.3.7 SUMMARY OF USE OF STANDARD PROMPTS

The following table summarizes the most important uses of the three standard RESOURCE formats:

	OUTPUT FORMAT	INPUT FORMAT	PROMPTED FORMAT
	-----	-----	-----
AR entry (DIRECT mode)	no	yes	no
AR entry (PROMPTED mode)	no	no	yes
AR output	::	:-	no
ER entry (MICRO mode)	no	no	yes
FP output	yes	no	no
FR output	yes	no	no
GR output	yes	no	no
XP output	no	yes	no
XR output	no	yes	no

3.4 THESAURUS MANAGEMENT COMMANDS

The RESOURCES system provides commands for ADDING, DISPLAYING, MODIFYING, REMOVING and TRANSMITTING (to a microcomputer) keywords in a part's thesaurus. These commands can only be used by someone with EDITOR (level 30) access or higher. They are explained below.

3.4.1 ADDING A KEYWORD TO THE THESAURUS (AK)

The AK (Add Keyword) command is used to add a keyword to the thesaurus for a part. When the AK command is entered by an editor the system will respond by asking ADD KEYWORD FOR WHICH PART? User support here includes a list of existing keywords in the thesaurus for that part when ? is entered. Next, the system will ask KEYWORD TO BE ADDED (Y/N)? The keyword is then entered by the editor

exactly as it is to appear in the thesaurus. Special characters such as +, -, / and others as well as numeric characters can appear at any point in a keyword, except that the keyword MUST BEGIN WITH A LETTER. KEYWORDS may even contain commas. The only character that CANNOT be embedded in a KEYWORD is a semi-colon since that is used to delimit KEYWORDS during entry and modification with the AR and MR commands.

After the keyword has been entered, the system will list it as a double-check and then ask OK TO ADD X (Y/N)? as in the example below:

```
COMMAND?ak
ADD KEYWORD FOR WHICH PART?key
KEYWORD TO BE ADDED (Y/N)?energy conservation
TO BE ADDED: ENERGY CONSERVATION
OK TO ADD ENERGY CONSERVATION (Y/N)?y
KEYWORD ENERGY CONSERVATION ADDED.
```

COMMAND?

Note that all lower case letters are automatically converted to upper case by the AK command. In general, all commands which ask for keywords always automatically convert to upper case to eliminate problems in retrieval later.

3.4.2 REMOVING KEYWORDS FROM THE THESAURUS (RK)

The RK (Remove Keyword) command is used to remove a keyword from the thesaurus for a database. When the RK command is entered by an editor the system responds by asking REMOVE KEYWORD FOR WHICH PART? The name of any valid keyword part in the database may be entered. User support here includes a list of valid keyword parts when a ? is entered. Then the system will ask for KEYWORD TO BE REMOVED (Y/N)? The keyword to be removed is then entered by the user after which the system lists it as a double-check and then asks OK TO REMOVE X (Y/N)? as shown in the example below:

```
COMMAND?RK
REMOVE KEYWORD FOR WHICH PART?key
KEYWORD TO BE REMOVED (Y/N)?energy conservation
TO BE REMOVED: ENERGY CONSERVATION
OK TO REMOVE ENERGY CONSERVATION (Y/N)?y
KEYWORD ENERGY CONSERVATION REMOVED.
```

COMMAND?

NOTE 1: the RK command not only removes the keyword from the thesaurus, but in order to retain the integrity of the database it must also (1) remove the keyword from all resources associated with the keyword, and (2) remove the keyword from the relational

information of other keywords to which it was related. If the keyword has any relations, you will be notified before the OK TO REMOVE question is asked.

NOTE 2: an editor must remove a keyword and then add it again to correct spelling, form, etc. There is no way to directly modify a keyword. The MK command, as discussed below, is used to modify the relationships among keywords, not to modify the spelling of a keyword.

3.4.3 MODIFYING THE RELATIONSHIPS AMONG KEYWORDS (MK)

The MK (Modify Keyword relationships) command is used to add and remove relational information to and from a keyword part's thesaurus. When the MK command is entered by an editor the system will respond by asking for the PART THESAURUS TO MODIFY? Any valid keyword part in the database may be entered. Support includes a list of valid keyword parts by entering a ? at this prompt. Then the system asks for the KEYWORD TO BE RELATED? and prompts for the RELATIONSHIP (B/-B/N/-N/R/-R/F/-F/I/-I)?

The letters B, N, R, F, and I stand for Broader than, Narrower than, Related to, use For, and use Instead, respectively. The minus sign indicates removal of a relationship. Absence of the minus sign indicates addition of a relationship. After the relationship has been entered by the user the system will feed back the keyword and the relationship asking for the second keyword as in SOLAR IS BROADER THAN (KEY)? if the keyword to be related was SOLAR and the relationship was B (broader than). At this point the user enters the keyword related to the first keyword, the system prints out the entire relationship as a double-check and then asks OK TO ADD RELATIONSHIP (Y/N)? as in the example below:

```
COMMAND?mk
PART THESAURUS TO MODIFY?key
KEYWORD TO BE RELATED (KEY)?solar
RELATIONSHIP (B/-B/N/-N/R/-R/F/-F/I/-I)?b
SOLAR IS BROADER THAN (KEY)?wind
SOLAR IS BROADER THAN WIND
OK TO ADD RELATIONSHIP (Y/N)?y
RELATIONSHIP ADDED TO SOLAR.
RELATIONSHIP ADDED TO WIND.
RELATIONSHIP (B/-B/N/-N/R/-R/F/-F/I/-I)?
```

```
COMMAND?mk
PART THESAURUS TO MODIFY?k
KEYWORD TO BE RELATED (KEY)?trolleys
RELATIONSHIP (B/-B/N/-N/R/-R/F/-F/I/-I)?i
TROLLEYS: USE INSTEAD (KEY)?street cars
TROLLEYS: USE INSTEAD STREET CARS
OK TO ADD RELATIONSHIP (Y/N)?y
```

RELATIONSHIP ADDED TO TROLLEYS
 RELATIONSHIP ADDED TO STREET CARS
 RELATIONSHIP (B/-B/N/-N/R/-R/F/-F/I/-I)?

COMMAND?

NOTE: when the relationship is added to the thesaurus, it is added to BOTH of the keywords being related, however not in the same way. In the example above, WIND would be listed as a NARROWER key than SOLAR under the SOLAR keyword entry, SOLAR would be listed as a BROADER key under WIND, TROLLEYS would be listed as USE INSTEAD STREET CARS, and STREET CARS would be listed as USE FOR TROLLEYS. Note also that after adding a relationship the system loops back to the RELATIONSHIP (B/-B/N/-N/R/-R/F/-F/I/-I)? question offering an opportunity to relate another key to the initial keyword. The system will continue to loop in this way until the user enters either a + or presses RETURN to stop the loop and return to COMMAND?.

Removing relationships is quite similar to adding them except that the minus sign must be used to specify removal.

In the same way that the N and B relationships form a complementary pair, the F and I relationships do. A keyword that is USED FOR (F) another is one that users should use in keying resources instead of the other keyword. A USE INSTEAD (I) keyword indicates that another term should be used instead and that it is NOT to be used as a keyword. The USE INSTEAD term is included in the thesaurus as a convenience to users so that if an attempt is made to use it during entry or retrieval, the system will suggest the preferred term instead. Once a keyword has been defined as USE INSTEAD, either by being defined as USE INSTEAD directly, or through reference by a USE FOR keyword (via entry of either the I or F in response to the RELATIONSHIP question), it cannot be used as a regular keyword and any such use will be prevented by the system, including any attempt to relate it in any other way to any other keyword. On the other hand, if a keyword is either related to others with B, N, R, or F, or has resources associated with it, it cannot be defined as a USE INSTEAD key until all those relationships and associated resources are first removed.

3.4.4 DISPLAYING KEYWORDS IN THE THESAURUS (DK)

The DK (Display Keywords) command is used to display either all or part of a part thesaurus in a highly compressed format. When the DK command is entered, the system will ask for the part thesaurus to be examined, then will ask DISPLAY ENTIRE LIST OF KEYWORDS (Y/N)? and if the answer is N it then asks for DISPLAY WHICH KEYWORDS (KEY PART/AAA-ZZZ)? Here the user can enter either a key part (first few characters of a series of keywords such as COMM for COMMUNICATIONS and COMMUNITY, etc.) or a range of keywords such as

COMMUNICA...COMMUNITY. The system then prints out the specified part of the thesaurus in alphabetical order and in compressed format as shown below.

```
COMMAND?dk
DISPLAY KEYWORDS FOR WHICH PART?key
DISPLAY ENTIRE LIST OF KEYWORDS (Y/N)?y
```

```
KEYWORD THESAURUS FOR RESOURCES IN THE EXAMPLE DATABASE:
12/12/80 12:23 PM
```

```
AARDVARK; ABLE BODIED PEOPLE; APPLES; BOXES; CRATES;
CRAZY GLUE; CROUTONS; CRUMBS; DOORSTOPS; ERASERS; FISH;
```

```
<BREAK>
```

```
COMMAND?
```

3.4.5 LISTING THE THESAURUS FOR MICRO MODE (XK)

MICRO mode in the RESOURCE EDITOR (via the ER command) allows KEYWORDS to be transmitted in the form of internal keyword ID numbers. In order for the micro to do this, it must know the ID numbers so that it can translate from the KEYWORDS as entered by the micro user into the ID numbers for sending to EIES.

The XK command (Xmit Keywords) is used to send a file of all keywords and their associated ID numbers to the micro. Before beginning the transmission, the XK command asks if ALL KEYWORDS are to be transmitted or, if not, the number of the first keyword to be transmitted. In either case, the keywords are transmitted in alphabetical order. Below is an example showing the format in which keys and numbers are output by the XK command.

```
COMMAND?xk
ALL KEYWORDS OR STARTING ID (Y/#)?y
435 KEYWORDS
ABANDONED BUILDINGS,138
ABANDONMENT,175,138
ABSENTEE LANDLORDS,209
ACTION PROGRAMS,315
ADOLESCENTS,13
ADULT EDUCATION,27
```

```
.
.
.
```

```
COMMAND?
```

NOTE 1: the very first line contains the number of keywords to follow. This is followed by keywords, one per line, including the

associated ID number, separated from the keyword by a comma. Note also that the keywords are in alphabetical order, regardless of the content of the ID numbers which may not necessarily correspond to alphabetical order since they may have been entered in a different, non-alphabetical order for one reason or another.

NOTE 2: keywords that are USE INSTEAD and cannot be used to describe resources include an extra ID number: the ID of the USE FOR preferred keyword so that the micro will be able to tell the micro user to use the preferred term instead. In the example above, ABANDONED BUILDINGS is to be used for ABANDONMENT, which is a USE INSTEAD term. The XK printout thus includes not only the ID number of ABANDONMENT, 175, but also the ID of the keyword to be used instead, 138.

3.4.6 GETTING THE THESAURUS (GK)

The GK (Get Keyword) command is used to display either all or part of a part thesaurus. When the GK command is entered the system will ask GET KEYWORDS FOR WHICH PART?, then will respond by asking GET ENTIRE LIST OF KEYWORDS (Y/N)? and if the answer is N it then asks for GET WHICH KEYWORDS (KEY PART/AAA-ZZZ)? Here the user can enter either a key part or a range of keywords such as COMMUNICA-COMMUNITY. Then the system prints two lines explaining alternative responses to the next question as shown in the example below and then asks COMPLETE DISPLAY (Y/N/K/R)? If the user enters a N the display will include the keywords only but not the lists of associated resources or related keywords. If the user enters an K the display will include the keywords and related keywords but NOT associated resources, and if the user enters an R the display will include the keywords and associated resources but NOT related keywords. If the user enters a Y or presses RETURN the display will include keywords, related keywords, and associated resources.

When resources are listed by the GK command, they are listed by their PRIMARY IDs using the ID PART specified by the monitor as the PRIMARY ID (see Sections 2.4 and 3.6 for more on the PRIMARY ID).

Below is an example showing how the GK command works:

```
COMMAND?gk
GET KEYWORDS BY WHICH PART?key
GET ENTIRE LIST OF KEYWORDS (Y/N)?n
GET WHICH KEYWORDS (KEY PART/AAA-ZZZ)?ener
Enter Y for keywords, related keys, and resources;
N for keywords only; K for keywords and related keys; or
R for keywords and resources.
COMPLETE DISPLAY (Y/N/K/R)?y
```

```
KEYWORD THESAURUS FOR RESOURCES IN THE ORGANIZATIONS DATABASE
6/18/80 12:23 AM
```

KEYWORD: RESOURCES AND RELATED KEYS

ENERGY: O#3, O#13, O#62, O#68
 Broader: NATURAL RESOURCES
 Related: BIOMASS, ENERGY AVAILABILITY
 ENERGY AVAILABILITY: O#69
 Related: ENERGY, RESOURCES AVAILABILITY
 WATER FOR ENERGY PRODUCTION
 Use for: ENERGY ACCESS
 ENERGY ACCESS:
 Use instead: ENERGY AVAILABILITY

COMMAND?

NOTE 1: the printout title RESOURCES AND RELATED KEYS, above, is tailored to the contents of the printout so that when only RESOURCES or only RELATED KEYS are printed, the title reflects that.

NOTE 2: if switch 13 is on, the system will go to the top of the next page before printing the GK display if the user answers Y, K, or R to the COMPLETE DISPLAY (Y/N/K/R)? question.

3.5 GET/FIND RESOURCE COMMANDS

Resources in a database may be accessed in several different ways. The most direct method is to use the GR (Get Resources) command (3.5.2). This command allows specification of a single PART NAME and associated CONTENTS and responds by printing out all those resources which are so specified in standard OUTPUT format. If the user wishes to display only the DISPLAY PARTS of the resource, the DR command can be used.

If more complex specifications are needed to get the desired resources, the FIND commands (FR--Find Resources, and FP--Find Resources by Patterns) must be used. The FR command is detailed in Section (3.5.9) below on finding resources, and the FP command is detailed in Section (3.5.10) on Pattern Commands. Both of these commands allow specification of Boolean combinations of PARTS. Section (3.5.9.1) provides additional details on the precise nature of such Boolean combinations.

If printout of resources in standard INPUT format is desired (usually for entry into another database or entry into a microcomputer database), the XR and XP commands can be used. These are equivalent to the FR and FP commands except that they print out resources in INPUT format.

In addition to these different ways of retrieving resources, the +RCY and +RCIN (Copy Resources) commands (3.11) have been provided to support copying of a resource into the scratchpad and a series of commands has been provided to allow users to easily add,

modify, and use patterns as explained in Section (3.5.10).

3.5.1 PRINTING RESOURCES WAITING (PR)

The PR (Print waiting Resources) command allows you to display all the new resources entered into the database since the last time you used the PR command. The resource marker may be moved with the Modify Markers command. This command also modifies the comment marker (3.7.9). The PR command operates much the same as a conference system, allowing users to print a series of new items and have the ability to move that marker where ever they please. If no items have been entered, the message NO ITEMS WAITING will be displayed.

3.5.2 GETTING RESOURCES BY A SINGLE PART (GR)

The GR (Get Resource) command can be used to get a resource on any INDEXED part, including DATES, IDS, KEYWORDS, LABELS, NUMBERS, and TYPES. Until a resource has been indexed it may only be retrieved by it's system parts. For example, getting resources by SYSTEM ID is specified as follows:

```
GR;system id=123
```

This will get the 123rd resource added to the database. Automatic indexing occurs periodically as established by the database monitor or it may be invoked immediately by using the UD command (Update Database). When GR is entered by a user, the system responds by asking GET BY WHICH PART? If the user then enters a valid PART specification, the system will print out the resource or resources specified in standard OUTPUT format. This is the same format that the RESOURCE EDITOR uses to list a resource with the :: command, plus an extra line at the beginning with the DATABASE name followed by the SYSTEM ID of the resource, the SYSTEM DATE and TIME, and the SYSTEM SOURCE (EIES ID under which the resource was entered), separated from the rest of the printout by a blank line. In addition to this printout of the resource itself, the GR command also prints out an additional line showing how many times the resource has been accessed (SYSTEM ACCESSES) and how many associated comments (SYSTEM COMMENTS) and tally ratings (SYSTEM TALLIES) have been entered in response to the resource. This is shown in the example below.

```
COMMAND?gr
GET BY WHICH PART?id=p#303
```

```
EXAMPLE 432 6/19/80 12:12 PM SOURCE: 118
```

ID: P#303
 DATE: 5/18/80
 NAME: Peter+Trudy Johnson-Lenz
 ADDRESS: 695 Fifth Street, Lake Oswego, Oregon 97034
 PHONE: (503) 635-2615
 OCCUPATION: 27
 PEOPLE: 2
 POPULATION: 20000
 TYPE: SUBURBAN

KEYWORDS:
 NETWORKING; COMPUTERS; TRANSFORMATION; NEIGHBORHOODS

DESCRIPTION:

P+T are independent consultants who work out of their home office in Lake Oswego where they "teleconsult" and live the life of "electronic cottage" workers predicted by futurist Alvin Toffler.

TIMES ACCESSED: 27; FEEDBACK TO DATE: 9 COMMENTS, 14 TALLY RATINGS

To get these COMMENTS use the GC command. To get the TALLIES use the GT command. Use the AC command to add your own COMMENTS.

Use the AT command to add your own ratings to the TALLIES.

GET BY WHICH PART?+

COMMAND?

NOTE: the system prints a message at the end of each successful GET that explains how users can get feedback on the gotten resources as well as enter their own feedback. This message is not printed out after each resource, but rather at the end of each series of resources, just before the system asks GET BY WHICH PART? again.

