

# Lawrence Berkeley Laboratory

UNIVERSITY OF CALIFORNIA, BERKELEY

## Information and Computing Sciences Division

RECEIVED  
LAWRENCE  
BERKELEY LABORATORY

OCT 21 1987

LIBRARY AND  
DOCUMENTS SECTION

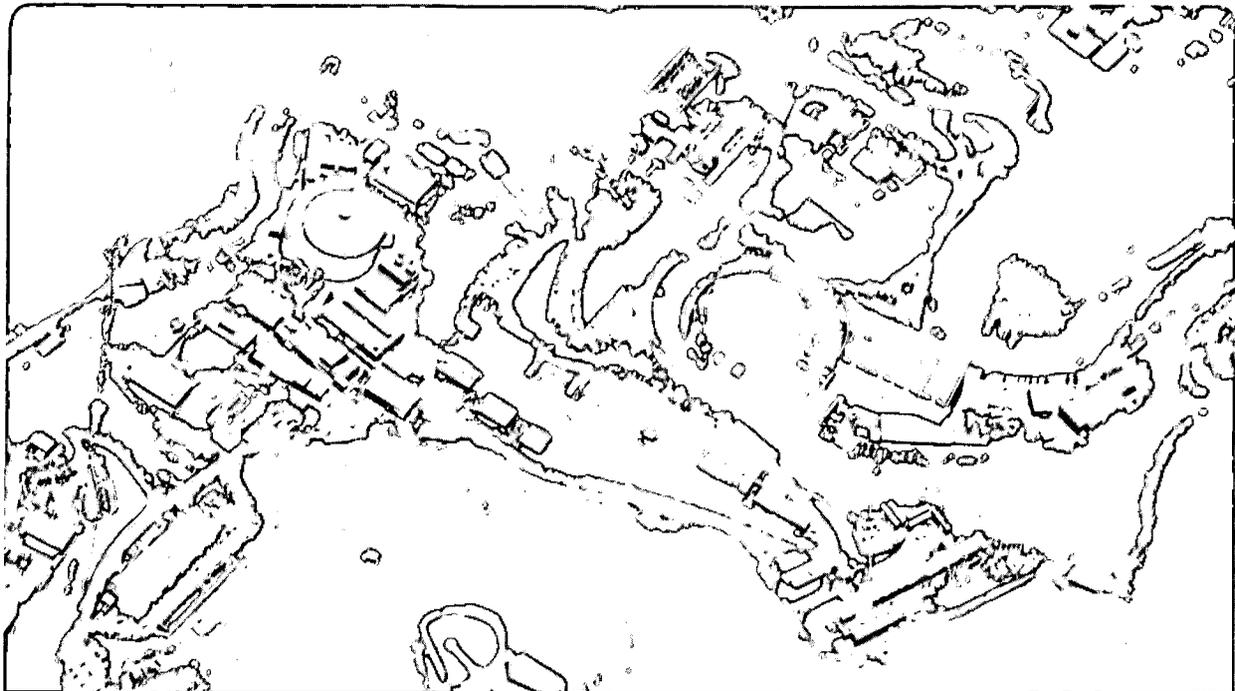
User's Guide for the ICS Database System

A. Konrad

August 1987

**For Reference**

**Not to be taken from this room**



PUB-3067  
c.1

## **DISCLAIMER**

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

PUB-3067

**User's Guide**  
**for the ICS Database System**

**Prepared for the U.S. Department of Energy under Contract DE-AC03-76SF00098**

**Allan Konrad, X5458**  
**Office of Computing Resources**  
**Computing Division**

**Version 1.1**  
**10 August 1987**

## Contents

I.	Introduction:	1
	Purpose	1
	CMS, SPIRES, the CMS/SPIRES interface	1
	General SPIRES Information	3
II.	Getting started	4
	Logging ON	4
	Logging OFF	5
III.	Organization of the database	6
IV.	Using the ICS subfile	9
	Description of elements in the ICS subfile	10
	Displaying records	12
	Searching in SPIRES; Searching the ICS subfile	12
	Updating records	13
	Adding new records	13
	Removing records	14
	Changing the key of a record	14
V.	Using the STATIONS subfile	15
	Description of elements in the STATIONS subfile	16
	Displaying records	18
	Searching the STATIONS subfile	19
	Updating records	20
	Adding new records	23
	Removing records	25
	Changing the key of a record	25
VI.	Using the KEY SYSTEM subfile	26
	Description of elements in the KEY SYSTEM subfile	26
	Displaying records	28
	Searching the KEY SYSTEM subfile	29
	Updating records	30
	Adding new records	32
	Removing records	33
	Changing the key of a record	33
VII.	Using the INSTRUMENTS subfile	34
	Description of elements in the INSTRUMENTS subfile	34
	Displaying records	36
	Searching the INSTRUMENTS subfile	37
	Updating records	38
	Adding new records	40
	Removing records	41
	Changing the key of a record	41

VIII.	Using the BLDG subfile	42
	Description of elements in the ICS subfile	42
	Displaying records	44
	Searching in SPIRES; Searching the ICS subfile	44
	Updating records	45
	Adding new records	47
	Removing records	48
	Changing the key of a record	48
IX.	Using the SERVICE subfile	49
	Description of elements in the SERVICE subfile	49
	Searching in the SERVICE subfile	49
	Adding, Updating, Deleting records	49
X.	Generating Reports	50
XI.	Printing; the LPR, LPRCC, and LABEL commands	51

APPENDIX A	Dip-switch settings for ADM3A
APPENDIX B	Using campus YTERM
APPENDIX C	Using campus KERMIT
APPENDIX D	Terminal control
APPENDIX E	INPUT FORMAT subcommands
APPENDIX F	Looking at your CMS files
APPENDIX G	Documentation
APPENDIX H	Human Help
APPENDIX I	Using Xedit

## I. Introduction:

- .1 Purpose
- .2 CMS, SPIRES, the CMS/SPIRES interface
- .3 General information about SPIRES.

### I.1 Purpose

The ICS database was implemented in support of the Integrated Communications System (ICS) Project. Additionally, it is linked to the LBLSTAFF database using the OCTOPUS, ICS, and SERVICE subfiles so that staff need not maintain employee information such as mailstop, payroll account number, termination date etc. This relieves the staff of tracking mailstop changes and other personal information.

### I.2 CMS, SPIRES, the CMS/SPIRES interface

The Stanford Public Information Retrieval System (SPIRES) is a product of Leland Stanford Junior University in Palo Alto, CA. The SPIRES database management system at LBL runs on the UC Berkeley Campus IBM 3090-200 under the VM/CMS operating system. VM SPIRES consists of three components:

- SPIRES itself (database management system)
- CMS (the operating system that manages the computer)
- SPIRES/CMS interface (maps SPIRES activity onto the CMS environment)

Figure 1 indicates how these components relate to one another. Normally, SPIRES users are not and need not be concerned with the subsystems between themselves and SPIRES. The diagram is provided only to demonstrate context.

Most of the icons are self-explanatory. The purpose of the SERIES/1 is to make the user's ASCII terminal appear as an IBM 3270 terminal to the IBM 3090, and to make the IBM 3090 appear to communicate in ASCII to the user.

Section II will describe the commands to move along the path from terminal through the gateways into SPIRES. This generally requires less than 10 seconds and becomes routine.