3.5.3 DISPLAYING RESOURCES (DR)

The DR (Display Resource) command is just like the GR command except that it displays resources in a brief format which includes only the first X parts (where X is specified by the monitor in the FORMAT and the first X parts are referred to as the DISPLAY PARTS of the RESOURCE -- see Section 3.6 for details). For example:

COMMAND?dr
 GET BY WHICH PART?id=p#303

EXAMPLE 432 6/19/80 12:12 PM SOURCE: 118

ID: P#303

DATE: 5/18/80
NAME: Peter+Trudy Johnson-Lenz
ADDRESS: 695 Fifth Street, Lake Oswego, Oregon 97034
PHONE: (503) 635-2615

GET BY WHICH PART?+

COMMAND?

3.5.4 UNKNOWN CONTENTS

UNKNOWN CONTENTS may be used to locate resources when using, for example, the GR or DR commands. The entry of the part name, an equals sign, a comma and a carriage return following the GET BY WHICH PART? question (in the form <partname>=,) will locate and print out all resources that have UNKNOWN CONTENTS for the specified PART. Entering a carriage return after the part-specific prompt will have the same result.

See the section on Resource Parts (2.4), and (2.4.8) and (2.4.9), for more discussion.

3.5.5 PART-SPECIFIC PROMPTS

PART-SPECIFIC PROMPTS are returned by the system when the user enters a PART NAME and no contents. If, when using the GR or DR command, the user enters only a part name to the prompt GET BY WHICH PART?, the system will respond with a prompt to guide the user in entering further information. These prompts are specific to the KIND OF PART only, and not to the part name.

See further discussion in (2.5.2) and (3.3.1.5), and examples of the prompts for each KIND OF PART.

3.5.6 LISTS AND RANGES

As with most other inputs in the +RESOURCES system, when a user is asked for CONTENTS of a PART, or supplies such CONTENTS directly by answering-ahead, the system will accept both lists and ranges of CONTENTS (2.5.1). In general, resources in such a list will be printed out in the order they appear in the list, or when an element of such a list defines more than a single resource, they are listed in inverse SYSTEM ID order within the element, and in input order between elements of the list.

3.5.7 USER SUPPORT FOR GETTING RESOURCES

If a user enters a ? in response to the GET BY WHICH PART? question, the system will respond with a compressed list of all indexed PART NAMES in the FORMAT, as shown below:

```
COMMAND?gr
GET BY WHICH PART??
```

```
Resources may be retrieved by the following part names:
SYSTEM ID; ID; DATE; ADDRESS; PHONE; OCCUPATION; PEOPLE;
KEYWORDS; POPULATION
Enter the name of the part you wish to use to get resources.
```

```
GET BY WHICH PART?
```

If a user enters a KEYWORD that is not in the thesaurus, the system will respond with NO SUCH KEYWORD. If the user enters a KEYWORD that is a USE INSTEAD keyword, the system will respond with USE X INSTEAD OF Y, where Y was entered by the user.

If a user enters part of a KEYWORD, and the part matches only one keyword in the thesaurus, that keyword will be used. If, however, the part matches more than one keyword, the system will respond by printing a list of all keywords that match the part and ask the user to choose one of them. At this input, the system will accept any keyword in the thesaurus, even one not in the list just printed out, but the keyword cannot be abbreviated unless the abbreviation uniquely identifies a single keyword.

If a user enters a range of keywords, the system will respond by printing a list of all keywords in that range and ask the user to choose one of them. At this input, the system will accept any keyword in the thesaurus, even one not in the list, but the keyword cannot be abbreviated unless the abbreviation uniquely identifies a single keyword.

Below are examples showing how the system responds to both parts and ranges of keywords:

```
COMMAND?gr
GET BY WHICH PART?keyword
KEYWORD (keyword,keyword...keyword)?ener
Resources are indexed under the following 2 keywords:
ENERGY; ENERGY AVAILABILITY
KEYWORD?energy
```

(the system prints out resources indexed under energy)

.
.
.

```

GET BY WHICH PART?keyword
KEYWORD (keyword,keyword...keyword)?communica...community
Resources are indexed under the following 4 keywords:
  COMMUNICABLE DISEASES; COMMUNICATION; COMMUNITY
  COMMUNITY OWNERSHIP
KEYWORD?community

```

(the system prints out resources indexed under community)

.
.
.

```

GET BY WHICH PART?+

```

3.5.8 TRACE OF GETS

If the option to trace user activities has been turned on, an entry will be made in the user's trace file for each resource gotten with the GR command showing the date and time of the action and the system ID of the gotten resource.

In either case, regardless of the status of the trace option, the count of resources gotten by the user and the count of times the resource itself has been gotten will be incremented each time a resource is accessed with the GR command.

3.5.9 FINDING RESOURCES (FR)

While the GR and DR commands can be used to print out resources specified by a single PART, often it is desirable to be able to search a database for resources which match combinations of PARTS. The FR (Find Resources) command is used for this purpose. It allows the user to specify an initial PART and CONTENTS, see how many resources match that specification, and then enter additional PARTS and CONTENTS to either reduce or add to the list of resources by ANDing, ORing, or NOTing the lists specified by each PART together as detailed in Section 2.5. Since any KIND of INDEXED PARTS may be specified, including DATES, IDS, KEYWORDS, LABELS, NUMBERS, TYPES, and SYSTEM parts, this command provides a highly flexible and powerful way of finding resources specified in almost any way.

When the FR command is entered by a user, the system responds by asking FIND BY WHICH PART? At this point the user can enter a complete PART specification of the form PART=CONTENTS, or s/he can enter just a PART NAME. If only a PART NAME is entered, the system will respond with the part-specific prompt just as with the GR command, above, and as detailed in Section 2.5.

As with the GR command, above, if a user specifies a KEYWORD part and enters either a non-unique abbreviation or a range of keywords, the system will respond by showing the user the keywords in that range or which match the abbreviation and then ask for a single, specific keyword from the user. This specific, single keyword cannot be abbreviated unless the abbreviation uniquely identifies a single keyword.

Once the user has entered a complete PART and CONTENTS specification, the system continues by telling the user how many resources are associated with that part.

After listing the resources indexed by the user-specified part, the system asks the user for ACTION (G,I,D,N,&,/,,-)? If the user enters Y (or just presses RETURN) the system will list each resource in the list in its entirety, in INVERSE SYSTEM ID order from most recent to least recent. If the user enters I, only the PRIMARY ID for each resource in the list will be printed out, in INVERSE SYSTEM ID order. If the user enters D, only the DISPLAY PARTS will be printed out for resources in the list, again in INVERSE SYSTEM ID order. If N is entered the user will be returned to COMMAND? Entry of a + has the same effect as entry of an N.

The system will continue to ask ACTION (G,I,D,N,&,/,,-)? after each input from the user, whether it be a printout request or another PART specification, so that the user may make intermediate listings and/or displays of the found resources and then continue on to make more complete listings or make further modifications of the list.

3.5.9.1 BOOLEAN COMBINATIONS OF PARTS

If an &, /, or - is entered at this point, the system will begin a Boolean search in which the list of resources already found will be combined with the list specified by the next additional PART specification. The lists will be ANDed together (including only those resources in both lists) if the character & is entered. The lists will be ORed together (including any resources in either list) if the character / is entered. The second list will be subtracted from the first list (NOTed) if the character - is entered. As with the initial PART specification, any DATE, ID, KEYWORD, LABEL, NUMBER, or TYPE part can be specified in combination with any other KIND of part.

After entry of the Boolean operator and another PART and CONTENTS, whether or not it is a keyword, the system combines the lists of resources associated with each keyword and then prints out the number of resources in the combined list. The process can be iterated until a list of satisfactory size is achieved, or until the user gives up on the search request.

3.5.9.2 USER SUPPORT FOR MULTIPLE KEYWORD FINDS

If the user specifies a KEYWORD part in response to the FIND BY WHICH PART question, and if the previously specified PART was also a KEYWORD part, it will print out the RELATIONAL information for the keyword just used showing the keywords to which it is related as a guide in selection of the next keyword. Then the system asks KEYWORD (keyword,keyword...keyword)? If the user enters a unique abbreviation or keyword, the list of resources indexed by that keyword is combined with the current list as specified by the Boolean operator, else the system lists the keywords in the range specified by the user and then asks KEYWORD? as before. If, however, the user answers ahead by also supplying a keyword with the &, /, or - the relational information will be suppressed and the system will NOT ask for a KEYWORD?

Below is an example interaction with the FR being used to find resources by a Boolean combination of two keywords with various types of informative support from the system:

```
COMMAND?fr
FIND BY WHICH PART?keyword
KEYWORD (keyword,keyword...keyword)?ener
ENERGY:
ENERGY AVAILABILITY:
4 resources in list
```

```
Enter G to get resources; I for IDs only; D to display; N to
return
to COMMAND?; & to AND, / to OR, and - to NOT with another part.
ACTION (G,I,D,N,&,/, -)?&
AND WITH WHICH PART?keywords
```

```
Keywords related to ENERGY:
  Broader: NATURAL RESOURCES
  Related: BIOMASS, ENERGY AVAILABILITY
KEYWORD (keyword,keyword...keyword)?biomass
```

```
KEYWORD=ENERGY;&KEYWORD=BIOMASS
2 resources in list
```

```
Enter G to get resources; I for IDs only; D to display; N to
return
to COMMAND?; & to AND, / to OR, and - to NOT with another part.
ACTION (G,I,D,N,&,/, -)?n
```

```
COMMAND?
```

NOTE 1: additional keywords used in refining a search through Boolean combinations with previously entered keywords are not necessarily limited to only related keywords. While in the example above the keyword BIOMASS was listed as a related keyword, the user could have entered ANY keyword in the thesaurus at this point.

NOTE 2: if the user enters a keyword that is defined as a USE INSTEAD keyword (2.3) that cannot be associated with a resource, the system will respond with USE Y INSTEAD OF X where X is the keyword entered by the user and Y is the USE FOR keyword association with X.

NOTE 3: if the user enters an &, /, or - to refine the list of resources, the system prompts the user with AND WITH WHICH PART?, OR WITH WHICH PART?, or NOT WITH WHICH PART? instead of the FIND BY WHICH PART? question asked the first time.

3.5.9.3 FINDING RESOURCES WITH NON-KEYWORD PARTS

While the above example shows finding resources by combinations of keywords, the system also allows finding resources by Boolean combinations of other KINDS of PARTS as well, either in combination with each other or in combination with keywords. This makes it possible to find resources specified by virtually any combination of PARTS and CONTENTS.

Below is an example showing finding of resources by combinations of KINDS of PARTS other than keywords:

```
COMMAND?fr
FIND BY WHICH PART?date=12/12/80
```

```
DATE=12/12/80
10 resources in list
```

Enter G to get resources; I for IDs only; D to display; N to return

to COMMAND?; & to AND, / to OR, and - to NOT with another part.

```
ACTION (G,I,D,N,&,/,,-)?&
AND WITH WHICH PART?id=p#4000...p#5000
```

```
DATE=12/12/80;&ID=P#4000...P#5000
4 resources in list
```

Enter G to get resources; I for IDs only; D to display; N to return

to COMMAND?; & to AND, / to OR, and - to NOT with another part.

```
ACTION (G,I,D,N,&,/,,-)?-zipcode=98000...;d
```

```
DATE=12/12/80;&ID=P#4000...P#5000;-ZIPCODE=...98000
1 resource in list
```

```
EXAMPLE 399 12/12/80 9:10 AM SOURCE: 653
```

```
ID: P#4023
NAME: Steve Johnson
ADDRESS: PCRC, 1723 NE 10th, Portland, Oregon
```

PHONE: (503) 284-9465

COMMAND?

NOTE: in the above example, the user answered ahead after specifying the NOT (-) Boolean operator by also specifying the PART to be "notted" (zipcode=98000...) and the action to be taken at that point (d for Display). This caused suppression of the normal user-support prompt explanation.

Just as the GR command allows getting of all resources with UNKNOWN CONTENTS, the FR command allows the PART= specification to be used to find resources with UNKNOWN CONTENTS.

Just as the GR command provides users with PART-specific prompts if they enter only a PART NAME with no CONTENTS specification, the FR command does likewise, following the prompt conventions detailed in Section (2.5.3).

Just as the GR command allows for specification of lists and ranges of resources within a PART specification, so does the FR command. The only difference is that resources will be listed by the GR command in the order of the elements in the input list, while resources listed by the FR command will be in inverse SYSTEM ID order.

3.5.9.5 USER SUPPORT FOR FINDING RESOURCES

Just as the GR command provides support for users in terms of ? feedback of PART NAMES, feedback of USE INSTEAD keywords, and feedback of keywords which match an abbreviation and/or lie within a range of keywords, the FR command provides the same user support in addition to the special types of support it uniquely provides. ? at ACTION displays a short explanation of each action the user can take at the input. ? at FIND BY WHICH PART? will display all of the valid parts in the database that can be used at that input.

3.5.9.6 ACCESS TO COMMENTS AND TALLY FEEDBACK

When the FR command prints out resources, it includes a line at the end of EACH resource showing the number of times the resource has been accessed, and the number of comments and tally ratings that have been entered as feedback for that resource. Then, at the end of the entire list of resources being printed out, it prints out a message reminding users how they can get the comments and tally

results as well as enter their own. The format of these printouts is exactly like that of the GR command (3.5.2).

3.5.9.7 TRACE OF FINDS

If the option to trace user activities has been turned on, an entry will be made in the user's trace file for each resource actually printed out with the FR command showing the date and time of the retrieval and the system ID of the found resource.

In either case, regardless of the status of the trace option, the count of resources gotten/found by the user and the count of times the resource itself has been gotten will be incremented each time a resource is actually printed out with the FR command.

3.5.9.8 COMPARISON OF GR AND FR FEATURES

The GR and FR commands provide similar and yet somewhat different ways of retrieving resources from a database. These similarities and differences are summarized in the table below:

	GR	FR
	--	--
use PART=CONTENTS specifications	yes	yes
access by any KIND of part	yes	yes
provide ? feedback of PART NAMES	yes	yes
allow access by UNKNOWN CONTENTS	yes	yes
prompt for PART-specific CONTENTS	yes	yes
allow lists and ranges	yes	yes
feed back keywords matching an abbreviation or in range	yes	yes
feed back of related keywords	no	yes
print out number of COMMENTS and number of TALLY ratings	yes	yes
print in inverse SYSTEM ID order	yes	yes
print in input list order	yes	no
Boolean combinations of parts	no	yes
loops for more parts after printing out resources	yes	yes
option to display resources	no	yes

3.5.10 PATTERN COMMANDS

When a user simply answers ahead with the FR command, the effect is to enter a PATTERN (as described in Section 2.5) into the system. Note that the syntax of answering ahead with these commands produces EXACTLY the same character string as the definition of a PATTERN. However, the FR command does not remember these defacto patterns, since they are assumed to be of temporary value when this command is being used. For finds that use the same pattern over and over again, the commands below should be used.

The only difference between finding by PATTERN and finding as in the previous section are: (1) the machine can remember a user's favorite PATTERNS by name for convenient access later on, (2) the system scans only the MOST RECENT entries through use of a PATTERN marker, and (3) pattern searching may be performed automatically by the system on a regular basis.

A PATTERN MARKER is a marker in the sequential file of resources ordered by the SYSTEM ID. It is initially set to zero (no resources processed with the pattern) when a pattern is initially defined by a user. Then, each time the user uses the FP command (see below) to find resources, the pattern scan begins with the first resource with a higher SYSTEM ID number than the marker and then when the scan is complete the marker is advanced to the most recently added resource.

NOTE 1: the marker is not advanced resource by resource, but rather is advanced only one at the end of the scan just after the list of resources is printed out and before the user is asked ACTION (G,I,D,N)?.

NOTE 2: while the scan begins with the first resource with a SYSTEM ID number higher than the PATTERN MARKER and proceeds to process all resources from there on up to the most recent, the printout of the list of resources is nevertheless in INVERSE order so that the most recently added resources are listed first. This then provides a way a user can keep up with entries in specific areas of the database, as specified by a pattern. The output from a PATTERN find is exactly like that of the FR command, except that the lists of resources produced include only resources from the most recent back to the PATTERN MARKER for that particular user and pattern.

3.5.10.1 ADDING PATTERNS (AP)

The AP (Add Pattern) command may be used to add a pattern to a user's pattern file. When the AP command is entered, the system will respond by asking for the PATTERN NAME?. Then it will ask for the PART specification, as described in Section (2.5.2), and continue to ask for BOOLEAN OPERATORS and PARTS until a + is entered (or RETURN is pressed) to end the definition. An example is shown below:

```
COMMAND?ap
PATTERN NAME?the usual
PART?keyword=energy
BOOLEAN OPERATOR AND PART?&keyword=solar energy
BOOLEAN OPERATOR AND PART?/keyword=wind
BOOLEAN OPERATOR AND PART?+
PATTERN TO BE ADDED:
THE USUAL: KEYWORD=ENERGY;&KEYWORD=SOLAR ENERGY;/KEYWORD=WIND
OK TO ADD PATTERN (Y/N)?y
THE USUAL PATTERN ADDED.
```

COMMAND?

3.5.10.2 MODIFYING PATTERNS (MP)

The MP (Modify Pattern MARKER) command may be used to MOVE the PATTERN MARKER for a pattern either forward or backward in the sequence of resources in the database scanned by the pattern. When the MP command is entered, the system will respond by asking for the PATTERN NAME? and then will display the current PATTERN MARKER (last resource scanned) and the current highest SYSTEM ID number for that database and then will ask for NEW LAST RESOURCE (#)?. It then prints the new marker and as a double-check asks OK TO MODIFY X PATTERN (Y/N)? as shown in the example below:

```
COMMAND?mp
PATTERN NAME?the usual
Last resource scanned: 432
Highest SYSTEM ID for resources in EXAMPLE database: 979
NEW LAST RESOURCE (#)?858
Last resource scanned now: 858
OK TO MODIFY THE USUAL PATTERN (Y/N)?y
THE USUAL PATTERN MODIFIED.
```

COMMAND?

NOTE: The MP command only allows modification of a pattern marker. It does not allow direct modification of the pattern itself. If a

user wishes to modify a pattern, s/he must first REMOVE it and then ADD it again.

3.5.10.3 REMOVING PATTERNS (RP)

The RP (Remove Pattern) command may be used to remove a pattern from a user's pattern file. When the RP command is entered, the system will respond by asking for the PATTERN NAME? and then will ask as a double-check if it is OK TO REMOVE X PATTERN (Y/N)? The following example demonstrates this process:

```
COMMAND?rp
PATTERN TO REMOVE?the usual
OK TO REMOVE THE USUAL PATTERN (Y/N)?
```

THE USUAL pattern removed.

```
COMMAND?
```

3.5.10.4 DISPLAYING PATTERNS (DP)

The AP (Add Pattern) command may be used to add a pattern to a user's pattern file, or just a single pattern, depending on the user's request. When the DP command is entered, the system will respond by asking PATTERN NAME? If the user enters the name of an existing pattern the system will display that pattern. If the user just presses RETURN, the system will display all patterns. In either case, the PATTERN MARKER (last resource scanned) for each pattern displayed is also shown as in the example below:

```
COMMAND?dp
PATTERN NAME?the usual
THE USUAL: KEYWORD=ENERGY;&KEYWORD=SOLAR ENERGY;/KEYWORD=WIND
Last resource scanned: 858
```

```
COMMAND?
```

3.5.10.5 FINDING BY PATTERNS (FP)

The FP (Find resources by PATTERN) command will perform a pattern find and display the number of resources in the final list. Since finding by patterns may take time, the system first asks if the user wishes to have the processing of the find request done overnight via a background task with the results being returned in a message the following day. Then it will ask the user for ACTION

(G,I,D,N)?, similar to the FR command. Unlike the FR command, the FP command will return to COMMAND? after the user answers the ACTION (G,I,D,N)? question, since the FP command is not designed to allow interactive modification of the list of resources found. This is shown in the example below:

COMMAND?fp

Your request can be processed overnight while you are not on line and

the results sent to you as an EIES message, if you prefer.

PROCESS OVERNIGHT (Y/N)?n

PATTERN NAME?the usual

PATTERN USED FOR FIND:

THE USUAL: KEYWORD=ENERGY;&KEYWORD=SOLAR ENERGY;/KEYWORD=WIND

...beginning scan at resource 415

Since your last search using this pattern,

3 resources matching the pattern have been entered.

Enter G to get resources; I for IDs only; D to display;

N to return to COMMAND?

ACTION (G,I,D,N)?n

COMMAND?

NOTE: If the user responds with a Y to the PROCESS OVERNIGHT (Y/N)? question, the ACTION (G,I,D,N)? question must be asked in advance of invoking the task so that the task can know what to do once the list of matches has been found.

3.5.10.6 ACCESS TO COMMENT AND TALLY FEEDBACK

When the FP command prints out resources, it includes a line at the end of each resource showing the number of times the resource has been accessed, and the number of comments and tally ratings that have been entered as feedback for that resource. Then, at the end of the entire list of resources being printed out, it prints out a message reminding users how they can get the comments and tally results as well as enter their own. The format of these printouts is exactly like that of the GR command (3.5.2).

3.5.10.7 TRACE OF PATTERN FINDS

If the option to trace user activities has been turned on, an entry will be made in the user's trace file for each resource printed out by the FP command showing the date and time of the retrieval and the PRIMARY ID of the gotten resource. [NOTE: if processing is done overnight by a task, these trace file entries

will be entered into the trace file of the user executing the FP command to whom the message returned by the task will be sent, rather than in the trace file identified by the EIES ID of the task, since the task ID is not a valid RESOURCES user ID.]

In either case, regardless of the status of the trace option, the count of resources gotten by the user and the count of times the resource itself has been retrieved will be incremented each time a resource is printed out with the FP command.

3.5.10.8 PRINTING OUT FOUND RESOURCES IN INPUT FORMAT (XP)

If the user wishes to print out resources found with the FP command in standard input format, usually for transmission to a microcomputer or for entry into another RESOURCES database, the XP (Xmit Pattern) command should be used instead. It is functionally equivalent to the FP command except that it prints out resources in standard INPUT format. See Section 3.1.1 for the details of this standard resource format.

The XP command increments the counts of resources retrieved by a user and makes entries in the user's activity trace file in the same way that the FP command does.

The XP command prints out resources in standard INPUT format and therefore signals the end of each resource with a line containing +,Y,Y. It signals the end of the last resource in the list or series of resources by a line containing +,Y,N.

3.5.10.9 PRINTING OUT PARTS OF RESOURCES

If the user wishes to print out only some PARTS of resources found with the FP command, either the I option to print out the first PART only (usually an ID PART), the D option to print out the standard DISPLAY PARTS (specified in the FORMAT by the monitor) or the MTE command to Modify TEmplate may be used to specify any desired PARTS in any desired order before using the FP command. See Section (3.5.11.1) for more on the MTE command.

3.5.11 SPECIAL RESOURCE OUTPUT FEATURES

The system normally prints out all PARTS of resources retrieved by the GR and FR commands. This is because the default TEMPLATE, which specifies printing of all PARTS, is in effect. If the user wishes to print out only a subset of the PARTS, the DR command may be used to print out the standard FORMAT-defined DISPLAY PARTS, or the FR command may be used with the I-specification to list just the initial ID PART. A more general and flexible approach is to use the TEMPLATE to specify exactly which PARTS and in what order they are to be printed out.

Each user has his or her own TEMPLATE which is normally set to the default TEMPLATE, which is the completely empty or blank TEMPLATE and signifies printing of all PARTS. There are two commands (GTE and MTE) that may be used to get and modify the TEMPLATE.

3.5.11.1 MODIFYING THE TEMPLATE (MTE)

The MTE command can be used to modify the current template setting. Any valid parts in the database format, including abbreviations, are accepted. If a template is to be turned off, or cleared, enter a carriage return as shown in the example below. As many parts as exist in the format may be specified for the template, but no part can be used twice.

```
COMMAND?gte
CURRENT TEMPLATE SPECIFIES ALL PARTS.
```

```
COMMAND?mte
ENTER PARTS IN PRINT OUT ORDER -- END WITH +
PART?name
PART?address
PART?description
PART?+
OK TO MODIFY TEMPLATE (Y/N)?y
TEMPLATE MODIFIED: 3 PARTS CURRENTLY SPECIFIED.
```

```
COMMAND?mte
ENTER PARTS IN PRINT OUT ORDER -- END WITH +
PART?+
OK TO MODIFY TEMPLATE (Y/N)?y
TEMPLATE MODIFIED: ALL PARTS CURRENTLY SPECIFIED.
```

```
COMMAND?
```

3.5.11.2 GETTING THE TEMPLATE (GTE)

The GTE command (get template) allows the user to look at the current template setting. If no template is set, the system responds with CURRENT TEMPLATE SPECIFIES ALL PARTS. Otherwise the part values for all the parts in the template are printed, as in the examples below:

```
COMMAND?gte
CURRENT TEMPLATE SPECIFIES ALL PARTS.
```

Or, if the template is already set.

```
COMMAND?gte
CURRENT TEMPLATE SPECIFIES THE FOLLOWING 3 PARTS.
NAME; ADDRESS; DESCRIPTION
```

```
COMMAND?
```

3.5.11.3 THE TEMPLATE IN EFFECT NOTICE

When a non-default TEMPLATE is in effect, some PARTS of resources may not be printed out. Therefore, to make sure the user is aware of this, a special message is issued by the system whenever a user invokes a command that uses the TEMPLATE and the TEMPLATE does not specify printout of all PARTS. This message says "NOTICE: YOUR TEMPLATE CURRENTLY SPECIFIES PARTIAL RESOURCE PRINTOUT."

3.5.11.4 PRINTING RESOURCES IN STANDARD INPUT FORMAT (XR)

Both the GR and FR commands print out resources in standard OUTPUT format. However, the XR (Xmit Resources) command may be used to print out resources in standard INPUT format. The XR command operates in the same way as the FR command, allowing Boolean combinations of PARTS and interactive development of the list of resources to be printed out, except that it prints out resources in standard INPUT format as defined in Section (3.3.3.3). The XR command is usually used to transmit resources to a microcomputer for direct entry into the database on that machine.

The XR command increments the counts of resources retrieved by a user and makes entries in the user's activity trace file in the same way that the FR command does.

The XR command prints out resources in standard INPUT format and therefore signals the end of each resource with a line containing +,Y,Y. It signals the end of the last resource in the list or series of resources by a line containing +,Y,N.

3.5.11.5 ADDING LAYOUTS (AL)

The AL (add layout) command allows the user to add a layout to the system. If the layout is declared as a global one, other users can use it for resource printing. Otherwise it is a local layout that only the person under whose id the layout was added can use it. The AL command then asks for the name that the user wishes to store the layout definition under. The name must start with a letter and can contain any number of characters. After the layout is set-up, the user is taken into the scratchpad to compose the text. All EIES indirect edit commands are operational inside of this text. Following is an example of creating a layout:

```
COMMAND?al
GLOBAL OR LOCAL LAYOUT (G/L/CR=L)?l
LAYOUT NAME?test6?
SCRATCHPAD CONTAINS 7 LINES.
OK TO OVERWRITE (Y/N)?y
```

Entering scratchpad:

```
1?.page
2?.cent @[id]
3?.blank2
4?      @[name]
5?      @[address]
6?      @[city]
7?      @[state] @[zip code]
8?.blank1
9? This abstract is of type @[type]
10?@[abstract]
11?.page
12?+
```

```
OK TO ADD TEST6? (Y/N)?y
TEST6? LAYOUT ADDED.
USE UL TO UPDATE LAYOUT FOR RESOURCE PRINTING.
```

COMMAND?

The @[...] descriptor is used to tell the system that what is between the brackets is a part in the database and the part of the resource is put in when the resources are printed out. The UL command is used to tell the system to use the layout instead of a Template. Part abbreviations are valid inside the brackets. If the

system cannot evaluate the contents between the brackets into a valid part name it just leaves it alone.

3.5.11.6 MODIFYING LAYOUTS (ML)

The ML (modify layout) command allows the user to modify the contents of an existing layout. Only the person who added the layout is granted privileges to modify the text. The name and global definition cannot be modified, only the layout contents. The system only asks the user for the name of an existing layout then takes the user into the scratchpad with the text of the layout already loaded.

```
COMMAND?ml
LAYOUT NAME?test6?
```

Entering scratchpad:

```
18?
18?/.cent//1
1: @[id]
18?+
```

```
TEST6? LAYOUT MODIFIED.
USE THE UL COMMAND TO UPDATE LAYOUT FOR RESOURCE PRINTING.
```

3.5.11.7 REMOVING LAYOUTS (RL)

The ability to remove layouts is achieved with the RL command. The system only asks for the name under which the layout is stored under to look it up. A double check question is asked before the layout is actually removed.

```
COMMAND?rl
LAYOUT TO REMOVE?test6?
OK TO REMOVE TEST6? (Y/N)?y
```

```
TEST6? LAYOUT REMOVED.
```

```
COMMAND?
```

3.5.11.8 GETTING LAYOUTS ON FILE (GL)

The get layout command allows the user to view the layout definitions that are defined either locally or globally. If RETURN is hit at the LAYOUT NAME prompt, then all the layouts the user has defined under their id will be displayed. If the user number of another user in the database is prefixed to a layout name then the global layout definition will be displayed (if it exists). The user can also enter the name of a layout defined under the id without the user number prefix to default to that id. Here is a couple of examples:

```
COMMAND?gl
LAYOUT NAME?118:test
.
.
.
  layout is displayed
.
.
.
```

```
LAYOUT NAME?test6?
1: @[id]
2: ...
  layout is displayed
```

```
LAYOUT NAME?+
```

```
COMMAND?
```

3.5.11.9 UPDATING LAYOUTS FOR RESOURCE PRINTING (UL)

The update layout command is used when the user wants to use a specified layout for resource printing. If RETURN is hit here the default layout is assumed, which is no layout at all, and control is returned to the previous definition of the Template. If the user prefixes the layout name with another user number, a global layout definition is assumed and the system goes to that user's layout definitions looking for a global layout with the specified name. The person can also enter just a name defined under his or her own id to use a previously defined layout. The UL command over writes any previously updated layout.

3.6 REPORT-WRITING COMMANDS

The report-writing commands are divided into two general categories: the SORT and TABULATE commands. The SR (Sort Resources) command prints lists of resources sorted on a specified PART. It does no computation beyond sorting. The tabulate commands (TD, TR, and TU) accumulate various statistics which are then formatted into reports which are then printed out. Since these commands often take some time to process, the user is given an option to run them as background tasks and have the results returned to them in the form of an EIES message. All of these commands are described below.

3.6.1 GETTING SORTED LISTS OF RESOURCES (SR)

When the SR command is entered, the system will first ask if the user wishes to have the processing done overnight by a background task, as shown below. Then it will ask PART ON WHICH TO SORT? Here the user may enter the PART NAME of any PART in the FORMAT or any SYSTEM PART, even if not indexed, since the SR command will index any unindexed PARTS if it needs to. The specified PART will be used to sort either all or a specified portion of the resources in the database before printing them out. The sort order will be determined by the KIND of PART. Note that if a NAME part is to be used for sorting, the names will need to be entered last name first since the sort is only a simple alphabetic sort.

The system will continue by asking SORT ALL RESOURCES IN DATABASE (Y/N)? If the user responds with a Y (for Yes) or presses RETURN, the SR command will include all resources in the sorted output. If the user responds with a N, the system will ask SELECT RESOURCES BY WHICH PART? Here the user can enter any indexed PART specification to specify a subset of the entire database of resources.

The SR command uses the output TEMPLATE (3.5.11) to specify which PARTS of resources are to be included in the printout. If only some PARTS are to be included, the MTE (Modify TEmplate) command (3.5.11.1) is used to specify those parts, and their order in the printout, before using the SR command.

Regardless of the KIND of PART used to sort the resources, the sort assumes that upper and lower case letters are equivalent.

For each resource printed out with the SR command, the system increments the count of resources retrieved by the user, and if the trace option is on, makes an entry into the users trace file for each resource printed out. [NOTE: if processing is done overnight by a task, these trace file entries will be entered into the trace

file of the user executing the SR command to whom the message returned by the task will be sent, rather than in the trace file identified by the EIES ID of the task, since the task ID is not a valid RESOURCES user ID.]

Below is an example showing how the SR command works.

COMMAND?sr

Your request can be processed overnight while you are not on line and

the results sent to you as an EIES message, if you prefer.

PROCESS OVERNIGHT (Y/N)?y

ENTER PARTS ON WHICH TO SORT -- END WITH +

PART ON WHICH TO SORT?name

PART ON WHICH TO SORT?+

SORT ALL RESOURCES IN DATABASE (Y/N)?n

SELECT RESOURCES BY WHICH PART?id=p#303...

GET (1) OR DISPLAY (2) RESOURCES (1-2)?1

Your request is being processed. The results will be sent to you as an EIES message.

COMMAND?

The output from the SR command, whether delivered to the user in an EIES message or printed out directly uses the format shown below:

EXAMPLE RESOURCES AS OF 12/12/80 12:12 PM

SORTED BY NAME

INCLUDING ALL RESOURCES WITH ID=P#303...

(resources in standard OUTPUT format)

NOTE 1: if switch 14 is on, the system goes to the top of the next page before printing the output from the SR command, if it is being printed out directly.

NOTE 2: if a template is specified for the user, it will be used when printing the sorted list of resources.

3.6.2 TABULATING RESOURCES BY PARTS (TR)

The TR (Tabulate Resources) command will compute and print a report showing how many resources are indexed with the specified KEYWORD, LABEL, NUMBER, or TYPE PART. The format of the tabulation depends on the KIND of PART specified. The first question asked by the TR command is PROCESS OVERNIGHT (Y/N)?, because the tabulations can be time-consuming. If the user enters a Y, or presses RETURN, the tabulations will be done by a background task overnight and the results returned to the user in an EIES message.

Following this initial question, the TR command continues by asking for the PART FOR TABULATION? Here the user enters a PART name. If a keyword PART name is entered, the system will then ask for a specific keyword to be tabulated, or a range of keywords to be tabulated, or as shown below, the user may answer this ahead by providing the list of keywords with the part name. If a NUMBER or TYPE part name is entered, the system does not ask for a CONTENTS specification because all CONTENTS for the specified NUMBER or TYPE part are always tabulated. The system then continues to ask PART FOR TABULATION? until the user presses RETURN or enters a + to terminate the specification of parts.

Once the PARTS FOR TABULATION have been specified, the system asks TABULATE ALL RESOURCES (Y/N)?. If the user enters a Y (for Yes) or presses RETURN, all resources in the database will be included in the tabulation. Otherwise the system will ask SELECT RESOURCES BY WHICH PART?. Here the user can enter any indexed PART specification that will be used to determine which resources are included in the tabulation.

Below is an example of using the TR command to tabulate the frequency with which certain keywords are associated with all resources in the database:

COMMAND?tr

Your request can be processed overnight while you are not on line and

the results sent to you as an EIES message, if you prefer.

PROCESS OVERNIGHT (Y/N)?n

PART FOR TABULATION?keyword=communica...community

TABULATE ALL RESOURCES (Y/N)?y

TABULATION OF EXAMPLE RESOURCES AS OF 12/12/80 10:08 AM
TABULATION INCLUDES ALL RESOURCES IN DATABASE

RESOURCES	PERCENT	KEYWORD
-----	-----	-----
14	9.95%	COMMUNICATION
49	22.01%	COMMUNITY ACTION
10	7.80%	COMMUNITY CENTERS
13	9.54%	COMMUNITY WORK

80 KEYWORDS OF 154 RESOURCES IN EXAMPLE

When used to tabulate NUMBER PARTS, the TR command will check the maximum content for each NUMBER PART to be tabulated. If the number has an undefined range, the system will compute the minimum and maximum observed contents and the average content. If the number has a defined range (minimum and maximum), the system will count the number of resources with each possible content. In either case, resources with a UNKNOWN CONTENTS will be counted separately and excluded from the computation of percentages and averages.