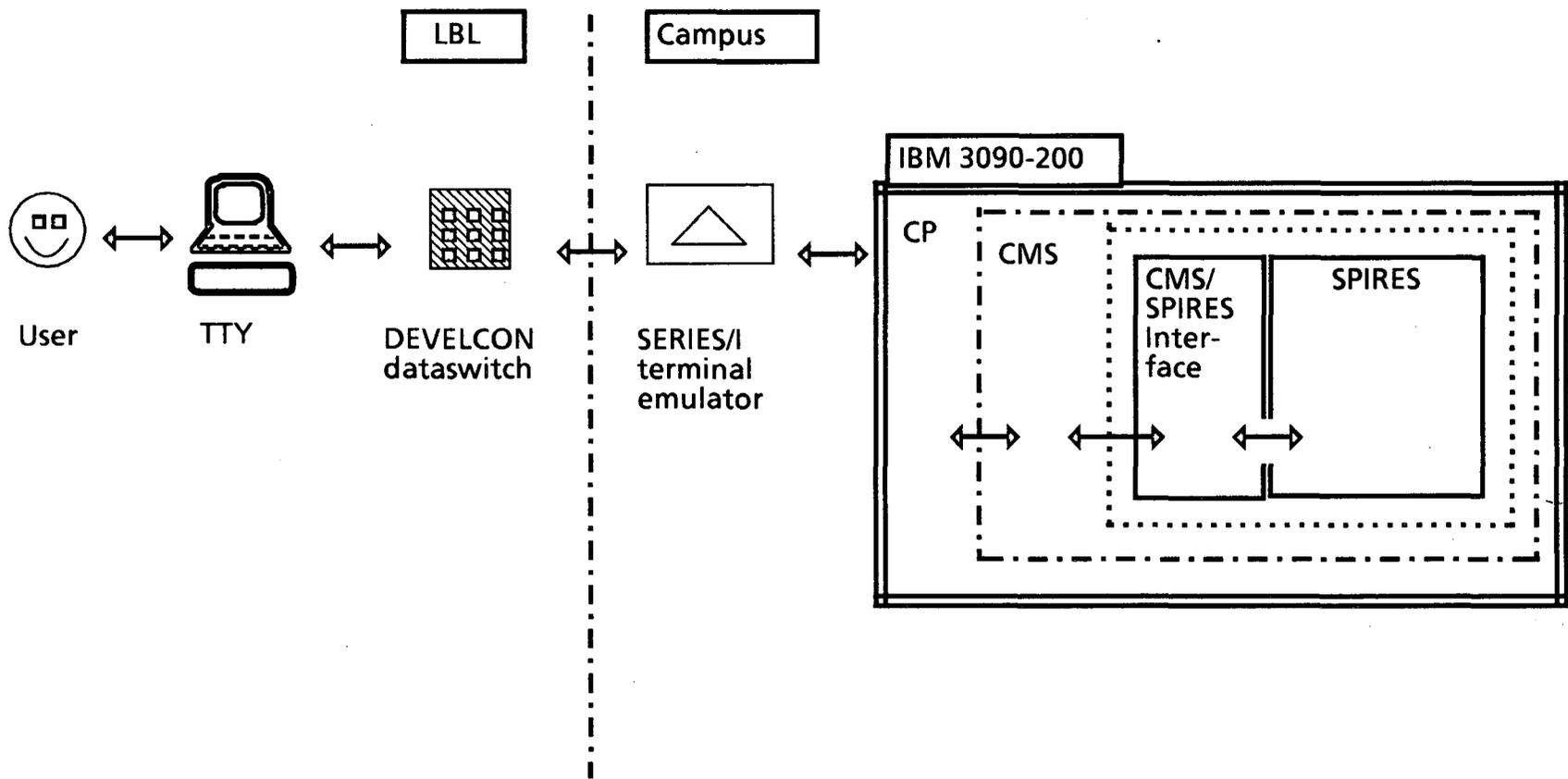


Figure 1. Pathway between user and SPIRES

### I.3 General Information about SPIRES.

Information for each ICS entity (extensions, buildings, instruments, key systems, stations, people) is stored directly into a SPIRES *record*. A record represents a single occurrence of that entity. Each record in a SPIRES database has a unique identifier referred to as the *key*. Further explanation of keys used in each of the ICS-related subfiles is described in the first chapter of the Section describing that subfile.

For each record, a particular element may be required or optional, singly or multiply occurring, have controlled allowable values, be limited to a particular type of value, and be indexed for ease in searching, etc.

If you are not in SPIRES, the CMS prompt is: **R;**

If you have EXITed SPIRES and you wish to re-enter, enter the command:

**SPIRES**

The normal SPIRES prompts are as follows:

- ? for UPPER case only
- > for upper and lower case
- +? UPPER case in Global For
- +> upper and lower case in Global For

All the modifications made to the database during the day (adds, updates, and removes) take effect immediately and are reflected the very next time the record is displayed or searched in an index.

For most SPIRES commands, only the first three characters need be entered. For example, the FIND command requires only FIN <index> <value>. In of this document, commands will be fully spelled out, with the first three letters capitalized; e.g., FINd, SHOW ACTive, indicating that only the capitalized characters need be entered. Use the SHOW INDEXes command to see what element can be searched using the FINd command. Use the BROWse <index name> [value] command to see what values are in the index.

The term *file* as used in "STATIONS file" or "LBLSTAFF file", is distinct from *physical* CMS files and refers to SPIRES files, which are "logical" files that are physically stored in CMS files. The "active file" is also a SPIRES concept, and usually refers to the CMS file ACTIVE FILE A. Any CMS filename can be used as the SPIRES active file and is specifiable by the user with the

SET Active <filename> <filetype> <filemode>

command. For example,

SET Active ACTIVE FILE A

## II. Getting Started.

- .1 Logging ON
- .2 Logging OFF

### II.1 Logging ON with and ADM3A terminal.

See APPENDIX B or C for special instructions if using a PC with YTERM or KERMIT.

1. Turn terminal on and make sure the blue TSB box displays either a green or red light.
2. If red light is illuminated, press the blue button and wait for green light.
3. When green light is illuminated, enter carriage return [CR].

The following dialogue should occur. The system response is in **bold**. The user response is in modern font.

4. **Request:** ccdb [CR].
5. System will respond with a bell, and cursor and will jump to next line. Enter carriage return [CR].
6. **YALE ASCII TERMINAL COMMUNICATIONS SYSTEM V2.1**  
enter terminal type: adm3a [CR]. (or appropriate terminal type)
7. System will respond with a pseudo-three-dimensional display CFO over the letters VM. Enter another [CR].
8. The screen will clear. Enter:  
L WILEY1 [CR].
9. **ENTER PASSWORD:**  
enter your password. It is not a good idea to write your password in this set of instructions. If you write it down, do so elsewhere.

Note: If your previous session ended "abnormally", e.g., by simply pushing the blue button on the TSB box to obtain a red light, you will have to enter, at this point in the logon procedure, the command: IPL CMS and then a [CR]. This should always be done when a paragraph beginning with the word "RECONNECTED..." appears.

10. Enter yet another [CR]. This causes your PROFILE EXEC to execute. The system will then perform the following tasks automatically:

```
call SPIRES
SET LENGTH 80
SET UPLOW (for upper and lower case)
provide and introductory message
```

Note: Henceforth in this document, commands are assumed to be followed by a [CR], except for ESC-sequences and CNTL-sequences.

## II.2 To LOGOFF

If you have one of the SPIRES prompts (-?, +?, ->, +>), enter: EXIT

The system will respond: **Leaving SPIRES.**

Enter: LOG

### III. Organization of the database.

The ICS database system is comprised of two *files* as shown in Figure 2.

The first file is WILEY1:STATIONS, which consists of twenty-one record-types, of which the first, the goal record record-type is accessed as subfiles and also serve as lookup tables for both verification of input data and conversion to reduce redundancy of data storage. The others serve only as indices. Note that the file name, WILEY1:STATIONS is similar to the subfile name, STATIONS. Except as noted in this document, STATIONS will always refer to the subfile of that name.

The other file, TPHHH:LBLSTAFF consists of 15 record types, of which 5 have subfile access. They contain all of the general employee information such as name, payroll account number, mailstop, building, room, extensions, and termination dates. The SERVICE, ICS, BLDG and OCTOPUS subfiles are part of the LBLSTAFF subfile owned by the TPHHH virtual machine. The LBLSTAFF file is maintained by the Telephone Services Department in the Administration Division. SERVICE and OCTOPUS are merely different subfile accesses to the same goal record record-type; SERVICE is configured to provide rapid name searching, OCTOPUS is configured to make data available to other subfiles in other files easily. The ICS subfile is really the subfile access to the EXT (extensions number index), but with additional elements features. The BLDG subfile provides subfile access to the BLDG index, with additional elements.

Note that the ICS Subfile, the extension number goal-index record-type, is derived from the extension numbers entered into employee and guest records in LBLSTAFF. It will not necessarily coincide with EXT element values in the STATIONS subfile, and the employee name associated with an EXT in LBLSTAFF (SERVICE or OCTOPUS) will not necessarily coincide with the value of the CONTACT element in STATIONS.

Each subfile is selectable as a database in its own right. The solid lines with two-way arrows in figure 2 indicate that data is shared between the two subfiles. The dotted line between the ICS Subfile and the SERVICES Subfile indicates that SERVICE is also used interactively for ID number verification. However, except for key elements, *data is never redundantly stored, but stored only in one database and then accessed by the others. This has the advantage that, when data is updated, it need only be updated in one place, yet this has the effect of updating all the user's data simultaneously.* Not only does this conserve staff effort, it assures consistency from database to database, from mailing list to mailing list.

The ICS database system contains several virtual elements in several subfiles. These are indicated in the listings of elements for each subfile. Virtual elements are elements which do not exist but appear to exist. That is, they are not stored and they cannot be edited. They *can* be displayed, records *can be* sequenced according to their values, and indexes *can be* built based on their values. They are used to avoid duplicate storage of data, thereby avoid problems with "out-of-sync" data. Generally, they are derived from elements in other subfiles, or system information.

By default, only the real elements are displayed in SPIRES. To be able to see the virtual elements, a SET ELEM command must be issued in SPIRES. The SEARCH PRIMER provides details (see Appendix G). This will override any output format that is SET however. To return to the output format, enter CLEAR ELEM [cr]. The command SHOW ELEM will indicate which virtual elements are to be

displayed by notating "- SET" after the element name. If you clear away the virtual elements with CLR ELEM, you can reset them easily by the command

SET ELEMend <element list>.

The ICS system provides a variety of indexes to pertinent information. Some of the indexes index and allow searching for data that does not exist in the ICS system, yet the appropriate records are retrieved.. These are called *indirect indexes*, and can be implemented upon request.

Occasionally you may wish to use some other format than the default SPIRES format. Customized formats have been provided, as described below. The command SHOW FORMats will list these and also indicate if either one is currently in effect by notating "- SET" after the format name.

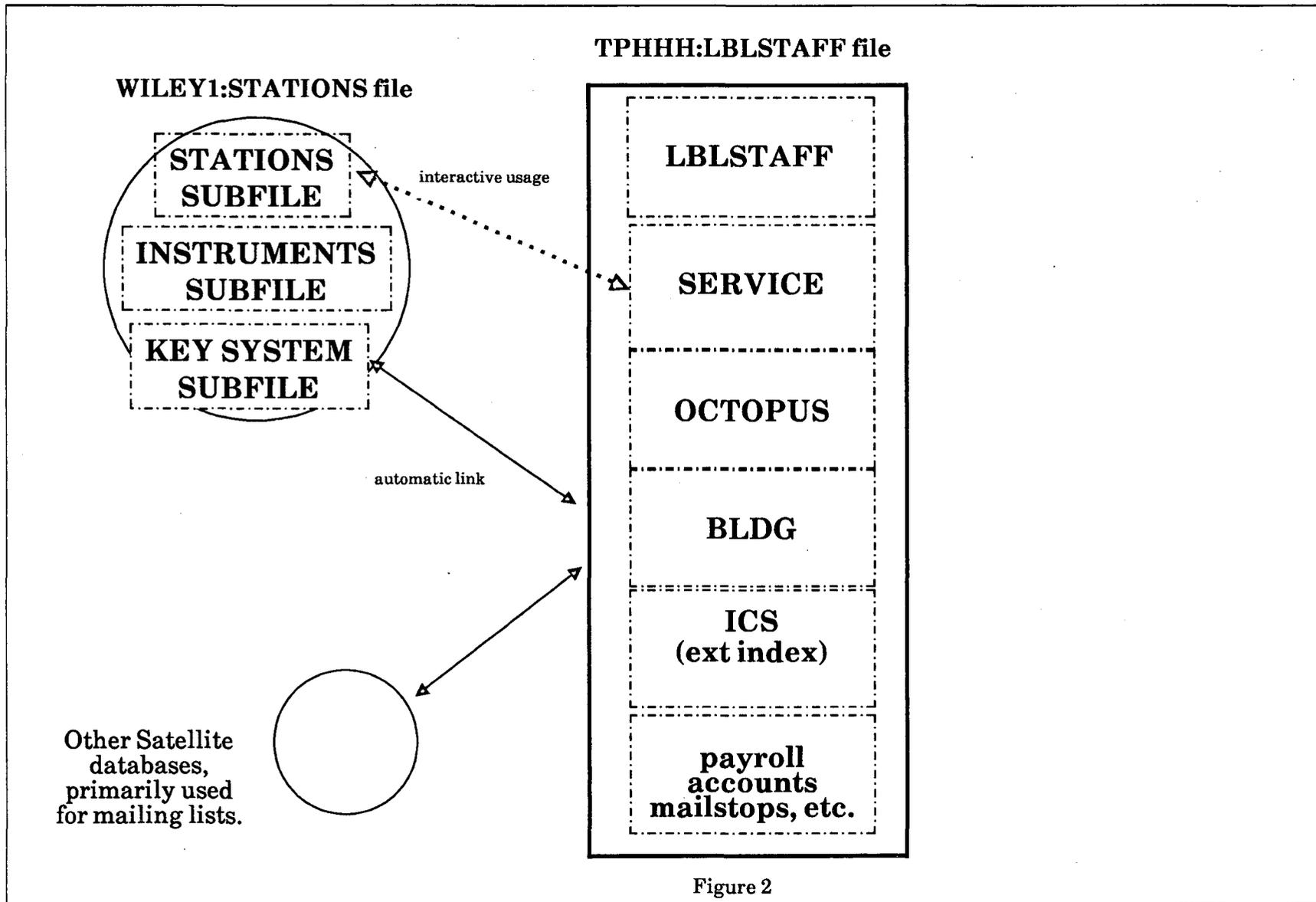


Figure 2

#### IV. Using the ICS Subfile

- 1 Description of elements in the ICS subfile
- 2 Displaying records
- 3 Searching in SPIRES; Searching the ICS subfile
- 4 Updating records
- 5 Adding new records
- 6 Removing records
- 7 Changing the key of a record

The ICS Subfile is a goal-index record-type in the TPHHH:LBLSTAFF file. The purpose of the ICS subfile is to maintain current descriptive information about each extension number as derived from personnel records in LBLSTAFF and to whom it is assigned. This subfile acts as a goal record and as an index record to personnel records in LBLSTAFF. Beside the key (EXT) and the pointer back to LBLSTAFF goal records, it contains special processing rules (phantom structures) that allow the names and other personnel information such as location and termination date to be displayed with an EXT. It also has a customized output format that allows this information to be displayed in an easily readable format and also uses the BLDG subfile to display the ICS SITE NODE for that particular EXT. This format may be used by doing:

#### SET FORMAT ICS

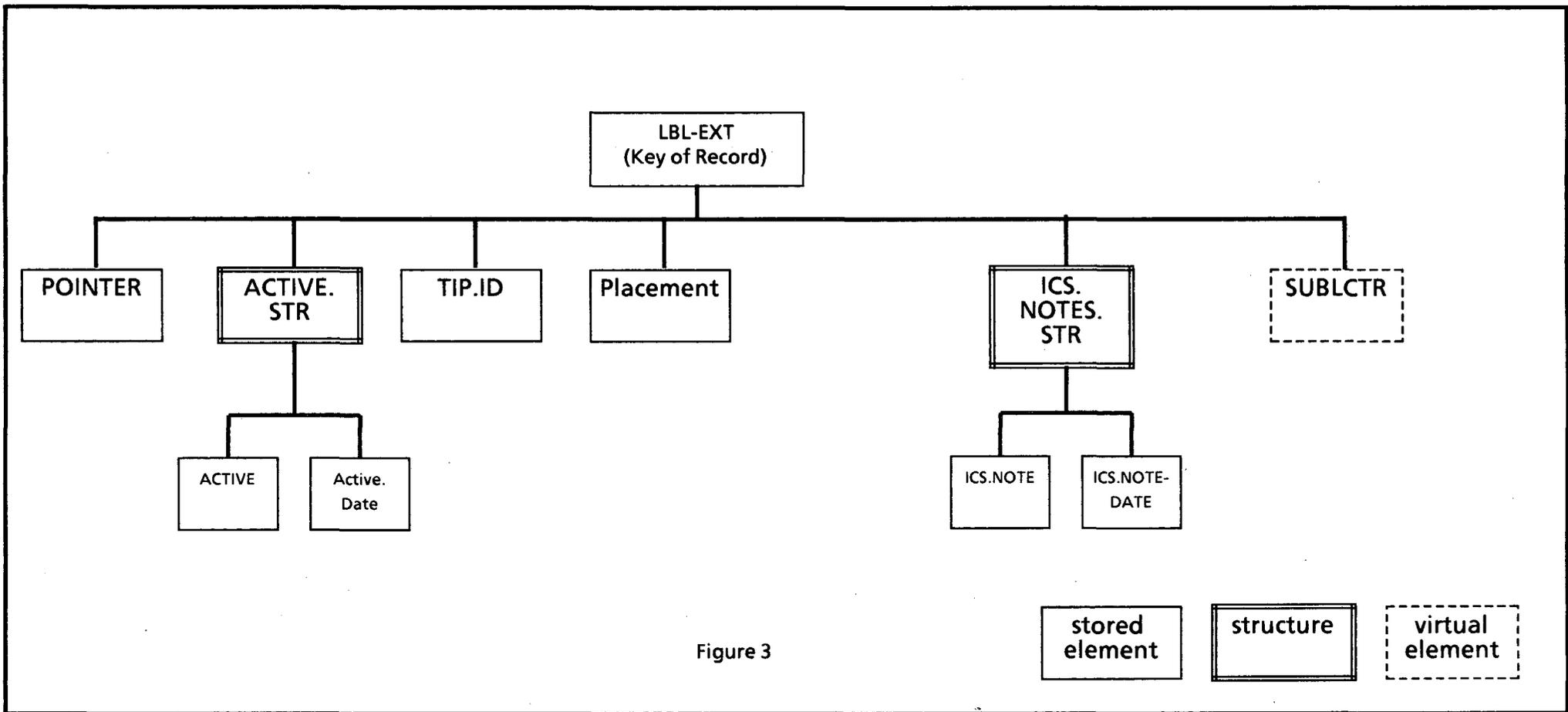
This format is automatically SET when you SElect ICS.

For each record, a particular element may be required or optional, singly or multiply occurring, have controlled allowable values, be limited to a particular type of value, or be indexed for ease in searching, etc. The element listing below describes the characteristics of each element.

#### IV.1 Description of elements in the ICS subfile

<u>Element Name</u>	<u>Required/Opt</u>	<u>Length</u>	<u>Occurrences</u>	<u>Data Type</u>	<u>Indexed</u>
LBL-EXT	Required	variable	Single	Character	
POINTER	Optional	fixed	Multiple	Hex	
Active.Structure	Optional	Variable	Multiple	Structure	
ACTIVE	Required	1	Single	YESNO	
Active.date	Required	4	Single	Date	
TIP.ID (TIP)	Optional	Variable	Single	Text	
Placement (P)	Optional	Variable	Single	Text	
ICS.NOTE.STR	Optional	Variable	Multiple	Structure	
ICS.NOTE (INOTE)	Optional	Variable	Single	Character	
ICS.NOTE-DATE (INDAT)	Automatic	Fixed	Single	Date	
SUBLCTR		0	1	Virtual	Indirect
(Redefines POINTER to fetch data from LBLSTAFF records having that EXT)					

Graphically, the hierarchical nature of a typical ICS record appears:



## IV.2 Displaying Records.

A complete description of all the ways to display records in SPIRES is described in the document Searching and Updating listed in Appendix G.

To use the ICS subfile, you must SElect it with the command:

**SElect ICS**

If you select some other subfile, the you must again SElect ICS to display ICS records.

If you know the LBL-EXT for a record which you wish to see, you may use the DISplay command to view it directly:

**DISplay <LBL-EXT>**

You may also SET FORMAT ICS or SET FORMAT ICSEXT to use the features in those formats.

Global for may be used to display groups of records. For example,

**SET FORMAT ICS  
FOR ADDS or FOR UPDATES  
DISplay ALL or IN ACT CLR DISplay ALL**

will display all the records added or updated that day.

## IV.3 Searching in SPIRES; Searching the ICS Subfile

A complete description of all the searching capabilities in SPIRES is described in the document Searching and Updating listed in Appendix G.

You may search on a non-indexed element, e.g., TIP.ID (See Sect IV.3) i.e., using Global For, then use the DISplay <all/first/last/n/ext> command.

To put the results in your active file so that they may be printed or viewed, prefix the TYPE or DISplay commands with IN ACT CLR:

**IN ACT CLR DIS <LBL-EXT>**

**IN ACT CLR TYPE**

Or you may search for an LBL-EXT in some other appropriate subfile.

#### IV.4 Updating Records.

A complete description of updating records in SPIRES is described in the document Searching and Updating listed in Appendix G. This brief summary provides an overview. See Appendix E for \$PROMPT format subcommands.

To update an ICS record, enter the following commands:

1. **SET FORMAT \$PROMPT** [cr]
2. **MERGE <LBL-EXT>** You will be prompted for a new values.  
If you wish not to change a value, simply enter a [cr].  
Otherwise, the text that you enter will become the new value.
3. **DISplay <LBL-EXT>** to verify that the record is correct.

Finally, it is always a good idea to retain your source documents after you complete any updating. In seven years of running SPIRES at LBL, no data has ever been lost, but users have forgotten why they changed some records.

After the update is finished, the current version of the record will be displayed by SPIRES.

**NOTE:** The key of a record (LBL-EXT) cannot be modified by editing its value as it is primarily an index key to goal records in the LBLSTAFF subfile.

#### IV.5 Adding a new record.

A complete description of adding records in SPIRES is described in the document Searching and Updating listed in Appendix G. Below, a very brief summary and sample session provides an adequate overview. See Appendix E for \$PROMPT format subcommands.

Note that SPIRES creates ICS records whenever a new EXT value is used in the LBLSTAFF subfile.

**SET FORMAT \$PROMPT**

**ADD**

You will be prompted for the value of each element. If an *optional* element should be left blank, enter a carriage return [CR]. Also note that you will be prompted twice for each multiply occurring element. Just enter a [CR] to proceed to the next element. Please see Appendix E for the subcommands used in the \$PROMPT format.

To add several records, simply reissue the ADD command after each previous record is ADDED and DISplayed.

After ADDing a new record, always DISplay <key> to examine it for correctness.

#### IV.6 Removing Records.

A complete description of REMoving records in SPIRES is described in the document Searching and Updating listed in Appendix G. Below, a very brief summary and sample session provides an overview.