When used to tabulate TYPE PARTS, the TR command will count the number of resources with each possible TYPE NAME content. As with NUMBERS, resources with UNKNOWN CONTENTS will be counted separately and excluded from the computation of percentages.

Below is an example showing tabulation of NUMBERS and TYPES on a subset of the entire database:

```
COMMAND?tr
Your request can be processed overnight while you are not on
line and
the results sent to you as an EIES message, if you prefer.
PROCESS OVERNIGHT (Y/N)?n
PART FOR TABULATION?population;people;type;+
TABULATE ALL RESOURCES (Y/N)?n
SELECT RESOURCES BY WHICH PART?date=12/1/80...12/31/80
```

TABULATION OF EXAMPLE RESOURCES AS OF 12/31/80 12:00 PM
 TABULATION INCLUDES ALL RESOURCES WITH DATE=12/1/80...12/31/80

POPULATION:

RANGE:	128	TOTAL:	158		
MINIMUM:	1000	MAXIMUM:	109832	AVERAGE:	20324
UNKNOWN:	8		16.0%		

PEOPLE:

CONTENT	COUNT	PERCENT
0	0	.0%
1	10	20.0%
2	35	70.0%
3	1	2.0%
TOTAL	46	92.0%
UNKNOWN:	4	8.0%

TYPE:

CONTENT	COUNT	PERCENT
URBAN	30	60.0%
SUBURBAN	8	16.0%
RURAL	12	24.0%
TOTAL	50	100.0%
UNKNOWN:	0	0.0%

Tabulations can also be performed on LABEL PARTS in the database. The tabulation can be done on the entire database or just a subset. Tabulating label parts is useful when attempting to count up the number of occurrences of a string of characters. A unique entry is made in the table for each set of label contents. An example is shown below:

```
COMMAND?tr
PART FOR TABULATION?zip code;+
TABULATE ALL RESOURCES (Y/N)?y
```

```
TABULATION OF EXAMPLE RESOURCES AS OF 12/12/80 10:15 AM
TABULATION INCLUDES ALL RESOURCES IN DATABASE
```

ZIP CODE	RESOURCES	PERCENT
01024	2	0.20%
01025	3	0.30%
.	.	.
.	.	.
.	.	.
TOTALS	1000	100.00%

NOTE 1: the printout for TYPES must be able to handle type name CONTENTS of varying lengths. To allow for reasonably long names, the column above labeled COUNT should start in the 21st column position.

NOTE 2: the title of this report includes the PART specification used to define the subset of resources included in the tabulations.

3.6.3 TABULATING FREQUENCY OF ACCESSES TO RESOURCES (TD)

The TD (Tabulate and report on Database statistics) command will compute and print a report showing how many times a resource was accessed (SYSTEM ACCESSES), how many comments have been entered associated with each resource (SYSTEM COMMENTS), and how many tally ratings have been entered in response (SYSTEM TALLIES). It also reports the date each resource was entered (SYSTEM DATE), the date each was last modified, if modified (SYSTEM MDATE), the date each was last accessed, if ever accessed (SYSTEM ADATE), and the EIES ID of the user who entered the resource (SYSTEM SOURCE). Resources are identified in the reports by their PRIMARY ID as detailed in (2.4.4).

When the TD command is entered, the system will first ask PROCESS OVERNIGHT to give the user an opportunity to get the results the next day in a message. Then the system will ask TABULATE ALL RESOURCES (Y/N)? If the user presses RETURN or enters Y (for Yes), all resources will be reported, otherwise the system will ask SELECT RESOURCES BY WHICH PART? to which the user can respond with any indexed PART to specify a subset of the entire database of resources.

Below is an example showing the format of the printout of this command.

```
COMMAND?td
Your request can be processed overnight while you are not on line and
```

the results sent to you as an EIES message, if you prefer.

PROCESS OVERNIGHT (Y/N)?n

TABULATE ALL RESOURCES (Y/N)?n

TABULATE RESOURCES BY WHICH PART?date=05/01/80...05/31/80

EXAMPLE DATABASE: ACCESS STATISTICS AS OF 6/19/80 12:12 PM
 REPORT INCLUDES ALL RESOURCES WITH DATE=05/01/80...05/31/80

Primary ID	System Accss./	System Cmnts./	System Tallys	System Source	Date Entered	Date Last Modified	Date Last Accessed
P#303	13	9	0	653	5/19/80	---	6/14/80
P#304	2	13	4	653	5/19/80	5/21/80	6/17/80
TOTALS	345	179	79				

NOTE: the title of this report includes the PART specification used to define the subset of resources included in the tabulations.

3.6.4 TABULATING USER STATISTICS (TU)

The TU (Tabulate and report on User statistics) command will compute and report on the activity of users. When the TU command is entered the system will first ask PROCESS OVERNIGHT (Y/N)? and then USERS (#,#...#)? Here the user can either press RETURN to have the report include all users or can specify specific users or ranges of users. Then the system will compute and report on user activity, including level of access, times in the system, date/time last active, resources gotten and added, and comments gotten and added, as shown below.

COMMAND?tu

Your request can be processed overnight while you are not on line and the results sent to you as an EIES message, if you prefer.

PROCESS OVERNIGHT (Y/N)?n

USERS (#,#...#)?

EXAMPLE DATABASE: USER STATISTICS AS OF 6/19/80 12:12 PM

USER ID	Level of Access	Times in Database	Date/Time Last Active	Resources adds/gets	Comments adds/gets
118	60	44	6/19/80 12:12 PM	4/60	3/75
653	50	129	6/19/80 8:01 AM	99/354	87/413
...
anon.	13
TOTALS		932		432/893	434/998

NOTE 1: a special row in this report is needed to report on comments added under a pen/anonymous identity.

NOTE 2: if switch 14 is on, the system goes to the top of the next page before printing the report of user statistics.

3.6.5 CROSS TABULATING RESOURCES BY PARTS (CTR)

The cross tabulate feature was added to enhance the tabulating abilities of data stored in the resource system. Cross tabulate allows tabulation of one part's content against another part in the database. The contents of the first part can be any KEYWORD, NUMBER, TYPE or LABEL part in the format. The field is only limited to eight characters in the final print-out field. The second part field is the part contents to be tabulated against. This field can be any part in the format (except text). This field is limited to fifteen characters, and can consist of up to 400 unique values. Like TR, cross tabulations can be performed on any range as well as the entire database. Below is an example of cross tabulating a number part against a keyword part:

```
COMMAND?ctr
PART FOR TABULATION?peo
CROSS TABULATE BY WHICH PART?key=communica...community
TABULATE ALL RESOURCES (Y/N)?y
```

CROSS TABULATION OF RESOURCES IN EXAMPLE DATABASE AS OF 12:29

AM

TABULATION INCLUDES ALL RESOURCES IN DATABASE

KEYWORD:	PEOPLE:				TOTALS
	1	2	3	4	
COMMUNICATION	2	1	11	0	14
COMMUNITY ACTI	13	9	18	10	50
COMMUNITY CENT	5	0	3	2	10
COMMUNITY WORK	4	7	2	0	13
TOTALS	24	17	34	12	87
	13.49%	10.55%	25.00%	8.05%	100.00%

3.7 FEEDBACK COMMANDS

The RESOURCES system allows any user with level 10 or greater access to enter feedback on resources into the system for subsequent exchange with other users. This feedback can take on two forms: (1) comments in the feedback file (in effect, a simple computer conference, one per database), and (2) ratings of resources on scales defined by a database monitor. Each of these forms uses different commands, explained in detail below.

3.7.1 ENTERING COMMENTS IN THE FEEDBACK FILE (AC)

When the AC (Add Comment) command is entered, the system will ask ASSOCIATED RESOURCE? and then put the user into the scratchpad as shown below. If the user does not wish to associate the comment with a particular resource, the user may simply press RETURN to indicate the comment is a general one. Otherwise, the user should enter a PART specification that uniquely defines a single resource in the database. Any indexed PART can be used for this purpose, although usually an ID PART is chosen for obvious reasons. Note that if a comment is associated with a specific resource, the system prints out the SYSTEM ID line and the DISPLAY PARTS as a double check. Note also that the system asks for a TITLE? for each comment.

```
COMMAND?ac
ASSOCIATED RESOURCE?id=p#303
COMMENT TO BE ASSOCIATED WITH:
EXAMPLE 432 6/18/80 12:12 PM SOURCE: 118
```

```
ID: P#303
NAME: Peter and Trudy Johnson-Lenz
ADDRESS: 695 Fifth Street, Lake Oswego, Oregon 97034
PHONE: (503) 635-2615
```

ENTERING SCRATCHPAD:

```
1?(first line of comment)
```

```
...
```

```
...
```

```
15?+
```

```
TITLE?our reactions to P+T
```

```
TITLE: OUR REACTIONS TO P+T
```

```
OK TO ADD COMMENT (Y/N)?y
```

```
ADDED AS:
```

```
EXAMPLE 219 PETER+TRUDY JOHNSON-LENZ (P+T,118) 6/19/80 9:32
```

```
PM
```

```
ASSOCIATED RESOURCE 432: ID=P#303
```

```
COMMAND?
```

NOTE: when the comment is added the system lists the first line of the header which begins with the database name, followed by a unique comment ID numbers (from 1 up to the highest entered to date within a database), the name, nickname and number of the author and the date and time. If the comment is associated with a particular resource, a second line is included showing BOTH the SYSTEM ID of the associated resource, AND the PART specification entered by the user to associate the comment. Both of these lines are part of the permanent header of the comment. If the comment is not associated with any resource, the second line reads GENERAL COMMENT.

3.7.2 ENTERING PEN OR ANONYMOUS COMMENTS

When the system asks OK TO ADD COMMENT (Y/N)? and the user responds with N, the system will normally say COMMENT NOT ADDED and go to COMMAND? However, if the PEN/ANONYMOUS COMMENT option has been enabled by the monitor, the system will ask instead USE PEN NAME (Y/N)? If the user responds with a N, the system will repeat the OK TO ADD COMMENT (Y/N)? question, but if the user responds with a Y, the system will ask for PEN NAME? Then it will feed back the pen name entered enclosed between double quotation marks as a double-check (as shown in the partial example below) and then ask OK TO ADD COMMENT (Y/N)? again.

```
OK TO ADD COMMENT (Y/N)?n
USE PEN NAME (Y/N)?y
PEN NAME?the resource wizard
Pen name of "THE RESOURCE WIZARD" used.
OK TO ADD COMMENT (Y/N)?y
```

When a comment is entered with a pen name or anonymously (pen name of "ANONYMOUS"), the system will keep track of its author to allow modification and removal by the author, but it will not reveal the author's identity to any other user, even to those with higher levels of access, regardless of which command they may use to print or display statistics and information about the comment or the author. Such pen/anon comments are NOT included in any statistics by author, but rather under a special pen/anon category only.

NOTE: If the Pen/anonymous switch is off (see AD and MD in (3.10)), then the comment will not be added if No is the response to OK TO ADD COMMENT (Y/N).

3.7.3 TRACE OF COMMENTS ADDED

If the option to trace user activities in the database is turned on, the system will record the date and time the comment was added, the comment ID number, and the associated resource (identified by its PRIMARY ID) in the user's trace file.

Whether or not the trace option is on, the system will also increment the count of comments added by the user as well as the count of comments associated with the specific resource.

The AC command also increments the total count of comments in the database by one when a comment is added.

3.7.4 MODIFYING COMMENTS (MC)

The MC (Modify Comment) command may be used to make changes in comments already entered. This command may be used only by the original author of the comment or by someone with EDITOR (level 30) or higher access to the database. MC will ask for the the COMMENT ID (#)?, then print the comment header as a double-check. The MC command takes the user to the scratchpad where the comment will be waiting. When + is entered to terminate the modification, the system asks OK TO MODIFY (Y/N)? Then, in either case, the system asks if the TITLE is TO BE MODIFIED (Y/N)? and if so lists the current title and asks for NEW TITLE?, feeds it back and then asks OK TO MODIFY TITLE (Y/N)?

3.7.5 REMOVING COMMENTS (RC)

The RC (Remove Comment) command may be used to delete comments entered into the database. This command may be used only by the original author of the comment or by someone with EDITOR (level 30) or higher access to the database. Remove comment will ask for the the COMMENT ID (#)?, then print the comment header as a double-check. Then, RC asks OK TO REMOVE COMMENT (Y/N)? as a double-check. When a comment is removed, the backpointers in the resource to which it points are also removed to prevent errors later on. When a comment is removed, the total number of comments in the database is decremented by one.

3.7.6 GETTING COMMENTS FROM THE FEEDBACK FILE (GC)

The GC (Get Comment) command can be used to retrieve comments from the feedback file for a database. This command will print out the comment header and the full text as well. When the GC command is entered, the system responds by asking ASSOCIATED RESOURCE(S)?. Here the user enters a PART specification to which the system responds by printing out all comments associated with each resource in the list of resources defined by the PART specification. All comments for a given resource are printed out together. However, if the user presses RETURN in response to the ASSOCIATED RESOURCE(S)? question, the system then ask COMMENTS (#,#...#)?, allowing the user to get comments by their own ID numbers.

If a range of comments is specified by the user, and if the range is specified in ascending order (such as 123...150) the comments will be printed out in that ascending order. On the other hand, if the range is specified in descending order, the comments will be listed from most recent to least recent so that the user may

press BREAK to terminate the output when older comments begin to appear.

After printing the specified comments, the system will continue to ask ASSOCIATED RESOURCE(S)? or COMMENTS (#,#...#)?, depending on which one the user last responded to, until the user enters a +.

Each time a comment is printed out with the GC command, the count of comments gotten for the user will be incremented by one.

3.7.7 DISPLAYING COMMENT HEADERS ONLY (DC)

The DC (Display Comment) command can be used to retrieve comments from the feedback file for a database much like the GC command. The only difference between the two is that DC prints out only the header of the comments while GC prints out the full Text. Below is a sample interaction with the DC command.

```
COMMAND?dc
ASSOCIATED RESOURCE(S)?
COMMENTS (#,#...#)?219
```

```
EXAMPLE 219 PETER+TRUDY JOHNSON-LENZ (P+T,118) 6/19/80 9:32
```

PM

```
ASSOCIATED RESOURCE 432: ID=P#303
TITLE: OUR REACTIONS TO P+T
```

```
COMMENTS (#,#...#)?+
```

```
COMMAND?
```

Please refer to the section on GC (section 3.7.6) for list and range conventions for the Display Comment command. The interaction is the same but the output is different.

3.7.8 PRINTING COMMENTS WAITING (PC)

The PC (Print Comments waiting) command can be used to get up to date in the feedback file of comments. When entered by a user, this command will start printing out comments from the user's MARKER on forward, just as in a regular computer conference, and will move that marker as comments are delivered.

The PC command will increment the count of comments gotten by the user once for each comment delivered.

Normally the PC command will NOT deliver a user's own comments. However, if EIES switch 6 has been turned on by the +SSW 6 command,

a user's comments will be included in the delivery along with all other comments. The +RSW 6 command can be used to turn off this feature.

3.7.9 MODIFYING COMMENT MARKERS (MM)

The MM (Modify Markers) command allows a user to move his or her comment marker forward or backward. When the MM command is entered it lists the current marker (last comment received), the highest comment number used to date, and then asks for the NEW LAST COMMENT? When the new marker is entered, the system feeds it back and asks OK TO MODIFY MARKER (Y/N)?

```
COMMAND?mm
MODIFY COMMENT MARKER (Y/N)?y
Last comment received: 200
Highest comment entered to date: 234
NEW LAST COMMENT (#)?219
New last comment received: 219
OK TO MODIFY MARKER (Y/N)?y
MARKER MODIFIED.
```

```
MODIFY RESOURCE MARKER (Y/N)?n
```

```
COMMAND?
```

3.7.10 COPYING INTO THE SCRATCHPAD (+RCY) AND (+RCIN)

Two commands have been provided that allow copying of comments into the scratchpad from either inside or outside of RESOURCES. These commands, +RCY and +RCIN, are explained in (3.11).

3.7.11 ADDING TALLIES TO THE FEEDBACK FILE (AT)

The AT (Add response to Tally) command allows a user to enter responses to the tally questions defined by the database monitor when the database was added in order to rate resources previously accessed in a database. Section (3.8.2), Modifying Questions in a Tally, explains how these questions are specified.

When the AT command is entered, the system asks ASSOCIATED RESOURCE? If the user enters a PART specification that defines a unique resource, the system prints out the SYSTEM ID line and display parts of the resource and then paces the user through the

tally as shown below:

COMMAND?at
ASSOCIATED RESOURCE?id=p#303

TALLY FOR:
EXAMPLE 432 6/18/80 12:12 PM SOURCE: 118

ID: P#303
NAME: Peter and Trudy Johnson-Lenz
ADDRESS: 695 Fifth Street, Lake Oswego, Oregon 97034
PHONE: (503) 635-2615

QUESTION 1:
On a scale from 1 (not at all) to 7 (extremely), how useful was this resource to you?

RESPONSE OR 0 TO ABSTAIN (#)?7

TALLY AS OF 6/18/80 12:12 PM: 29 responses:
Response categories: abs 1 2 3 4 5 6 7
Number of responses: 3 1 1 2 4 9 7 2

QUESTION 2:
On a scale from 1 (not at all) to 7 (extremely), in your experience how unique or difficult to find was this resource?

RESPONSE OR 0 to ABSTAIN (#)?7

TALLY AS OF 6/18/80 12:13 PM: 29 responses:
Response categories: abs 1 2 3 4 5 6 7
Number of responses: 3 1 1 2 2 3 7 9

COMMAND?

Each time a user adds ratings to a tally for the FIRST time, the count of users who have responded to the tallies is incremented by one for the particular resource involved. This counter is NOT incremented when the AT command is used to change a previously entered rating as detailed below.

The AT command may be used to change responses to tallies previously responded to. When the AT command is used for a tally that a user has already responded to the only difference is that the system feeds back the user's previous response before asking for a new one, as shown in the partial example below:

QUESTION 2:
On a scale from 1 (not at all) to 7 (extremely), in your experience how unique or difficult to find was this resource?