Because the ICS subfile is primarily an index record record-type, you are disallowed from REMoving an ICS record where any occurrences of POINTER exist. If you enter a REMove command and the system responds PRIVILEGED COMMAND, then an employee record in LBLSTAFF has that LBL-EXT as the value of its EXT element. You may remove ICS records if there are no occurrences of POINTER.

To REMove a record from the ICS subfile, enter the command:

**REM <LBL-EXT>**

For example, to REMove record 5222, enter:

**REM 5222 [cr]**

Note: If you wish to restore a record that was erroneously removed, please call for assistance.

**NOTE:** To safeguard indexing searching capability, SPIRES will not let you modify or remove any POINTER elements (in fact, they are not displayed in the ICS subfile), *nor can you remove a record that contains occurrences of POINTER.*

#### IV.7 Changing the Key of a Record.

The key of a record (LBL-EXT) cannot be modified because it is the key of an index record for LBLSTAFF goal records. Further, it is unlikely that such an approach is desirable since ICS records also serve as index records to LBLSTAFF. The proper procedure is to SElect LBLSTAFF, FINd EXT <key of desired EXT>, change the EXT values in the result. This will create a new ICS record with the desired key.

Please ask for human help if you believe you need to change the key of an ICS record. (Appendix H).

## V. Using the STATIONS Subfile

- 1 Description of elements in the STATIONS subfile
- 2 Displaying records
- 3 Searching in SPIRES; Searching the STATIONS subfile
- 4 Updating records
- 5 Adding new records
- 6 Removing records
- 7 Changing the key of a record

The purpose of the STATIONS subfile is to maintain current descriptive information about each instrument or telephone set. It may contain special processing rules (phantom structures and paths) that allow the names and other personnel information such as location and termination date to be displayed with an EXT value. It also may have customized output formats, if requested.

The STATION identifier is a unique identifier (a "slot key" in SPIRES terminology) assigned to the record. It is assigned automatically by SPIRES when the record is added to the STATIONS subfile.

Note that the EXT element value in the STATIONS subfile is as found on key-system data sheets and similar source data. The EXT element in LBLSTAFF (SERVICE and OCTOPUS subfiles) and the LBL-EXT element in the ICS subfile are from a different source in the Telephone Services operation and may not necessarily coincide with the EXT element in the STATIONS subfile.

For each record, a particular element may be required or optional, singly or multiply occurring, have controlled allowable values, be limited to a particular type of value, or be indexed for ease in searching, etc. The element listing below describes the characteristics of each element.

Certain data validation is provided. For example, an error is reported if an attempt is made to add a STATION record with a STATION.TYPE value that is not already in the INSTRUMENTS subfile, or a KEY.SYSTEM.ID that is not already in the KEY SYSTEM Subfile. Those subfiles serve as lookup tables for data validation and conversion, and are described below in Sections VI and VII.

## V.1 Description of elements in the STATIONS subfile

<u>Element Name</u>	<u>Required/Opt</u>	<u>Length</u>	<u>Occurrences</u>	<u>Data Type</u>	<u>Indexed</u>
STATION (key of record)	Automatic	Fixed (4)	Single	SLOTCHECK (numeric)	Immed.
Station.type (STYPE) (Values validated in the INSTRUMENTS subfile upon entry.)	Optional	Variable	Single	String	Immed
Active.Flag (FLAG) (default = YES)	Optional	Fixed (1)	Single	YESNO	Immed
Active.History (AHIST)	Optional	Variable	Multiple	Text	
EXT	Optional	Variable	Multiple	Text	Immed
Phone.Number (phone)	Optional	Variable	Multiple	Text	Immed in EXT
Contact	Optional	Variable	Multiple	Name	Immed
Location	Optional	Variable	Single	Structure	
BLDG (Values validated in the BLDG subfile upon entry)	Optional	Variable	Single	Text	Immed
Room	Optional	Variable	Single	Text	Immed
Placement (P)	Optional	Variable	Single	Text	
ICSSITE	Virtual	0	1	Virtual	
Key.System (KSYS)	Optional	Variable	Multiple	Structure	
Key.System.ID (KSID) (Values validated in the KEY SYSTEM Subfile upon entry.)	Optional	Variable	Single	Text	Immed
Key.System.Desig (KSD)	Optional	Variable	Single	Text	Immed
Key.System.Contact (KSC)	Virtual	0	1	Virtual	
Key.System.Division (KSDIV)	Virtual	0	1	Virtual	
UR.STR	Optional	Variable	Single	Structure	
UR.Flag (UFLAG)	Optional	1	Single	YESNO	Immed
UR.Number (UNO)	Optional	Variable	Single	Text	Immed
UR.HISTORY (UHIST)	Optional	Variable	Multiple	Text	
UR.Date	Automatic	Fixed 4	Single	Date	
NOTE.STR	Optional	Variable	Multiple	Structure	
NOTE	Optional	Variable	Single	Character	
NOTE-DATE	Automatic	Fixed 4	Single	Date	
Date.Update (DUPD)	Automatic	Fixed 4	Single	Date	
Instr.desc	Virtual	0	1	Virtual	
Instr.Price	Virtual	0	1	Virtual	
EXTLINK	Virtual	0	1	Virtual	

Graphically, the hierarchical nature of a typical STATIONS record appears:

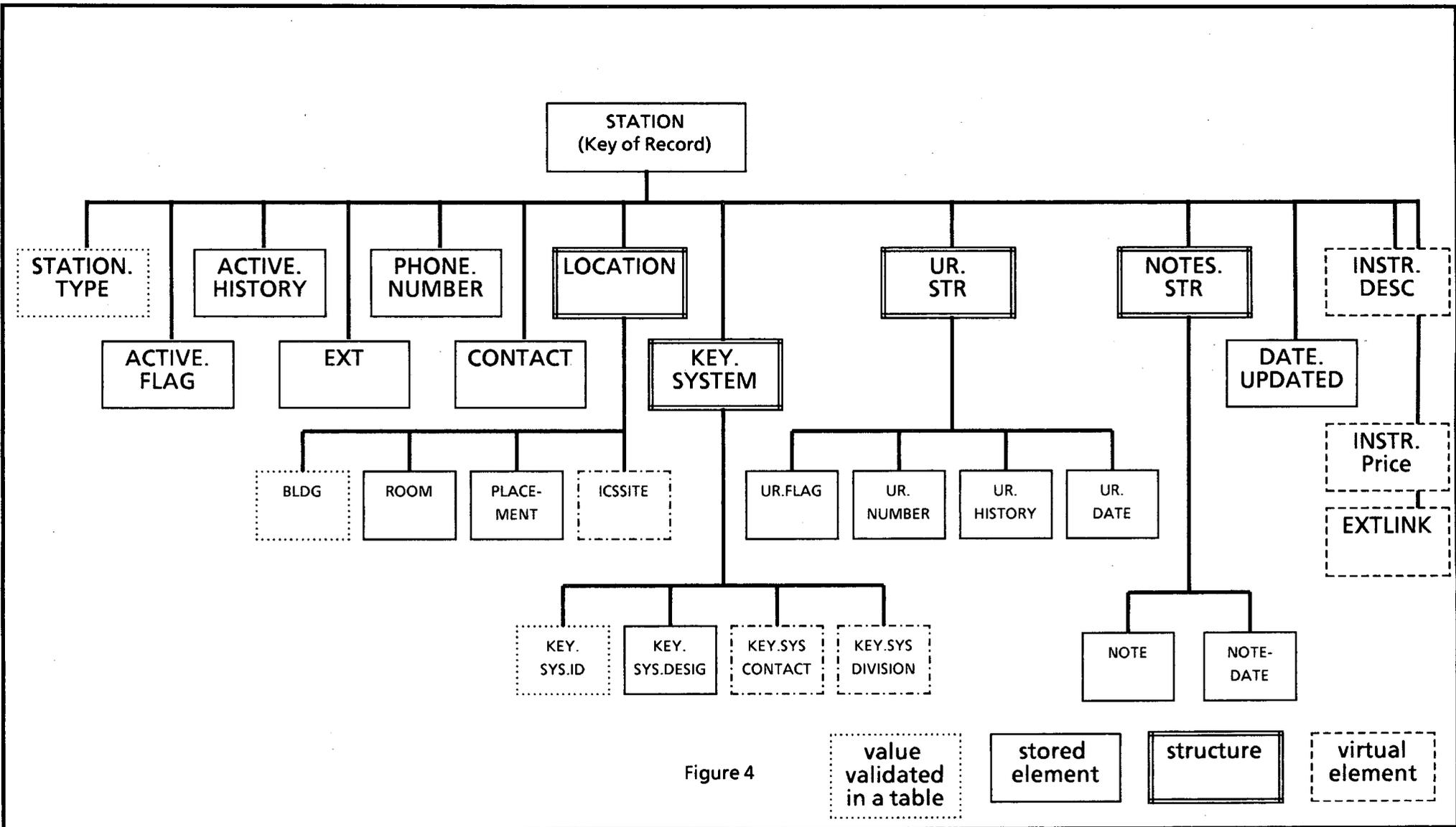


Figure 4

## V.2 Displaying Records.

A complete description of all the ways to display records in SPIRES is described in the document Searching and Updating listed in Appendix G.

To use the STATIONS subfile, you must SElect it with the command:

**SElect STATIONS**

If you select some other subfile, the you must again SElect STATIONS to display STATIONS records.

If you know the STATION identifier for a record which you wish to see, you may use the DISplay command to view it directly:

**DISplay <STATION key value>**

or, by using the FINd command, and TYPE:

**FINd <STATION key value>  
TYPE**

If you do not know the key of a STATIONS record which you wish to see, then you must search for it based upon some criteria you do know. Use the SHOW INDEXes and FINd commands to search for records in this way (Section V.3) Then, to look at the records which are the result of a FINd command, enter the command TYPE. All of the records in the search result will then be displayed.

If you search on a non-indexed element (See Sect V.3) i.e., using Global For, then use the DISplay <all/first/last/n/ext> command.

To put the results in your active file so that they may be printed or viewed, prefix the TYPE or DISplay commands with IN ACT CLR:

**IN ACT CLR DIS <STATION key value>  
IN ACT CLR TYPE**

Global for may be used to display groups of records. For example,

**FOR ADDS or FOR UPDATES or FOR SUBFile  
DISplay ALL or IN ACT CLR DISplay ALL**

will display all the records added or updated that day.

By default, only the real (i.e., stored) elements are displayed in SPIRES. To be able to see the virtual elements, a SET ELEM command must be issued in SPIRES. However, for the STATIONS subfile, you can reset them easily by the command SETELEM. This command is the equivalent of:

**SET ELEM ALL + Idesc Price KEY.SYS.CONTACT KEY.SYS.DIVISION ICSSITE**

If you wish *not* to see them, you may issue the command CLR ELEM. The command SHOW ELEM will indicate which virtual elements are to be displayed by notating "-SET" after the element name. Notice that SETELEM has no space between the words, but CLR ELEM does.

Customized output formats may also be made or format \$REPORT may be used to display records in *other than* the default SPIRES output format, i.e.,

**<Element name> <equal sign> <element value> <semicolon>**

### V.3 Searching in SPIRES; Searching the STATIONS Subfile

You may search for STATIONS records based on any element or combination of elements. However, some elements are used as the basis of searching much more often than others. Those elements are *indexed* in the same way as selected keywords are indexed in the back of a book. Rather than searching sequentially through a book to find a particular topic, you find the topic in the index. Associated with its entry is an *address*, usually a page number. SPIRES indexes work in much the same way. Indexed elements are listed along with their "addresses". However, you never have to worry about the addresses. You simply enter a FIND command, and SPIRES fetches the addresses and then allows you to display, re-sequence, or update the records as desired.

To see a list of the elements in the subfile, enter the command **SHOW ELEMENTS**.

To see a list of indexes, enter the command **SHOW INDEXES**.

Those elements which can be searched using the FIND command are indicated in the column marked "Indexed" in the element list in section V.1.

To search for STATIONS based on any of these elements, use the FIND command, as follows:

**FIND <index name> <relational operator> <value>**

For example, to find the extension 5458, enter:

**FIND EXT = 5458**

Then use the TYPE command to see the result (Section V.2).

If you do not include the relational operator in your search, SPIRES assumes an "equals" operator:

**FIND EXT 5458**

If you are not sure how the values in an index appear, you can BROWSE the index:

**BROWSE <index name> [optional value]**

For example,

**BROWSE EXT 5458**

will show the values in the vicinity of 5458, including 5458 if a goal record has 5458 as the value for its EXT element.

Indexes are updated immediately when a record containing indexed elements is added or updated.

A complete description of all the searching capabilities in SPIRES is described in the document Searching and Updating listed in Appendix G.

#### V.4 Updating Records.

A complete description of updating records in SPIRES is described in the document Searching and Updating listed in Appendix G. This brief summary provides an overview. See Appendix E for \$PROMPT format subcommands.

To update a STATIONS record, enter the following commands:

1. Determine the key of the record you wish to modify.(e.g., with FINd and TYPE commands)
2. DISplay <Station key value> Recommended but not mandatory.
3. SETPROMP [cr]
4. MERGE <STATION key value> You will be prompted for a new values of existing elements and new occurrences of element that are multiply occurring . If you wish not to change a value, simply enter a [cr]. Otherwise, the text that you enter will become the new value.
5. DISplay <STATION key value> to verify that the record is correct.

Finally, it is always a good idea to retain your source documents after you complete any updating. In seven years of running SPIRES at LBL, no data has ever been lost, but users have forgotten why they changed some records.

After the update is finished, the current version of the record will be displayed by SPIRES. Indexes are updated immediately.

It is advisable to DISplay a record before you modify it.

**NOTE:** The key of a record (STATION) **cannot** be modified by editing its value. To change the key of a record, please see Section V.7.

Here's a sample session showing how to add a record (system responses in **bold**):

```
-?  
sel stations (not necessary if STATIONS is already SElected)  
-?  
Clear format  
-?  
setelem  
-?  
dis 522  
STATION = 522;  
STATION.TYPE = MX10;  
ACTIVE.FLAG = Yes;  
EXT(1) = 5458;  
Contact(1) = Joe Blow;  
Structure LOCATION  
  BLDG = 50B;  
  ROOM = 2258;  
  Placement = By the window;  
  ICSSITE = 1;  
Structure Key.System  
  KEY.SYSTEM.ID = 35;  
  KEY.SYSTEM.DESIG = 2B;
```

```
KEY.SYSTEM.CONTACT = Bob Fink;
KEY.SYSTEM.DIVISON = ICSD;
Structure NOTES(1)
NOTE = The source document is in my desk.;
NOTE.DATE = July 23, 1987;
Date.Updated = July 23, 1987;
INSTR.DESC = Merlin X10;
INSTR.PRICE = $35.00;
-?
```

```
setprompt
-?
```

```
MERGE 522
:STATION.TYPE = MX10
:STATION.TYPE: [CR]
:ACTIVE.FLAG = Yes
:ACTIVE.FLAG: [CR]
:EXT(1) = 5458
:EXT(1): [CR]
:EXT:(2)
:Phone.Number(1):
:Contact(1) = Joe Blow
:Contact(1): Charles Doe
:Contact(2): [CR]
```

```
Structure: LOCATION
: BLDG = 50B
: BLDG: [CR]
: ROOM = 2258
: ROOM: [CR]
: Placement = By the window
: Placement: [CR]
```

```
Structure: Key.System
: KEY.SYSTEM.ID = 35
: KEY.SYSTEM.ID: [CR]
: KEY.SYSTEM.DESIG = 2B
: KEY.SYSTEM.DESIG: [CR]
: KEY.SYSTEM.ID:
```

```
Structure: UR.STR
: UR.FLAG: [CR]
```

```
Structure: NOTES(1)
: NOTE = The source document is in my desk.
: NOTE [CR]
: NOTE(2):This is a second note
```

```
-?
Clear format
-?
setelem
```