PREVIOUS RESPONSE WAS: 7
PRESS RETURN TO LEAVE PREVIOUS RESPONSE UNCHANGED

RESPONSE OR 0 TO ABSTAIN (#)?

3.7.12 DISPLAYING TALLY RATINGS (DT)

The DT (Display Tally) command may be used to display the results of a feedback tally. The DT command, unlike the GT command only prints the tally results but not the text of the question. Since the same questions are asked for each resource within a database, the DT command can be used to avoid printing out the same questions over and over. When the DT command is entered the system will first ask ASSOCIATED RESOURCE(S)? and will then print out the current tally results for all resources specified by whatever PART specification the user enters. After printing the tally results, the system will continue asking ASSOCIATED RESOURCE(S)? until the user enters a + or just presses RETURN.

3.7.13 GETTING TALLY QUESTIONS AND RATINGS (GT)

The GT (Get Tally) command may be used to display the results of a feedback tally. The difference between the GT and DT commands is that GT prints the tally results as well as the text of the question. When either command is entered the system will first ask ASSOCIATED RESOURCE(S)? and will then print out the current tally results for all resources specified by whatever PART specification the user enters. After printing the tally results, the system will continue asking ASSOCIATED RESOURCE(S)? until the user enters a + or just presses RETURN.

COMMAND?gt
ASSOCIATED RESOURCE(S)?id=p#303

TALLY FOR:
EXAMPLE 432 6/18/80 12:12 PM SOURCE: 118

ID: P#303
NAME: Peter and Trudy Johnson-Lenz
ADDRESS: 695 Fifth Street, Lake Oswego, Oregon 97034
PHONE: (503) 635-2615

QUESTION 1:
On a scale from 1 (not at all) to 7 (extremely), how useful was this resource to you?

TALLY AS OF 6/18/80 12:12 PM: 29 responses:
Response categories: abs 1 2 3 4 5 6 7
Number of responses: 3 1 1 2 4 9 7 2

QUESTION 2:

On a scale from 1 (not at all) to 7 (extremely), in your experience how unique or difficult to find was this resource?

TALLY AS OF 6/18/80 12:13 PM: 29 responses:

Response categories:	abs	1	2	3	4	5	6	7
Number of responses:	3	1	1	2	2	3	7	9

COMMAND?

3.8 FORMAT MANAGEMENT COMMANDS

A database monitor must specify the FORMAT for a database once the database itself has been added. This must be done before any resources are entered. The AF (Add Format) command is used for this purpose. In order to make a database usable, the monitor must define the FORMAT and specify any TALLY questions that are to be associated with resources in the database. The AF command will guide the monitor through the first of these tasks. The MT command can be used to do the second. Once so defined the FORMAT may be modified later with the MF (Modify Format) command. The TALLY may be modified with the MT (Modify Tally) command.

Once a DATABASE has been added and the FORMAT and TALLY have been specified, data can be entered into that database using the commands in Section 3.1. The DF (Display Format) and GF (Get Format) commands may be used to display or get a complete printout of the FORMAT for a database as a reminder of what the format specifications are. The XF (Xmit Format) command may be used to transmit the FORMAT to a microcomputer. All of these commands are described in detail below.

3.8.1 ADDING A FORMAT TO A DATABASE (AF)

When a monitor enters the AF command the system will respond by asking for the first PART NAME? and KIND (D/I/K/L/N/T/X)? The KIND of PART is indicated by a single letter (D for DATE, I for ID, K for KEYWORD, L for LABEL, N for NUMBER, T for TYPE, and X for TEXT). All parts can be multiply defined as long as they are assigned unique PART NAMES. Furthermore, parts can be defined in any order without regard to the KIND of part, so that DATE, ID, KEYWORD, LABEL, NUMBER, TYPE, and TEXT parts can be in any order that is desired. Parts will be prompted by the RESOURCE EDITOR in the order they are specified in the FORMAT. Resources will also be output in FORMAT order unless a TEMPLATE is used to change that order.

The only restrictions are that a PART NAME can be used only

once within a given database and that only ONE KEYWORD part can be specified (one is sufficient since any number of KEYWORDS can be entered into a single part). Depending on the KIND of part being specified, the system will ask for additional qualifying information.

If a DATE part is being specified, the system will ask only if the date part is to be INDEXED (Y/N)?

If an ID part is being specified, the system will ask no additional questions, since all ID parts are ALWAYS indexed and since the only check performed on ID parts is for uniqueness of the identified resource within the database.

If a KEYWORD part is being specified, the system will ask for the MAXIMUM NUMBER OF KEYWORDS (#)? per resource. Since all KEYWORDS are ALWAYS indexed, nothing more will be asked.

If a LABEL part is being specified, the system will ask for the MAXIMUM CHARACTERS (#)? per entry. Since LABELS are not automatically indexed (see Section 2.4), the system will ask if the label part is to be INDEXED (Y/N)?

If a NUMBER is being specified, the system will ask for the MINIMUM and MAXIMUM contents, as well as the NUMBER OF DECIMAL PLACES to be specified. Numbers with defined minima and maxima can be tabulated to show the frequency of EACH content. Numbers without any defined range of contents cannot be tabulated in this way -- only the average, minimum, and maximum contents observed during tabulation are reported in tabulations of such numbers (see Section 3.4 for details). The minimum and maximum contents are also used, if provided, for checking contents during data entry. If the monitor does not know the minimum and maximum contents of a NUMBER, s/he should press RETURN which will leave them undefined. Since NUMBERS are not automatically indexed (see Section 2.4), the system will ask if the number part is to be INDEXED (Y/N)?

If a TYPE is being specified, the system will ask for the NUMBER OF TYPES (#)? and then ask for the type names, one at a time, separated by commas. Since TYPE parts are not automatically indexed (see Section 2.4), the system will ask if the type part is to be INDEXED (Y/N)?

If a TEXT part is being specified, the system will ask for the MAXIMUM LINES (#)? per item.

The system will continue asking KIND (D/I/K/L/N/T/X)? until the monitor enters a + to terminate part definition.

The last specification question the system asks during format definition is the number of parts (starting with the first part in the FORMAT) to be included in the list of DISPLAY PARTS printed out with the DR command, the D options of the FP and FR commands, the AC command, and the AT, DT, and GT commands.

After the number of DISPLAY PARTS has been specified, the system asks if the monitor wishes to PRINT OUT FORMAT (Y/N)? as a way of checking it. If the monitor enters Y or presses RETURN, the system will list the FORMAT in the same way that the GF command (shown below) does. Whether or not the FORMAT is printed out, the system then asks OK TO ADD FORMAT (Y/N)? If there were any mistakes, the monitor can enter N at this point and the FORMAT will not be added to the database. In the event of such mistakes, the monitor must start the entire FORMAT-adding process FROM THE BEGINNING.

Below is an example showing the use of the AF command for specifying the FORMAT for a database:

```

COMMAND?af

PART NAME?id
KIND (D/I/K/L/N/T/X)?i
PART NAME?date
KIND (D/I/K/L/N/T/X)?d
INDEXED (Y/N)?y
PART NAME?description
KIND (D/I/K/L/N/T/X)?x
MAXIMUM LINES (#)?20
PART NAME?name
KIND (D/I/K/L/N/T/X)?L
MAXIMUM CHARACTERS (#)?32
INDEXED (Y/N)?y
PART NAME?address,L,64,n
PART NAME?phone,L,12,n
PART NAME?occupation
KIND (D/I/K/L/N/T/X)?n
MINIMUM CONTENT (#)?1
MAXIMUM CONTENT (#)?100
NUMBER OF DIGITS AFTER THE DECIMAL?0
INDEXED (Y/N)?n
PART NAME?people
KIND (D/I/K/L/N/T/X)?n
MINIMUM CONTENT (#)?0
MAXIMUM CONTENT (#)?10
NUMBER OF DIGITS AFTER THE DECIMAL?
INDEXED (Y/N)?n
PART NAME?population
KIND (D/I/K/L/N/T/X)?n
MINIMUM CONTENT (#)?
MAXIMUM CONTENT (#)?
NUMBER OF DIGITS AFTER THE DECIMAL?
INDEXED (Y/N)?n
PART NAME?type
KIND (D/I/K/L/N/T/X)?t
NUMBER OF TYPES (#)?3
TYPE NAME?urban
TYPE NAME?suburban
TYPE NAME?rural
INDEXED (Y/N)?y
PART NAME?keywords,k,20

```

PART NAME?+

NUMBER OF DISPLAY PARTS (#)?4

PRIMARY ID (PART)?id

PRINT OUT FORMAT (Y/N)?n

OK TO ADD FORMAT (Y/N)?y

FORMAT ADDED TO EXAMPLE DATABASE.
DATABASE READY FOR DATA ENTRY.

COMMAND?

If the format for the database already exists and the moderator wants to add new parts to the format, then AF can again be invoked. The system will respond with FORMAT CONTAINS XX PARTS and then continue from the position after the last part in the format. All existing resources in the database will report the new part as blank, but the RESOURCE EDITOR will prompt with the new part if in PROMPTED MODE. MR can be used to fill in the new part contents for the resources that have been added prior to the addition of the new part to the format.

COMMAND?af

FORMAT CONTAINS 10 PARTS

PART NAME? .

.

.

PART NAME?+

OK TO EXTEND FORMAT (Y/N)?y

Format has been extended to 11 parts.

COMMAND?

3.8.2 MODIFICATION OF FORMAT (MF)

Once a database has been added and resources entered into it, it is impossible to change the number and order of parts in the FORMAT since the resources already entered have been formatted. However, the names of parts can be changed as can their maxima. However, care must be taken in doing this, particularly in changing the maximum since resources may already have been entered with contents or lengths that exceed the new maximum, if the new one is less than the previous one. In such cases editing of some resources may be needed. The MF (Modify Format) command can be used to modify these names and maxima and also the TYPE names of TYPES as shown in the example, below. It can also be used to change the number of parts included in the list of DISPLAY PARTS, and modifying which

part is the PRIMARY ID.

```

COMMAND?mf
PART TO BE MODIFIED?phone
CURRENT MAXIMUM FOR PHONE IS 12
NEW PART NAME?telephone
NEW MAXIMUM CHARACTERS?20
NEW PART: TELEPHONE (20 CHARACTERS)
OK TO MODIFY (Y/N)?y
PART TELEPHONE MODIFIED.
PART TO BE MODIFIED?type
CURRENT NUMBER OF TYPES IS 3
NEW PART NAME?
NEW NUMBER OF TYPES?4
NEW TYPE NAME?NONE
MODIFY TYPE NAMES (Y/N)?y
NEW NAME FOR URBAN?
NEW NAME FOR SUBURBAN?
NEW NAME FOR RURAL?COUNTRY
NEW NAME FOR NONE?
NEW PART: TYPE(4 TYPES)
OK TO MODIFY (Y/N)?y
PART TYPE MODIFIED.
PART TO BE MODIFIED?+

MODIFY NUMBER OF DISPLAY PARTS (Y/N)?y
NUMBER OF DISPLAY PARTS (#)?5

MODIFY PRIMARY ID (Y/N)?n

COMMAND?

```

NOTE: as shown in the above example, when modifying the FORMAT with the MF command, a user can press RETURN to leave something unchanged.

3.8.3 MODIFYING QUESTIONS IN A TALLY (MT)

Before any feedback tally ratings can be entered in response to a resource, the monitor must specify the questions to be asked. Furthermore, since the tally commands accept ratings from 1 up through 9 or some smaller number specified by the monitor for each question, these maxima must also be defined. The MT (Modify Tally) command is used both to initially specify these questions and to modify them later on. The very first question this command asks is whether the monitor wishes to ADD/MODIFY/REMOVE QUESTION (A/M/R)? Adding a question simply adds it to the list of questions asked and sets the tally counts to zero. Removing a question destroys tally counts to date. Modification of a tally question does not affect tally counts. To zero out a tally, the question must be removed and added again. In such cases all counts for that question for ALL RESOURCES in the database will be lost.

Below is an example interaction with the MT (Modify Tally) command.

COMMAND?mt
 ADD/MODIFY/REMOVE QUESTION (A/M/R)?a
 ENTER QUESTION:

1?.text sk
 2? On a scale from 1 (not at all) to 7 (extremely) how timely
 for
 3?you was the receipt of this resource?
 4?+

MAXIMUM RESPONSE (#)?7
 OK TO ADD (Y/N)?y

Question 3 added to tally for database.

ADD/MODIFY/REMOVE QUESTION (A/M/R)?r
 QUESTION TO BE REMOVED (#)?2

Question 2:

On a scale from 1 (not at all) to 7 (extremely) how unique
 and difficult to find is this resource?
 OK TO REMOVE (Y/N)?y

Question 2 removed from tally for database.

ADD/MODIFY/REMOVE QUESTION (A/M/R)?m
 QUESTION TO BE MODIFIED (#)?3
 MODIFY TEXT OF QUESTION:

4?/timely for/timely was/2
 2? On a scale from 1 (not at all) to 7 (extremely) how timely
 was
 4?=3
 3?you was the receipt of this resource?
 3?receipt of this resource for you?
 4?+

MAXIMUM RESPONSE (#)?7
 OK TO MODIFY QUESTION 3 (Y/N)?y

Question 3 modified in tally for database.

ADD/MODIFY/REMOVE QUESTION (A/M/R)?+

COMMAND?

3.8.4 GETTING A PRINTOUT OF THE FORMAT (GF)

The GF (Get Format) command will list the complete format in detail showing each part by name, its type, whether it is indexed or not, and its minimum and maximum where meaningful.

Previously unindexed parts can become indexed parts if users attempt a get or find operation using them. Therefore, the list of parts printed out by the GF command may well show parts as indexed that were not initially indexed. When the GF command is entered the system prints out the format for that database as shown in the example, below:

COMMAND?gf

FORMAT FOR EXAMPLE DATABASE:

PART NAME	INDEXED	TYPE	CHARACTERISTICS
-----	-----	-----	-----
ID	Y	ID	
DATE	Y	DATE	
DESCRIPTION	N	TEXT	20 LINES
NAME	Y	LABEL	32 CHARS.
ADDRESS	N	LABEL	64 CHARS.
TELEPHONE	N	LABEL	20 CHARS.
OCCUPATION	N	NUMBER	MIN=1, MAX=100
PEOPLE	N	NUMBER	MIN=1, MAX=10
POPULATION	N	NUMBER	MIN=NO LIMIT, MAX=NO LIMIT
TYPE	Y	TYPE	4 TYPES: URBAN, SUBURBAN COUNTRY, NONE
KEYWORDS	Y	KEYWORDS	20 KEYWORDS

NUMBER OF DISPLAY PARTS: 4

PRIMARY ID: ID

COMMAND?

NOTE: if the TYPE names for TYPE PARTS require more than a single line to be listed within an 80-column format, they should be wrapped around onto the next line and indented to begin in the CHARACTERISTICS column for as many lines as needed to list them without breaking a name in the middle.

3.8.5 TRANSMITTING THE FORMAT TO A MICROCOMPUTER (XF)

The XF (Xmit Format) command can be used to transmit the FORMAT definition to a microcomputer for use there in offline data entry prior to transmission of resources from the microcomputer to EIES with the ER (Enter Resources) command. When the XF command is entered, the system begins transmitting the FORMAT as shown in the example below:

```
COMMAND?xf
ID,I
DATE,D,Y
DESCRIPTION,X,20
NAME,L,32,Y
ADDRESS,L,64,N
TELEPHONE,L,20,N
OCCUPATION,N,1,100,N
PEOPLE,N,1,10,N
POPULATION,N,,N
TYPE,T,4,URBAN,SUBURBAN,COUNTRY,NONE,Y
KEYWORDS,K,20
+,4,N,Y
COMMAND?
```

NOTE 1: each line transmitted corresponds to a PART in the FORMAT. Each line begins with the PART NAME followed by a single letter abbreviation for the type of part (D for DATE, I for ID, K for KEYWORD, L for LABEL, N for NUMBER, T for TYPE, and X for TEXT). This is followed by relevant parameters in the same order as they would be entered to define the parts using the AF command. The list of parameters includes the minima, maxima, type names, and indexing status for DATE, LABEL, NUMBER, and TYPE PARTS.

NOTE 2: the undefined range of the POPULATION number is represented by three consecutive commas as shown in the example above. This printout is therefore an exact duplicate of the inputs required to specify the format.

NOTE 3: the transmission ends with a line beginning with a +, followed by the number of DISPLAY PARTS, followed by N,Y -- the same way a monitor would normally end FORMAT definition with the AF command.

3.9 USER MANAGEMENT COMMANDS

A gatekeeper may use the AU (Add User) command to add users, the MU (Modify User) command to modify a user's access level, and the RU (Remove User) command to remove a user. Any user may use the DU (Display User) and GU (Get User) commands to display user information. Any user may use the DA (Display Activity) and GA (Get Activity) commands to display his or her own activity trace and a gatekeeper may use these commands to display any user's trace.

3.9.1 ADDING A USER (AU)

The AU (add user) command can be used to grant users access to the database. Gatekeeper access is required and access levels cannot be given out unless the person granting them has those privileges. The example below shows how the system responds when the AU command is entered:

```
COMMAND?au
USER TO BE ADDED (#)?10113
PETER+TRUDY JOHNSON-LENZ (P+T,10113)
ACCESS LEVEL FOR PETER+TRUDY JOHNSON-LENZ (#)?20
Access level: 20
OK TO ADD PETER+TRUDY JOHNSON-LENZ TO EXAMPLE (Y/N)?y
PETER+TRUDY JOHNSON-LENZ added to EXAMPLE.
USER TO BE ADDED (#)?+
```

```
COMMAND?
```

3.9.2 MODIFYING USERS IN THE DATABASE (MU)

The MU command can be used to modify a user's access level. Gatekeeper privileges to the database are needed to use this command and levels of access greater than that of the user making the modification cannot be granted. An example of MU is shown below:

```
COMMAND?mu
USER TO BE MODIFIED?10113
PETER+TRUDY JOHNSON-LENZ (P+T,10113)
Access level: 20
NEW ACCESS LEVEL (#)?30
New access level: 30
OK TO MODIFY PETER+TRUDY JOHNSON-LENZ (Y/N)?y
PETER+TRUDY JOHNSON-LENZ modified.
USER TO BE MODIFIED?+
```

3.9.3 REMOVING USERS FROM THE DATABASE (RU)

The RU (remove user) command is used to take users out of the database. When this is done all of the user's user files are removed, but not any resources or comments the removed user has added. The RU command is shown as follows:

```
COMMAND?ru
USER TO BE REMOVED?l0133
MIDNIGHT MAPPER (MAPPER,l0133)
OK TO REMOVE (Y/N)?y
MIDNIGHT MAPPER removed from the EXAMPLE database.
USER TO BE REMOVED?+
```

```
COMMAND?
```

3.9.4 DISPLAYING DATABASE USER LIST (DU)

The DU command can be used to make a list of all users of a database showing their full names, nicknames, and numbers, date/time last active, and access levels.

```
COMMAND?du
```

```
USERS OF THE EXAMPLE DATABASE AS OF 6/19/80 12:12 PM
```

User	Date Last Active	Access Level
PETER+TRUDY JOHNSON-LENZ (P+T,118)	6/19/80 12:12 PM	60
ROXANNE HILTZ (ROXANNE,120)	6/18/80 10:10 AM	20
MILES MARTIN (MILES,653)	6/18/90 5:14 PM	50
SUSAN HYATT (SUSAN,654)	6/19/80 10:19 AM	50
JOHN BRIGGS (TAVI,882)	6/18/80 5:55 PM	20

```
<BREAK>
```

NOTE: if switch 14 is on, the DU command goes to the top of the next page before printing out the list of users.

3.9.5 GETTING USER INFORMATION (GU)

The GU command can be used to print out the same information as DU, plus a few other details for specified users. These are shown below:

```
COMMAND?gu
USER (#)?10113
```

```
PETER+TRUDY JOHNSON-LENZ (P+T,10113)   Last Active:  6/18/80
12:12 PM
Access Level:  30,   Times in database:  43,   Comment marker: 129
Resources added:    17,   Resources gotten:   127
```

```
USER (#)?+
```

```
COMMAND?
```

If the users access level is less than 40, the question asking for user number will not be asked. Instead, only that user's data will be displayed. A carriage return can be used to exit the loop asking for users, and a ? at USER (#)? will return a list of all users in the database.

3.9.6 GETTING A USER'S ACTIVITY TRACE (GA)

The GA command can be used by a gatekeeper to print out the activity trace for any user, while any user may use the command to see his or her own activity trace. When entered by someone with level 40 or higher access it first asks USER (#)?, but if entered by someone with less than level 40 access it does not ask this initial question.

The GA command begins by printing out the total number of trace records that have been written for a particular user. Then it prints out these records in the INVERSE ORDER from MOST RECENT TO LEAST RECENT so that BREAK can be pressed when desired. NOTE that these commands will only work if the activity trace monitor option (3.10.1) has been turned on.

Below is an example of the output from the GA command:

```
COMMAND?ga
USER (#)?10113
```

ACTIVITY TRACE FOR USER 10113 AS OF 6/19/80 11:10 PM
49 ENTRIES IN TRACE:

Command	Date/Time	Resource	Comment
AC	6/19/80 10:15 PM	P#303	219
GR	6/19/80 10:10 PM	P#403	
FP	6/19/80 10:09 PM	P#407	
FP	6/19/80 10:09 PM	P#413	
FP	6/19/80 10:08 PM	P#422	
AR	6/18/80 9:18 AM	P#432	

<BREAK>

COMMAND?

NOTE 1: resources are identified in the output by the PRIMARY ID of the resource. When comments are added and gotten, the trace file first shows the associated resource (if any) and then the COMMENT ID of the comment.

NOTE 2: the trace records not only the date and time an action was taken by the user, but the specific command being used at the time.

NOTE 3: if switch 14 is on, the system will go to the top of the next page before printing the activity trace information.

3.9.7 PRUNING THE USER ACTIVITY TRACE FILES (MA)

Any user with gatekeeper access may use the MA (Modify Activity Trace) command to remove entries from the trace files of specified users that are older than a specified date. The primary purpose of this command is to make it possible to keep the trace files up to date and to minimize the disk space used to maintain them on line.

Because such modification can take a long time if the gatekeeper specifies modification of the traces for a large number of users, the MA command asks, PROCESS OVERNIGHT (Y/N)? Then it asks REMOVE ENTRIES PRIOR TO DATE (mm/dd/yy)? to request a cut-off date for pruning the file of old entries. Finally it asks for USERS (#,#...#)?, to which the gatekeeper may respond with a list of user numbers and ranges, or simply press RETURN to indicate all users in the database.

Below is an example of its use:

COMMAND?ma

Your request can be processed overnight while you are not on line and

the results sent to you as an EIES message, if you prefer.
 PROCESS OVERNIGHT (Y/N)?n
 REMOVE ENTRIES PRIOR TO DATE (mm/dd/yy)?5/01/80
 USERS (#,#...#)?118,653

SUMMARY OF MODIFICATIONS OF TRACE FILES AS OF 12/12/80 12:12 PM

USER	ENTRIES REMOVED	ENTRIES REMAINING
-----	-----	-----
118	129	155
653	300	33

COMMAND?

CAUTION: Activity trace files fill up fast for active users. Each file can only hold about 400 records of information because of the amount of data that has to be stored for each action. Monitors should prune the database often so the files never break. If they do overflow, the user will be denied access to the database.

3.10 RESOURCES SYSTEM MANAGEMENT COMMANDS

The AD (Add Database) command is used to open up a new database. The MD (Modify Database) command is used to modify the monitor options for a database once opened. The RD (Remove Database) command is used to remove a database. The GD (Get Database) command is used to display general information about a database and the status of its monitor options. The DD (Display Database) command is used to display a list of all databases and how they share user files. These commands are explained below.

3.10.1 ADDING A DATABASE (AD)

Anyone with level 55 access (database administrator) to any database may add another database with the AD (Add Database) command. The add operation involves assigning a name to the database as well as specifying the database monitor options. It also involves the specification of whether the file of users is to be shared with an existing database. Below is an example showing its use:

```
COMMAND?ad
DATABASE NAME?example
FIXED KEYWORD LIST (Y/N)?y
All keywords used must be in the thesaurus.
PRIVATE DATABASE (Y/N)?n
Public database: open to all EIES members.
DEFAULT LEVEL OF ACCESS (#)?10
Default level of access: 10
```

PEN/ANON COMMENTS ALLOWED (Y/N)?y
 Pen and anonymous comments allowed.
 TRACE USER ACTIVITY (Y/N)?y
 System will trace user activity.
 PROMPTED MODE DURING RESOURCE ENTRY (Y/N)?y
 System will prompt entry of resources.
 NUMBER OF DAYS BEFORE AUTOMATIC UPDATE (#)?0
 Wait of 0 day(s) before automatic update of database.
 DISPLAY EMPTY RESOURCE PARTS (Y/N)?y
 System will display empty resource parts.

SHARE USER FILE (Y/N)?n
 ENTER INITIAL USERS AND ACCESS LEVELS, END WITH +
 USER TO BE ADDED (#)?653
 MILES MARTIN (MILES,653)
 ACCESS LEVEL FOR MILES MARTIN (#)?50
 Access level: 50
 OK TO ADD MILES MARTIN TO EXAMPLE (Y/N)?y
 MILES MARTIN added to EXAMPLE.
 USER TO BE ADDED (#)?654
 NEIGH. INFO. SHARING EX. (NISE,654)
 ACCESS LEVEL FOR NEIGH. INFO. SHARING EX. (#)?50
 Access level: 50
 OK TO ADD NEIGH. INFO. SHARING EX. TO EXAMPLE (Y/N)?y
 NEIGH. INFO. SHARING EX. added to EXAMPLE.
 USER TO BE ADDED (#)?+

OK TO ADD EXAMPLE DATABASE (Y/N)?y

EXAMPLE database added.

COMMAND?

NOTE 1: the AD command feeds back to the user a brief summary of each option as it is set during the interaction. Note also that it asks for a DEFAULT ACCESS LEVEL for public databases. When entering the list of initial users, care should be taken to be sure that at least one user has at least level 50 access so that the FORMAT can be added prior to data entry.

NOTE 2: If the administrator indicates that the database is to share a file of users, the system will ask for the name of that DATABASE. In that case, the questions shown above which ask for initial members are not asked.

NOTE 3: the database administrator should be very careful when choosing to share user files between databases, since once this is done, it cannot be undone without copying the database into a completely new database without such sharing.

NOTE 4: if no users are entered and the database does not share a user file with another database then the person adding the database is automatically added as its initial user.

3.10.2 MODIFYING DATABASE MONITOR OPTIONS (MD)

The MD (Modify Database) command may be used to modify the monitor options for a database. Whereas level 55 access is required to ADD a database, anyone with level 50 access can modify one. Below is an example of an interaction with this command:

COMMAND?md

Press RETURN to leave option as is:

FIXED KEYWORD LIST (Y/N)?

PRIVATE DATABASE (Y/N)?

PEN/ANON COMMENTS ALLOWED (Y/N)?n

Pen and anonymous comments prohibited.

TRACE USER ACTIVITY (Y/N)?

PROMPTED MODE DURING RESOURCE ENTRY (Y/N)?

NUMBER OF DAYS BEFORE AUTOMATIC UPDATE (#)?1

Wait of 1 day(s) before automatic update of database.

DISPLAY EMPTY RESOURCE PARTS (Y/N)?

MODIFY DATABASE NAME (Y/N)?n

OK TO MODIFY EXAMPLE DATABASE (Y/N)?y

EXAMPLE database modified.

COMMAND?

NOTE 1: the system provides feedback on only those options which are changed.

NOTE 2: The moderator can change the name of the database here if necessary.

3.10.3 REMOVING DATABASES (RD)

A database may be removed by any user with database administrator access (level 55) through use of the RD (Remove Database) command. When entered, it asks OK TO REMOVE DATABASE (Y/N)? as a double check. It should be obvious that this command can only be used from within a database other than the one being removed. A database that has other database relationships tied to it cannot be deleted until all other relations associated with it are removed.

3.10.4 GETTING DATABASE OPTIONS (GD)

The GD (Get Database) command may be used by anyone to get the current status of options in the database as well as whether it shares a user file with another database. It also gets the total number of comments and resources in the database. Below is an example:

COMMAND?gd

As of 6/19/80 12:59 PM the EXAMPLE database contains 329 resources and 100 comments.

Option settings:

 All keywords used must be in the thesaurus.
 Public database: open to all members of EIES
 Default access level: 10
 Pen and anonymous comments allowed.
 System will trace user activity.
 System will prompt entry of resources.
 Wait of 0 day(s) before automatic update of database.

File of users not shared with any other database.

3.10.5 DISPLAYING DATABASE RELATIONS (DD)

The DD (Display Database) command can only be used by a database administrator with level 55 access. This command is the only way to get a complete listing of all databases in the RESOURCES system. For each database, it shows the total number of resources, the total number of comments, and the database with which it shares a user file, as shown in the example below:

COMMAND?dd

DATABASES IN THE RESOURCES SYSTEM AS OF 12/12/80 12:12 PM

DATABASE NAME	RESOURCES	COMMENTS	USER FILE
-----	-----	-----	-----
EXAMPLE	329	100	EXAMPLE
ORGANIZATIONS	1414	529	ORGANIZATIONS
PEOPLE	729	302	ORGANIZATIONS
BOOKS	771	53	ORGANIZATIONS

3.11 EIES SYSTEM COMMANDS (+RESOURCES, +RCY, +RCIN)

In addition to the commands that operate WITHIN the RESOURCES system, there are three EIES system-wide commands that connect EIES with RESOURCES and vice versa.

First the obvious +RESOURCES command is used to go from EIES to the RESOURCES system. ++ is used to go the other way. +RESOURCES begins by asking for the DATABASE? If the user enters either a full or abbreviated database name of an existing database to which he or she has access, the system will then respond with COMMAND? or a menu if the user is in EIES long-menu mode (see Section 3). Abbreviations must specify a unique database in order to work.

The other two commands are used for copying resources and comments out of the RESOURCES system into the EIES scratchpad to be sent as a message, added as a conference comment, or whatever. They are the +RCY and +RCIN commands. The difference between the two is that RCY copies the resource or comment into the scratchpad in the standard OUTPUT FORMAT for the RESOURCE system, including the SYSTEM ID line. On the other hand, the RCIN command copies the resource or comment into the scratchpad in the standard resource INPUT FORMAT with no headers and with all indirect edits intact. See Section 3.1.1 for the details of these standard formats. RCY stands for Resource Copy. RCIN stands for Resource Copy IN.

Both commands begin by asking for the DATABASE?. In response to the DATABASE? question, the user may enter either a full or abbreviated database name as long as the abbreviation identifies a single database uniquely. Then, if the database exists, and if the user has access to the database, these commands continue by asking ITEM? In response to the ITEM? question the user may enter either a PART specification that uniquely defines a resource within the particular database, or a number. If the user enters a number, the system assumes that it is a COMMENT number and copies the comment into the scratchpad. Otherwise it copies the uniquely defined resource into the scratchpad. The user must be in the scratchpad for either of these commands to work.

Both of these commands increment the count of resources and/or comments gotten by the user and are included in activity traces if that option has been turned on.

3.11.1 RESOURCE EDITOR MODE COMMANDS (+DIRECT,+PROMPTED)

The database monitor specifies via a monitor option whether or not the RESOURCE EDITOR begins in DIRECT or PROMPTED mode when resources are being added with the AR command. If a user wishes to change from this default mode during entry of a resource, s/he

should use either the +DIRECT or +PROMPTED command to change modes. If the user is using the RESOURCE EDITOR when the command is used, the mode of the editor changes immediately. Furthermore, once entered, these commands change the mode the editor uses when a user begins entering a new resource. This change becomes the new default mode for the user until s/he changes it again with either command.

3.11.2 SWITCH 14 FOR TOP-OF-PAGE CONTROL (+SSW14)

Any command which prints out one or more resources or comments tests the status of EIES switch 14. If the switch is ON then the command goes to the top of the next page BEFORE printing the item or list of items. The command +SSW14 will turn on this feature and the command +RSW14 will turn it off.

3.11.3 SWITCH 6 FOR RECEIVING YOUR OWN ITEMS (+SSW6)

The PC command which delivers waiting comments tests the status of EIES switch 6. If the switch is ON then comments authored by the user are delivered, otherwise delivery of the user's items is suppressed. This feature also applies to the delivery of new resources using the FP command.

3.11.4 SWITCH 20 FOR MICROCOMPUTER HANDSHAKE (+SSW20)

In order for any of the microcomputer commands (ER, XF, XK, XP, and XR) to work properly, the +SSW20 command must first be used in EIES to turn on switch 20. This enables the DC1 (control Q) handshaking protocol for EIES prompts needed by the microcomputer to assure accurate interfacing to EIES. Once turned on this switch can remain on even though the user is not working through a microcomputer.

3.12 BACKGROUND COMMAND PROCESSING (BG)

The RESOURCES system offers built-in background options for several commands. It also allows the user to set up and process a task while doing something else, or while not online at all. The results are delivered to the user later.

Most of the time-consuming processing done by the RESOURCES system is done by background tasks. Below is a summary of commands that use background tasks or give the user the option of having

processing done by background task.

COMMAND	PURPOSE OF TASK
ER	adding of resources to database after entry in EIES
FP	finding Boolean combinations of lists of resources
MA	removing old entries from activity trace files
SR	sorting lists of resources
TD	reporting database statistics
TR	tabulating frequency of use of parts of resources
TU	reporting user statistics
UD	updating inverted file indices for database

All of these commands use the standard background task prompts and messages specified in Section 4.2, unless otherwise noted.

To avoid problems in access to system files during the execution of such tasks and to assure that any entries into user activity trace files are entered into the file of the user invoking the task, all such tasks should run under the privileges of the invoker of the task rather than under the privileges of the task ID.

To further facilitate background processing by users, any time a background task enters the RESOURCES system, the system should check to see if it is running under a system task ID (140 through 149) and if so, use the ID of the INVOKER rather than the ID of the task itself for all processing done while in the RESOURCES system.

Also, the background command (BG) can be used to process a series of RESOURCES commands using EIES background tasks. The command will ask for the command to be processed background and then the successive answer-aheads needed to complete the task. The user must know which questions the system is going to ask so the answer-aheads can be entered in the proper order. Data entry is stopped by entering *END*. Since all traps are valid answer-aheads, this convention is used.

```

COMMAND?bg
COMMAND TO PROCESS BACKGROUND?tr
ENTER ANSWER-AHEADS IN ORDER WHICH THEY ARE TO BE PROCESSED
-- END WITH *END*
?keyword=...
?
.
.
.*end*

OK TO PROCESS ABOVE COMMANDS (Y/N)?y
Your request is being processed.
The results will be sent to you as an EIES message.

COMMAND?

```

Using the answer-ahead conventions at the =? prompt, more than just one RESOURCES command can be processed at a time. One limitation to this command is that it does not check the validity of all input except the first question COMMAND TO PROCESS?. The user should be familiar with the command before using BG so it can be used properly. Errors in the commands will not do any harm to the system.

The background task is started later the night the user enters the commands, and the background message will be sent to the user who invoked it the next day.

The system will handle the command Find Pattern with an added option. If at the prompt COMMAND TO PROCESS BACKGROUND? the user enters FP (find pattern), the system will respond by asking INTERVAL BETWEEN TASKS (# of days)?. What this means is that for the FP command, the system will automatically use the specified pattern at regular intervals specified by the user. If, for instance, a person wanted a certain pattern to be checked every month to see if any resources have been added to the database that fit that particular pattern, the response to the INTERVAL prompt would be 30. The user only has to start the process once and the system will run it at the specified interval without any user intervention. A response of 0 at the INTERVAL question will start the pattern search that night, but not reschedule it again. See section 3.5.10 for more information on patterns.

4.0 COMMAND SUMMARY

Below is an example of the menu printed when a user presses RETURN at COMMAND? in the RESOURCES system:

Choose a verb from the first column and a noun from the second column and make a command by combining the first letter (or two, where shown) of each.

Verbs -----	Nouns -----
A (Add)	A (Activity trace)
D (Display)	C (Comment)
E (Enter)	D (Database)
F (Find)	F (Format)
G (Get)	K (Keywords)
M (Modify)	L (Layout)
P (Print waiting)	M (Marker)
R (Remove)	P (Pattern)
S (Sort)	R (Resource)
T (Tabulate)	T (Tally)
U (Update)	TE (TEmplate)
X (Xmit)	U (User)

Enter ++ to return to EIES.

Enter ? to get a list of operational commands.

Below is a complete list of commands. A version of this list (without access levels) is printed out when a user enters a question mark (?) in response to the COMMAND? prompt. That list contains only those commands that the user has the access to use.

Command -----	Access -----	Description -----
AC	10	Add comment to feedback file for resource
AD	55	Add database
AF	50	Add format to database
AK	30	Add keyword to thesaurus
AL	0	Add layout
AP	10	Add pattern
AR	20	Add resource to database
AT	10	Add response to feedback tally
AU	40	Add user to database
BG	0	Queue up a series of commands for background processing
CM	0	Compose a message in the scratchpad
CTR	20	Cross tabulate resources by parts

DC	0	Display comment in feedback file for resource
DD	55	Display list of databases and relationships
DK	0	Display compressed list of keywords in thesaurus
DP	0	Display patterns
DR	0	Display resource
DT	0	Display summary of tally responses for resource
DU	0	Display list of users
ER	20	Enter resource from microcomputer in MICRO mode
FP	0	Find resources by patterns
FR	0	Find resources by Boolean combinations of parts
GA	40	Get activity trace on specified user
GC	0	Get comment in feedback file for resource
GD	0	Get counts of items and display of options for database
GF	0	Get format specification for database
GL	0	Get layout(s)
GR	0	Get resource by parts
GT	0	Get current results of tally for resource
GTE	0	Get output template
GU	0	Get detailed information on specified users
MA	40	Modify activity traces for specified users
MC	10	Modify comment in feedback file
MD	50	Modify database parameters and options
MF	50	Modify format for database
MK	30	Modify thesaurus and keyword relationships
ML	0	Modify layout
MM	0	Modify markers in comment files
MP	0	Modify pattern markers
MR	20	Modify resource in database
MT	50	Modify (or add) tally questions for database
MTE	0	Modify output template
MU	40	Modify user access levels
PC	0	Print waiting comments in feedback files
PR	0	Print waiting resources in database
RC	10	Remove comment from feedback file
RD	55	Remove database
RK	30	Remove keyword from thesaurus
RL	0	Remove layout
RP	0	Remove pattern
RR	20	Remove resource from database
RU	40	Remove user from database
SR	20	Sort and print resources
TD	20	Tabulate and report on database statistics
TR	20	Tabulate keywords, numbers, and types in resources
TU	20	Tabulate and report on user statistics

UD	20	Update database (invert resource files)
UR	20	Update resource in database
UL	0	Update layout
XF	30	Xmit format from EIES to microcomputer
XK	30	Xmit keyword (thesaurus) from EIES to microcomputer
XP	20	Xmit resources to microcomputer by patterns
XR	20	Xmit resources to microcomputer by Boolean

combinations

5.0 CAPACITIES AND LIMITATIONS

The system is able to handle databases with the following dimensions:

10,000 DATABASES

65,536 RESOURCES per DATABASE

65,536 COMMENTS per DATABASE

4096 USERS per DATABASE

1000 KEYWORDS per letter of the alphabet or per special character

256 related KEYWORDS per KEYWORD

4096 RESOURCES per KEYWORD

256 PARTS per RESOURCE

16 PARTS per PATTERN

4096 RESOURCES per FIND LIST

256 PATTERNS per USER

200 LINES per TEXT PART or COMMENT

16 TALLY QUESTIONS per TALLY

INDEX

OPERATOR\OR OPERATOR\ - OPERATOR\NOT OPERATOR\ -----	2.5.4
++ -----	3.1.2
+DIRECT -----	3.3.1, 3.3.1.9, 3.11.1
+PROMPTED -----	2.8, 3.3.1, 3.3.1.9, 3.11.1
+RCIN -----	3.3.3.3, 3.7.10, 3.11
+RCY -----	3.3.3.1, 3.7.10, 3.11
+RESOURCES -----	3.1, 3.11
+SSW14 -----	3.11.2
+SSW20 -----	3.3.2.1, 3.11.4
+SSW6 -----	3.11.3
... -----	2.5.1
:- -----	3.3.3.3, 3.5.10.8
:: -----	3.3.3.1, 3.5.2
? -----	3.1.1
?? -----	3.1.1
@[...] -----	3.5.11.5
ABBREVIATIONS -----	2.5.2
ABBREVIATIONS: RULE FOR -----	3.3.1.1
AC -----	3.7.1
ACCESS LEVEL -----	3.1.1
ACCESS LEVEL: MODIFYING -----	3.9.2
ACCESS LEVELS -----	3
ACCESS TO COMMENT AND TALLY FEEDBACK -----	3.5.10.6
ACCESS TO COMMENTS AND TALLY FEEDBACK -----	3.5.9.6
ACCESS -----	2.8
ACTION? -----	3.5.9
ACTIVITY TRACE BY COMMAND -----	3.1.4
ACTIVITY TRACE -----	2.7
AD -----	3.10.1
ADD PATTERN -----	2.5.5
ADDING A DATABASE -----	3.10.1
ADDING A FORMAT TO DATABASE -----	3.8.1
ADDING A KEYWORD TO THESAURUS -----	3.4.1
ADDING A RESOURCE -----	3.2.1
ADDING A USER -----	3.9.1
ADDING COMMENTS -----	3.7.1
ADDING LAYOUTS -----	3.5.11.5
ADDING PATTERNS -----	3.5.10.1
ADDING TALLY RESPONSES -----	3.7.11
ADMINISTRATOR -----	2.2
AF -----	3.8.1
AK -----	3.4.1
AL -----	3.5.11.5
ALGEBRA OF OPERATIONS -----	2
ANONYMOUS OR PEN COMMENTS -----	3.7.2
ANONYMOUS -----	2.8
ANSWER-AHEAD FOR NUMBERS -----	3.3.1.8
ANSWER-AHEAD FOR TYPES -----	3.3.1.8
ANSWER-AHEAD -----	3.1.3, 3.3.1
AP -----	2.5.5, 3.5.10.1
AR -----	3.2.1

ASCII -----	3.3.2.2
AT -----	3.7.11
AU -----	3.9.1
BACKGROUND COMMAND PROCESSING -----	3.12
BACKGROUND TASK: ER -----	3.3.2.8
BACKGROUND TASKS -----	3.12
BASIC SYSTEM CONCEPTS -----	2
BG -----	3.12
BILLING -----	2
BOOLEAN COMBINATIONS OF PARTS -----	3.5.9.1
BOOLEAN OPERATORS -----	2.5.4, 3.5.10.1
BROADER -----	3.4.3
CHECKING AND FORMATTING IN MICRO MODE -----	3.3.2.3
CHECKING FOR UNIQUENESS OF ID PARTS IN MICRO MODE -----	3.3.2.7
CHECKING PARITY -----	3.3.2.2, 3.3.2.3
CHECKS PERFORMED BY RESOURCE EDITOR -----	3.3.1.5
CLEARINGHOUSE -----	1
COMBINATIONS OF PARTS: BOOLEAN -----	3.5.9.1
COMMA NOT PREFERRED -----	3.1.3
COMMAND ACCESS -----	4
COMMAND INTERFACE: RESOURCES -----	3
COMMAND MENU -----	3.1.1
COMMAND NOUNS -----	3.1.1
COMMAND PROCESSING BY BACKGROUND TASKS -----	3.12
COMMAND VERBS -----	3.1.1
COMMAND -----	4
COMMANDS: ACCESS LEVELS OF -----	3
COMMANDS: DATA ENTRY MODIFICATION -----	3.2
COMMANDS: EIES SYSTEM -----	3.11
COMMANDS: FEEDBACK -----	3.7
COMMANDS: FORMAT MANAGEMENT -----	3.8
COMMANDS: GET FIND RESOURCE -----	3.5
COMMANDS: PATTERN -----	3.5.10
COMMANDS: REPORT-WRITING -----	3.6
COMMANDS: RESOURCE EDITOR MODE -----	3.11.1
COMMANDS: RESOURCE EDITOR -----	3.3.1
COMMANDS: RESOURCES SYSTEM MANAGEMENT -----	3.10
COMMANDS: RESOURCES USER INTERFACE -----	3
COMMANDS: THESAURUS MANAGEMENT -----	3.4
COMMANDS: USER MANAGEMENT -----	3.9
COMMENT-ONLY -----	2.2
COMMENTS -----	2.6
COMMENTS: ACCESS TO -----	3.5.9.6, 3.5.10.6
COMMENTS: ADDING -----	3.7.1
COMMENTS: DELIVERY OF WAITING ITEMS -----	3.7.8
COMMENTS: DISPLAYING HEADERS -----	3.7.7
COMMENTS: GETTING -----	3.7.6
COMMENTS: MODIFYING -----	3.7.4
COMMENTS: PEN OR ANONYMOUS -----	3.7.2
COMMENTS: REMOVING -----	3.7.5
COMMENTS: TRACE OF -----	3.7.3
COMMUNICATIONS PROTOCOLS -----	3.11.4
COMPARISON OF GR AND FR -----	3.5.9.8
CONTROL-Q -----	3.3.2.1
COPYING AS INPUT -----	3.7.10
COPYING AS OUTPUT -----	3.7.10
COPYING INTO SCRATCHPAD -----	3.7.10
COSTS -----	1

CROSS TABULATE RESOURCES BY PARTS -----	3.6.5
CTR -----	3.6.5
DATA ENTRY MODIFICATION COMMANDS -----	3.2
DATABASE ABBREVIATIONS -----	2.4.11
DATABASE FORMAT AND OUTPUT FEATURES -----	3.5.11
DATABASE FORMAT MANAGEMENT COMMANDS -----	3.8
DATABASE FORMAT -----	2.1
DATABASE MONITOR OPTIONS: MODIFYING -----	3.10.2
DATABASE OPTIONS: GETTING -----	3.10.4
DATABASE: ADDING A -----	3.10.1
DATABASE: EXPANDING OR ADDING TO FORMAT OF -----	3.8.1
DATABASE: GETTING FORMAT OF -----	3.8.4
DATABASE: MODIFYING FORMAT OF -----	3.8.2
DATABASE: REMOVING -----	3.10.3
DATABASE: TABULATING STATISTICS FOR -----	3.6.3
DATABASES -----	2.1
DATABASES: DISPLAYING -----	3.10.5
DATE PARTS -----	2.4.2, 2.5.2
DATE -----	2.4.2
DC -----	3.7.7
DC1 -----	3.3.2.1
DD -----	3.10.5
DECENTRALIZATION -----	1
DEFAULT ACCESS -----	2.8
DEFAULT ENTRIES IN PROMPTED MODE -----	3.3.1.11
DEFAULT TEMPLATE -----	2.5.6
DELIVERY OF WAITING COMMENTS -----	3.7.8
DELIVERY OF WAITING ITEMS AND SWITCH 6 -----	3.11.3
DELIVERY OF WAITING RESOURCES -----	3.5.1
DESCRIPTORS -----	2.4.6
DIMENSIONS -----	5
DIRECT MODE -----	3.3.1.3
DIRECT RESOURCE ENTRY -----	3.3.1.9
DISPLAY COMMENT HEADER -----	3.7.7
DISPLAY PARTS AND OUTPUT FEATURES -----	3.5.11
DISPLAY PARTS -----	3.5.3, 3.5.10.9
DISPLAY RESOURCES -----	3.5.3
DISPLAY USERS -----	3.9.4
DISPLAYING DATABASE RELATIONS -----	3.10.5
DISPLAYING KEYWORDS IN THESAURUS -----	3.4.4
DISPLAYING PATTERNS -----	3.5.10.4
DISPLAYING TALLIES -----	3.7.12
DK -----	3.4.4
DP -----	3.5.10.4
DR -----	3.5.3
DT -----	3.7.12
DU -----	3.9.4
EDITOR -----	2.2
EDITOR: RESOURCE -----	3.3
EIES SYSTEM COMMANDS -----	3.11
EIES -----	3.1
EIES: RETURNING TO -----	3.1.2
END OF RESOURCE SIGNAL -----	3.3.2.4
END OF TRANSMISSION SIGNAL -----	3.3.2.5
ENTRIES IN MICRO MODE: TRACE OF -----	3.3.2.8
ER BACKGROUND TASK -----	3.3.2.8
ER -----	3.3.2, 3.3.2.8
EXAMPLE OF MICRO MODE INTERACTION -----	3.3.2.6

EXAMPLE OF RESOURCE EDITOR -----	3.3.1.4
EXAMPLE: SUPPORT FOR GETTING RESOURCES -----	3.5.7
EXPLANATION -----	3.1.1
FEEDBACK COMMANDS -----	3.7
FEEDBACK FILE: ADDING COMMENTS TO -----	3.7.1
FEEDBACK MESSAGES: GET AND FIND -----	3.3.3.6
FEEDBACK -----	2, 2.6
FEEDBACK: ACCESS TO -----	3.5.9.6
FILE INVERSION: UPDATE DATABASE -----	3.2.4
FIND RESOURCES -----	2.5, 2.5.4
FIND-BY-PATTERN -----	2.5.4, 2.5.5
FIND -----	2.5.4
FINDING BY PATTERNS -----	3.5.10.5
FINDING RESOURCES WITH NON-KEYWORD PARTS -----	3.5.9.3
FINDING RESOURCES -----	3.5.9
FINDING RESOURCES: SUPPORT FOR -----	3.5.9.5
FINDS BY PATTERN: TRACE OF -----	3.5.10.7
FINDS: OUTPUT IN INPUT FORMAT -----	3.5.11.4
FINDS: SUPPORT FOR MULTIPLE KEYWORD -----	3.5.9.2
FINDS: TRACE OF -----	3.5.9.7
FLOW CONTROL -----	3.11.4
FORMAT MANAGEMENT COMMANDS: DATABASE -----	3.8
FORMAT OF DATABASE: GETTING -----	3.8.4
FORMAT -----	2.4, 2.4.3, 2.4.4, 2.4.7, 2.4.8, 2.4.8
FORMAT: DATABASE -----	2.1, 2.4.11, 3.8.1, 3.8.2
FORMAT: XMIT -----	3.8.5
FORMS -----	2.5.8
FP AND SWITCH 6 -----	3.11.3
FP -----	2.5.4, 3.3.3.5, 3.3.3.6, 3.5.10.5
FR COMPARED WITH GR -----	3.5.9.8
FR -----	2.5, 2.5.4, 3.3.3.5, 3.3.3.6, 3.5.9
FREE KEYWORDS OPTION -----	2.4.6
GA -----	3.1.4, 3.9.6
GATEKEEPER -----	2.2
GC -----	3.7.6
GD -----	3.10.4
GET FIND FEEDBACK MESSAGES AND OUTPUT -----	3.3.3.6
GET FIND RESOURCE COMMANDS -----	3.5
GET KEYWORDS -----	3.4.6
GET LAYOUT -----	3.5.11.8
GET RESOURCES -----	2.5
GETS: OUTPUT IN INPUT FORMAT -----	3.5.11.4
GETS: TRACE OF -----	3.5.8
GETTING COMMENTS -----	3.7.6
GETTING DATABASE OPTIONS -----	3.10.4
GETTING FORMAT OF DATABASE -----	3.8.4
GETTING KEYWORD RELATIONSHIPS -----	3.4.6
GETTING LAYOUTS ON FILE -----	3.5.11.8
GETTING RESOURCES BY A SINGLE PART -----	3.5.2
GETTING RESOURCES: SUPPORT FOR -----	3.5.7
GETTING TALLY QUESTIONS AND RATINGS -----	3.7.13
GETTING THE TEMPLATE -----	3.5.11.2
GETTING USER INFORMATION -----	3.9.5
GETTING USERS ACTIVITY TRACE -----	3.9.6
GF -----	3.8.4
GK -----	3.4.6
GL -----	3.5.11.8
GR COMPARED WITH FR -----	3.5.9.8

GR	-----	2.5, 3.2.3, 3.3.3.5, 3.3.3.6, 3.5.2, 3.5.8
GT	-----	3.7.13
GTE	-----	3.5.11.2
GU	-----	3.9.5
HANDSHAKE:	MICROCOMPUTER	----- 3.11.4
HANDSHAKING:	MICRO MODE	----- 3.3.2.1
HEADER LINE	-----	2.4.1
HEADER	-----	2.4.1, 3.3.3.2
HELP	-----	3.1.1
ID LINE FORMAT:	STANDARD SYSTEM	----- 3.3.3.2
ID ORDER:	INVERSE SYSTEM	----- 3.5.9
ID PARTS	-----	2.4.3, 2.5.2
ID PARTS:	MICRO MODE CHECKING	----- 3.3.2.7
ID	-----	2.4.3
IMPLEMENTOR	-----	2.2
INDEX	-----	3.2.4
INDEXED PARTS	-----	2.4.12, 3.2.4, 3.5.2, 3.5.9
INDEXING OF NEW ENTRIES	-----	2.5
INDEXING	-----	2.4.12
INPUT AND OUTPUT FORMATS:	RESOURCE EDITOR	----- 3.3.3
INPUT FORMAT IN OUTPUT	-----	3.3.3.5
INPUT FORMAT	-----	3.3.3.3
INTERACT	-----	1
INVERSE SYSTEM	ID ORDER	----- 3.5.9
INVERTED FILES	-----	2.4.12
INVERTED INDEX	-----	2.4.12
KEYS	-----	2.4.6
KEYWORD CONTROL	-----	3.4.3
KEYWORD FINDS:	SUPPORT FOR MULTIPLE	----- 3.5.9.2
KEYWORD PARTS	-----	2.4.6
KEYWORD RELATIONSHIPS	-----	2.3
KEYWORD RELATIONSHIPS:	MODIFYING	----- 3.4.3
KEYWORD:	ADDING TO THESAURUS	----- 3.4.1
KEYWORD:	MUST BEGIN WITH LETTER	----- 3.4.1
KEYWORD:	REMOVING FROM THESAURUS	----- 3.4.2
KEYWORDS	IN THE RESOURCE EDITOR	----- 3.3.1
KEYWORDS	-----	2.1, 2.3, 2.4.6, 2.5.2
KEYWORDS:	DISPLAYING	----- 3.4.4
KEYWORDS:	FINDS ON PARTS THAT ARE NOT	----- 3.5.9.3
KEYWORDS:	GETTING	----- 3.4.6
KEYWORDS:	NUMBERS FOR	----- 3.3.2.3, 3.4.5
KEYWORDS:	SUPPORT FOR ENTRY OF	----- 3.3.1.2
KINDS OF PARTS	-----	2.4
LABEL PARTS	-----	2.4.9, 2.5.2
LABELS	-----	2.4.9
LAYOUT INSTEAD OF TEMPLATE	-----	3.5.11.5
LAYOUT NAME	-----	3.5.11.5
LAYOUT:	GLOBAL OR LOCAL	----- 3.5.11.5
LAYOUT:	UPDATE	----- 3.5.11.9
LAYOUTS ON FILE:	GETTING	----- 3.5.11.8
LAYOUTS	-----	2.5.8
LAYOUTS:	ADDING	----- 3.5.11.5
LAYOUTS:	MODIFYING	----- 3.5.11.6
LAYOUTS:	RELATION TO TEMPLATES	----- 2.5.6
LAYOUTS:	REMOVING	----- 3.5.11.7
LEVEL OF ACCESS	-----	2, 3.1.1
LEVELS OF ACCESS	-----	2.2
LIST OF COMMANDS	-----	4

LISTING THESAURUS FOR MICRO MODE -----	3.4.5
LISTS AND RANGES -----	2.5.1, 3.5.6
LISTS -----	2.5.1
MA -----	3.9.7
MARKER: PATTERN -----	3.5.10
MARKERS -----	2.6
MARKERS: MODIFYING PATTERN -----	3.5.10.2
MARKERS: MODIFYING -----	3.7.9
MC -----	3.7.4
MD -----	3.10.2
MENU -----	4
MF -----	3.8.2
MICRO MODE CHECKING FOR UNIQUENESS OF ID PARTS -----	3.3.2.7
MICRO MODE FOR THE RESOURCE EDITOR -----	3.3.2
MICRO MODE HANDSHAKING -----	3.3.2.1
MICRO MODE -----	3.3.2.8
MICRO MODE: CHECKING AND FORMATTING IN -----	3.3.2.3
MICRO MODE: EXAMPLE OF INTERACTION -----	3.3.2.6
MICRO MODE: LISTING THESAURUS FOR -----	3.4.5
MICROCOMPUTER HANDSHAKING -----	3.11.4
MK -----	3.4.3
ML -----	3.5.11.6
MM -----	2.5.7, 3.7.9
MODIFY DATABASE FORMAT -----	3.8.2
MODIFY MARKERS -----	2.5.7
MODIFYING ACTIVITY TRACE FILES -----	3.9.7
MODIFYING AN EXISTING RESOURCE -----	3.2.2
MODIFYING AND UPDATING RESOURCES COMPARED -----	3.2.5
MODIFYING COMMENTS -----	3.7.4
MODIFYING DATABASE MONITOR OPTIONS -----	3.10.2
MODIFYING KEYWORD RELATIONSHIPS -----	3.4.3
MODIFYING LAYOUTS -----	3.5.11.6
MODIFYING MARKERS -----	3.7.9
MODIFYING OR ADDING QUESTIONS IN A TALLY -----	3.8.3
MODIFYING PATTERN MARKERS -----	3.5.10.2
MODIFYING RESOURCES DURING PROMPTED MODE -----	3.3.1.10
MODIFYING THE TEMPLATE -----	3.5.11.1
MODIFYING USER ACCESS LEVELS -----	3.9.2
MONITOR OPTIONS -----	2.8
MONITOR OPTIONS: MODIFYING -----	3.10.2
MONITOR -----	2, 2.2, 2.4.11
MP -----	3.5.10.2
MR -----	3.2.2, 3.3.1.4
MT -----	3.8.3
MTE -----	3.5.11.1
MU -----	3.9.2
NAMES: PART -----	2.4.11
NARROWER -----	3.4.3
NETWORK -----	2.4.3
NEW ENTRIES: INDEXING OF -----	2.5
NODE -----	2.4.3
NON-KEYWORD PARTS -----	3.5.9.3
NOTICE: TEMPLATE IN EFFECT -----	3.5.11.3
NULL STRINGS -----	3.3.2.3
NULL TEMPLATE -----	2.5.6
NUMBER PARTS -----	2.4.7, 2.5.2
NUMBERS FOR KEYWORDS -----	3.3.2.3, 3.4.5
NUMBERS -----	2.4.7

NUMBERS: ANSWER-AHEAD FOR -----	3.3.1.8
OPERAND -----	2, 3.1.1
OPERATION -----	2.4.10
OPERATIONS ON PARTS -----	2.4.10
OPERATIONS -----	2.1, 3.1.1
OPERATOR -----	2
OPTIONS -----	2.8
OPTIONS: GETTING -----	3.10.4
OUTPUT FEATURES -----	3.5.11
OUTPUT FORMAT -----	3.3.3.1, 3.5.2
OUTPUT FORMAT: SYSTEM ID SUPRESSED IN -----	3.3.3.1
OUTPUT FORMATTING -----	2.5.8
OUTPUT IN INPUT FORMAT -----	3.3.3.5, 3.5.11.4
OVERVIEW -----	1
PAGING -----	3.11.2
PARITY CHECK BYTES -----	3.3.2.6
PARITY CHECKING -----	3.3.2, 3.3.2.2
PART CONTENTS -----	2.5
PART NAME -----	2.5
PART NAMES -----	2.4.11
PART PARAMETERS -----	2.4.11
PART-SPECIFIC PROMPTS -----	2.5.3, 3.3.1.3, 3.5.5
PARTS -----	2.4, 2.4.3, 2.4.10, 2.5.2, 2.5.3
PARTS: BOOLEAN COMBINATIONS OF -----	3.5.9.1
PARTS: CONTENTS UNKNOWN -----	3.5.4
PARTS: CROSS TABULATION OF -----	3.6.5
PARTS: DISPLAY -----	3.5.10.9
PARTS: NON-KEYWORD AND FINDS -----	3.5.9.3
PARTS: SELECTIVE PRINTOUT OF -----	3.5.10.9
PARTS: SORTING RESOURCES BY -----	3.6.1
PARTS: TABULATING RESOURCES BY -----	3.6.2
PARTS: TEXT CANNOT BE INDEXED -----	2.4.12
PATTERN COMMANDS -----	3.5.10
PATTERN FINDS: OUTPUT IN INPUT FORMAT -----	3.5.10.8
PATTERN MARKER -----	2.5.5, 3.5.10
PATTERN MARKERS: MODIFYING -----	3.5.10.2
PATTERN: XMIT -----	3.5.10.8
PATTERNS -----	2.5.4, 2.5.5
PATTERNS: DISPLAYING -----	3.5.10.4
PATTERNS: FINDING BY -----	3.5.10.5
PATTERNS: REMOVING -----	3.5.10.3
PATTERNS: TRACE OF FINDS BY -----	3.5.10.7
PC AND SWITCH 6 -----	3.11.3
PC -----	3.7.8
PEN OR ANONYMOUS COMMENTS -----	3.7.2
PEN -----	2.8
PR -----	2.5.7, 3.5.1
PRIMARY ID -----	2.4.4, 2.7
PRINT COMMENTS -----	3.7.8
PRINT RESOURCES WAITING -----	2.5.7
PRINT RESOURCES -----	3.5.1
PRINTING OUT PARTS OF RESOURCES -----	3.5.10.9
PROJECT EVALUATION -----	2
PROMPTED FORMAT -----	3.3.3.4
PROMPTED MODE -----	3.3.1.2, 3.3.1.3, 3.3.3.4
PROMPTED MODE: DEFAULT ENTRIES IN -----	3.3.1.11
PROMPTED MODE: MODIFYING RESOURCES DURING -----	3.3.1.10
PROMPTED RESOURCE ENTRY -----	3.3.1.9

PROMPTED	-----	2.8
PROMPTS: PART-SPECIFIC	----- 3.3.1.3,	3.5.5
PUBLIC DATABASE	-----	2.8
RANGES AND LISTS	-----	3.5.6
RANGES	-----	2.5.1
RATING	-----	2.6
RC	-----	3.7.5
RD	-----	3.10.3
READ-ONLY	-----	2.2
RELATIONAL INFORMATION	-----	3.5.9.2
REMOVING A KEYWORD FROM THESAURUS	-----	3.4.2
REMOVING A RESOURCE	-----	3.2.3
REMOVING COMMENTS	-----	3.7.5
REMOVING DATABASE	-----	3.10.3
REMOVING LAYOUTS	-----	3.5.11.7
REMOVING PATTERNS	-----	3.5.10.3
REMOVING USERS FROM DATABASE	-----	3.9.3
REPORT-WRITING COMMANDS	-----	3.6
RESOURCE EDITOR COMMANDS	-----	3.3.1
RESOURCE EDITOR FORMATS FOR INPUT AND OUTPUT	-----	3.3.3
RESOURCE EDITOR MODE COMMANDS	-----	3.11.1
RESOURCE EDITOR OUTPUT FORMAT	-----	3.3.3.1
RESOURCE EDITOR	-----	3.2.1, 3.3
RESOURCE EDITOR: BACKING UP IN	-----	3.3.1.10
RESOURCE EDITOR: CHECKS PERFORMED BY	-----	3.3.1.5
RESOURCE EDITOR: EXAMPLE OF	-----	3.3.1.4
RESOURCE EDITOR: INTERRUPT PROTECTION FOR	-----	3.3
RESOURCE EDITOR: MICRO MODE FOR	-----	3.3.2
RESOURCE ENTRY: DIRECT	-----	3.3.1.9
RESOURCE ENTRY: PROMPTED	-----	3.3.1.9
RESOURCE MARKER	-----	2.5.7
RESOURCE OUTPUT FEATURES	-----	3.5.11
RESOURCE	-----	2
RESOURCE: ADDING A	-----	3.2.1
RESOURCES SYSTEM MANAGEMENT COMMANDS	-----	3.10
RESOURCES	-----	2, 2.1
RESOURCES: DELIVERY OF WAITING ITEMS	-----	3.5.1
RESOURCES: SORTING BY PART	-----	3.6.1
RESOURCES: TABULATING BY PART	-----	3.6.2
RESOURCES: UPDATING OF	-----	3.2.5
RESOURCES: XMIT	-----	3.5.11.4
RETURNING TO EIES	-----	3.1.2
RK	-----	3.4.2
RL	-----	3.5.11.7
RP	-----	3.5.10.3
RR	-----	3.2.3
RU	-----	3.9.3
RULE FOR ABBREVIATIONS	-----	3.3.1.1
SCRATCHPAD: COPYING INTO	-----	3.7.10
SEARCHES	-----	2.5.4
SEMI-COLONS	-----	3.1.3
SIGNAL: END OF RESOURCE	-----	3.3.2.4
SIGNAL: END OF TRANSMISSION	-----	3.3.2.5
SORTING RESOURCES BY PART	-----	3.6.1
SR	----- 3.3.3.5,	3.6.1
STANDARD FORMATS	-----	3.3.3.7
STANDARD RESOURCE INPUT FORMAT	-----	3.3.3.3
STATISTICS: TABULATING REPORTS ON DATABASE	-----	3.6.3

STATISTICS: TABULATING USER -----	3.6.4
STORED PATTERNS -----	2.5.5
SUB-DATABASE -----	2.4.3
SUBACCOUNTS -----	2.8
SUPPORT FOR ENTRY OF KEYWORDS -----	3.3.1.2
SUPPORT FOR FINDING RESOURCES -----	3.5.9.5
SUPPORT FOR GETTING RESOURCES -----	3.5.7
SUPPORT FOR MULTIPLE KEYWORD FINDS -----	3.5.9.2
SWITCH 14 -----	3.11.2
SWITCH 20 -----	3.11.4
SWITCH 6 -----	3.11.3
SYSTEM ACCESSES -----	2.4.1
SYSTEM ADATE -----	2.4.1
SYSTEM COMMENTS -----	2.4.1
SYSTEM DATE -----	2.4.1
SYSTEM ID -----	2.4.1, 3.2.1
SYSTEM MANAGEMENT COMMANDS -----	3.10
SYSTEM MDATE -----	2.4.1
SYSTEM PARTS -----	2.4, 2.4.1
SYSTEM SOURCE -----	2.4.1
SYSTEM TALLIES -----	2.4.1
TABULATE USER STATISTICS -----	3.6.4
TABULATING DATABASE STATISTICS -----	3.6.3
TABULATING ONE PART AGAINST ANOTHER -----	3.6.5
TABULATING RESOURCES BY PARTS -----	3.6.2
TALLIES -----	2.6
TALLIES: ADDING RESPONSES -----	3.7.11
TALLIES: DISPLAYING -----	3.7.12
TALLIES: GETTING QUESTIONS AND RATINGS -----	3.7.13
TALLIES: QUESTIONS FOR -----	2.6
TALLIES: SCALES FOR -----	2.6
TALLY QUESTIONS: ADDING OR MODIFYING -----	3.8.3
TALLY -----	2
TALLY: ACCESS TO -----	3.5.9.6, 3.5.10.6
TD -----	3.6.3
TEMPLATE IN EFFECT NOTICE -----	2.5.6, 3.5.11.3
TEMPLATE SETTING -----	3.5.11.1
TEMPLATE -----	2.5.6
TEMPLATE: GETTING THE -----	3.5.11.2
TEMPLATE: MODIFYING THE -----	3.5.11.1
TEMPLATE: NON-DEFAULT -----	3.5.11.3
TEMPLATE: PART PRINTOUT CONTROLLED BY -----	3.5.10.9
TEMPLATES -----	2.5.6
TEMPLATES: RELATION TO LAYOUTS -----	2.5.6
TEXT PARTS: CANNOT BE INDEXED -----	2.4.12
THESAURUS MANAGEMENT COMMANDS -----	3.4
THESAURUS -----	2.3, 2.4.6
THESAURUS: DISPLAYING KEYWORDS IN -----	3.4.4
THESAURUS: GETTING THE -----	3.4.6
THESAURUS: LISTING FOR MICRO MODE -----	3.4.5
THIS GUIDE -----	3.1.1
TOP-OF-PAGE ADVANCE -----	3.11.2
TR -----	3.6.2
TRACE BY COMMAND ACTIVITY -----	3.1.4
TRACE BY COMMAND -----	2.7
TRACE FILE OVERFLOW -----	2.7
TRACE OF COMMENTS ADDED -----	3.7.3
TRACE OF FINDS BY PATTERN -----	3.5.10.7

TRACE OF FINDS -----	3.5.9.7
TRACE OF GETS -----	3.5.8
TRACE OF MICRO MODE ENTRIES -----	3.3.2.8
TRACE OF USER ACTIVITY: MODIFYING -----	3.9.7
TRACE OPTION -----	2.7, 3.2.1
TRACE -----	2.7, 2.8
TRACES OF USERS: GETTING -----	3.9.6
TU -----	3.6.4
TYPE NAME -----	2.4.8
TYPE PARTS -----	2.4.8, 2.5.2
TYPE -----	2.4.8
TYPES: ANSWER-AHEAD FOR -----	3.3.1.8
UD -----	3.2.4
UL -----	3.5.11.9
UNINDEXED PARTS -----	2.4.12
UNKNOWN CONTENTS -----	2.4.7, 2.5.2, 3.5.4
UNKNOWN CONTENTS: SETTING -----	3.3.1
UNKNOWN ID -----	2.5.2
UPDATE DATABASE -----	2.5, 3.2.4
UPDATE LAYOUT -----	3.5.11.9
UPDATE MONITOR OPTION -----	3.2.4
UPDATE -----	2.8
UPDATING AND MODIFYING RESOURCES COMPARED -----	3.2.5
UPDATING RESOURCES -----	3.2.5, 3.5.2
UR -----	3.2.5
USAGE OF LAYOUT: UPDATING -----	3.5.11.9
USE FOR -----	3.4.3
USE INSTEAD -----	3.4.3
USE LAYOUT -----	3.5.11.9
USER INTERFACE -----	3
USER MANAGEMENT COMMANDS -----	3.9
USER STATISTICS: TABULATING -----	3.6.4
USER -----	2
USERS -----	2.2
USERS: ADDING -----	3.9.1
USERS: DISPLAY LIST OF -----	3.9.4
USERS: GETTING ACTIVITY TRACE OF -----	3.9.6
USERS: GETTING INFORMATION ON -----	3.9.5
USERS: MODIFYING ACCESS LEVELS OF -----	3.9.2
USERS: PRUNING TRACE FILES OF -----	3.9.7
USERS: REMOVING FROM DATABASE -----	3.9.3
XF -----	3.8.5
XK -----	3.3.2.3, 3.4.5
XMIT FORMAT -----	3.8.5
XMIT PATTERN -----	3.5.10.8
XMIT RESOURCES -----	3.5.11.4
XP -----	3.5.10.8
XR -----	3.5.11.4
\& OPERATOR\AND OPERATOR\ -----	2.5.4