-?  
dis 522 (or: /dis \$key)  
STATION = 522;  
STATION.TYPE = MX10;  
ACTIVE.FLAG = Yes;  
EXT(1) = 5458;  
Contact(1) = Charles Doe;  
Structure LOCATION  
BLDG = 50B;  
ROOM = 2258;  
Placement = By the window;  
ICSSITE = 1;  
Structure Key.System  
KEY.SYSTEM.ID = 35;  
KEY.SYSTEM.DESIG = 2B;  
KEY.SYSTEM.CONTACT = Bob Fink;  
KEY.SYSTEM.DIVISON = ICSD;  
Structure NOTES(1)  
NOTE = The source document is in my desk.;  
NOTE.DATE = July 23, 1987;  
Structure NOTES(2)  
NOTE = This is a second note note;  
NOTE.DATE = July 24, 1987;  
Date.Updated = July 24, 1987;  
INSTR.DESC = Merlin X10;  
INSTR.PRICE = \$35.00;

Notice that the Contact, Notes, and Date.Updated elements have changed.

## V.5 Adding a new record.

A complete description of adding records in SPIRES is described in the document Searching and Updating listed in Appendix G. Below, a very brief summary and sample session provides an adequate overview. See Appendix E for \$PROMPT format subcommands.

### SETPROMP

#### ADD

You will be prompted for the value of each element. If an *optional* element should be left blank, enter a carriage return [CR]. Also note that you will be prompted twice for each multiply occurring element. Just enter a [CR] to proceed to the next element. Please see Appendix E for the subcommands used in the \$PROMPT format.

To add several records, simply reissue the ADD command after each previous record is ADDED and DISplayed.

After ADDing a new record, always DISplay <key> to examine it for correctness.

If you attempt to enter a STATION.TYPE, BLDG, or KEY.SYSTEM.ID that does not occur in the appropriate subfile, you will receive an error. These elements are validate in the INSTRUMENTS, BLDG, and KEY SYSTEM subfiles which also act as lookup tables. If you receive an error, do: (backslash) X and select the proper subfile and add the appropriate record.

Here's a sample session showing how to add a record (system responses in **bold**):

```
sel stations (not necessary if STATIONS is already SElected)
```

```
-?
```

```
setprompt
```

```
-?
```

```
ADD
```

```
:STATION.TYPE: MX10 [CR]
```

```
:ACTIVE.FLAG: Yes [CR]
```

```
:EXT(1): 5458 [CR]
```

```
:EXT(2): [CR]
```

```
:PHONE.NUMBER(1): [CR]
```

```
:Contact(1): Joe Blow
```

```
:Contact(2): [CR]
```

```
Structure: LOCATION
```

```
: BLDG: 50B [CR]
```

```
: ROOM: 2258 [CR]
```

```
: Placement: By the window [CR]
```

```
Structure: Key.System(1)
```

```
: KEY.SYSTEM.ID: 35 [CR]
```

```
: KEY.SYSTEM.DESIG: 2b [CR]
```

```
: KEY.SYSTEM.ID(2): [CR]
```

```
Structure: NOTES(1)
```

```
: NOTE(1): = The source document is in my desk. [CR]
```

```
: NOTE(2): [CR]
```

```
-Added record: 522
```

-?  
Clear format  
-?  
setelem  
-?  
dis 522 (or: /dis \$key)  
STATION = 522;  
STATION.TYPE = MX10;  
ACTIVE.FLAG = Yes;  
EXT(1) = 5458;  
Contact(1) = Joe Blow;  
Structure LOCATION  
BLDG = 50B;  
ROOM = 2258;  
Placement = By the window;  
ICSSITE = 1;  
Structure Key.System  
KEY.SYSTEM.ID = 35;  
KEY.SYSTEM.DESIG = 2B;  
KEY.SYSTEM.CONTACT = Bob Fink;  
KEY.SYSTEM.DIVISON = ICSD;  
Structure NOTES(1)  
NOTE = The source document is in my desk.;  
NOTE.DATE = July 23, 1987;  
Date.Updated = July 23, 1987;  
INSTR.DESC = Merlin X10;  
INSTR.PRICE = \$35.00;

## V.6 Removing Records.

A complete description of REMoving records in SPIRES is described in the document Searching and Updating listed in Appendix G. Below, a very brief summary and sample session provides an overview.

To REMove a record from the STATIONS subfile, enter the command:

**REM <STATION key>**

For example, to REMove record 522, enter:

**REM 522 [cr]**

Note: If you wish to restore a record that was erroneously removed, please call for assistance.

## V.7 Changing the Key of a Record.

The key of a record (STATION) cannot be modified by editing its value and then issuing an UPDate command. To change the key of a record, enter the following commands:

1. **CLEAR FORMAT**
2. **TRAnsfer <old STATION key> CLR**
3. **X ACTIVE FILE**
4. edit the STATION and other elements as needed using the XEDIT editor. (APPENDIX I describes use of the editor).
5. **FILE** on the command line at the bottom of the screen.
6. **ADD**
7. **REMOve <key of the old STATION record>** (as described in Sect. V.6 above)

Please ask for human help if you encounter any difficulties (Appendix H).

## VI. Using the KEY SYSTEM Subfile

- 1 Description of elements in the KEY SYSTEM subfile
- 2 Displaying records
- 3 Searching in SPIRES; Searching the KEY SYSTEM subfile
- 4 Updating records
- 5 Adding new records
- 6 Removing records
- 7 Changing the key of a record

The purposes of the KEY SYSTEM subfile are: to maintain current descriptive information about each key system, to serve as an index to goal records in the STATIONS subfile, and to serve as a lookup table for data validation of goal records in the STATIONS subfile.

The KEY.SYSTEM (note the "dot") identifier is a unique key assigned to the record.

For each record, a particular element may be required or optional, singly or multiply occurring, have controlled allowable values, be limited to a particular type of value, or be indexed for ease in searching, etc. The element listing below describes the characteristics of each element.

### VI.1 Description of elements in the KEY SYSTEM subfile

<u>Element Name</u>	<u>Required/Opt</u>	<u>Length</u>	<u>Occurrences</u>	<u>Data Type</u>	<u>Indexed</u>
KEY.SYSTEM (key of record) (KS)	Required	Variable	Single	Text	Immed.
Contact	Optional	Variable	Multiple	Name	Immed
Division	Optional	Variable	Single	Text	Immed.
Key.Sheet.Date	Optional	Variable	Multiple	Date	
NOTE.STR	Optional	Variable	Multiple	Structure	
NOTE	Optional	Variable	Single	Character	
NOTE-DATE	Automatic	Fixed 4	Single	Date	
Date.Update (DUPD)	Automatic	Fixed 4	Single	Date	

Graphically, the hierarchical nature of a typical KEY SYSTEM record appears:

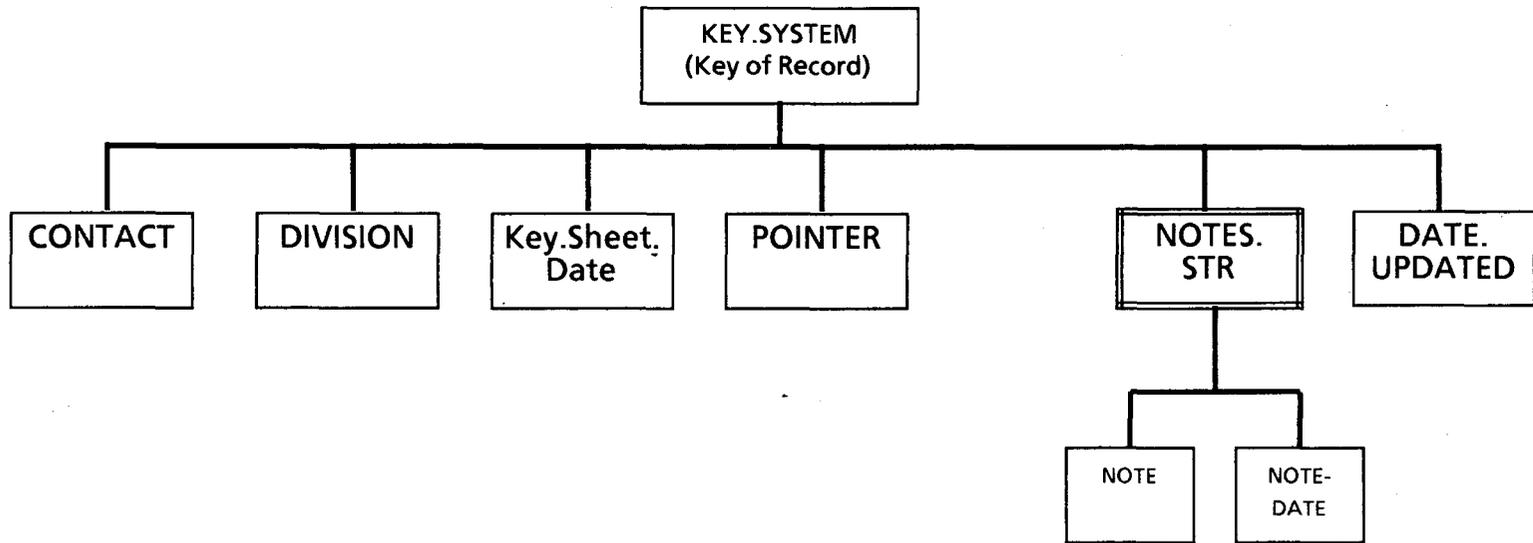
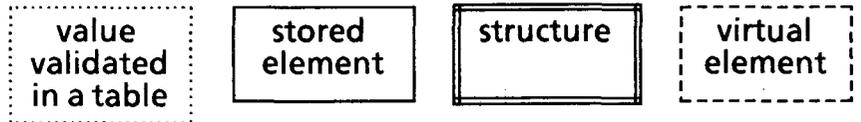


Figure 5



## VI.2 Displaying Records.

A complete description of all the ways to display records in SPIRES is described in the document Searching and Updating listed in Appendix G.

To use the KEY SYSTEM subfile, you must SElect it with the command:

**SElect KEY SYSTEM**

If you select some other subfile, the you must again SElect KEY SYSTEM to display KEY SYSTEM records.

If you know the KEY SYSTEM key for a record which you wish to see, you may use the DISplay command to view it directly:

**DISplay <KEY.SYSTEM>**

or, by using the FINd command, and TYPE:

**FINd <KEY.SYSTEM>  
TYPE**

If you do not know the key of a KEY SYSTEM record which you wish to see, then you must search for it based upon some criteria you do know, either contact name or division. Use the SHOW INDEXes and FINd commands to search for records in this way (Section VI.3) Then, to look at the records which are the result of a FINd command, enter the command TYPE. All of the records in the search result will then be displayed.

If you search on a non-indexed element (See Sect VI.3) i.e., using Global For, then use the DISplay <all/first/last/n/ext> command.

To put the results in your active file so that they may be printed or viewed, prefix the TYPE or DISplay commands with IN ACT CLR:

**IN ACT CLR DIS < key>  
IN ACT CLR TYPE**

Global for may be used to display groups of records. For example,

**FOR ADDS or FOR UPDATES  
DISplay ALL or IN ACT CLR DISplay ALL**

will display all the records added or updated that day.

Customized output formats may also be made or format \$REPORT may be used to display records in *other than* the default SPIRES output format, i.e.,

**<Element name> <equal sign> <element value> <semicolon>**

### VI.3 Searching in SPIRES; Searching the KEY SYSTEM Subfile

You may search for KEY SYSTEM records based on any element or combination of elements. However, some elements are used as the basis of searching much more often than others. Those elements are *indexed* in the same way as selected keywords are indexed in the back of a book. Rather than searching sequentially through a book to find a particular topic, you find the topic in the index. Associated with its entry is an *address*, usually a page number. SPIRES indexes work in much the same way. Indexed elements are listed along with their "addresses". However, you never have to worry about the addresses. You simply enter a FIND command, and SPIRES fetches the addresses and then allows you to display, re-sequence, or update the records as desired.

To see a list of the elements in the subfile, enter the command **SHOW ELEMENTS**.

To see a list of indexes, enter the command **SHOW INDEXES**.

Those elements which can be searched using the FIND command are indicated in the column marked "Indexed" in the element list in section VI.1.

To search for KEY SYSTEM records based on any of these elements, use the FIND command, as follows:

**FIND <index name> <relational operator> <value>**

For example, to find all the key systems in the ICSD division, enter:

**FIND DIVISION = ICSD**  
or  
**FIND DIV ICS#**

Then use the TYPE command to see the result (Section VI.2). The # is a truncation character or "wild card" character. In the example above, it causes the FIND command to find all records where the DIVISION element value begins with the characters ICS.

If you do not include the relational operator in your search, SPIRES assumes an "equals" operator:

**FIND DIV ICSD**

If you are not sure how the values in an index appear, you can BROWSE the index:

**BROWSE <index name> [optional value]**

For example,

**BROWSE DIV ICS**

will show the values in the vicinity of ICS.

Indexes are updated immediately when a record containing indexed elements is added or updated.

A complete description of all the searching capabilities in SPIRES is described in the document Searching and Updating listed in Appendix G.

#### VI.4 Updating Records.

A complete description of updating records in SPIRES is described in the document Searching and Updating listed in Appendix G. This brief summary provides an overview. See Appendix E for \$PROMPT format subcommands.

To update a KEY SYSTEM record, enter the following commands:

1. Determine the key of the record you wish to modify.(e.g., with FINd and TYPE commands)
2. DISplay <key> Recommended but not mandatory.
3. PROMPTKS [cr]
4. MERGE <key> You will be prompted for a new values of existing elements and new occurrences of element that are multiply occurring . If you wish not to change a value, simply enter a [cr]. Otherwise, the text that you enter will become the new value.
5. DISplay <key> to verify that the record is correct.

Finally, it is always a good idea to retain your source documents after you complete any updating. In seven years of running SPIRES at LBL, no data has ever been lost, but users have forgotten why they changed some records.

After the update is finished, the current version of the record will be displayed by SPIRES. Indexes are updated immediately.

It is advisable to DISplay a record before you modify it.

**NOTE:** The key of a record (KEY.SYSTEM) cannot be modified by editing its value. To change the key of a record, please see Section VI.7.

Here's a sample session showing how to add a record (system responses in bold):

```
-?  
sel key system (not necessary if KEY SYSTEM is already SElected)  
-?  
Clear format  
-?  
dis 35  
KEY.SYSTEM = 35;  
CONTACT(1) = Bob Fink;  
DIVISION = ICSD;  
Structure NOTES(1)  
NOTE = The source documents are in my desk.;  
NOTE.DATE = July 23, 1987;  
Date.Updated = July 23, 1987;  
-?
```

-?

promptks

-?

MERGE 35

KEY.SYSTEM = 35

CONTACT(1) = Bob Fink

CONTACT(1): Robert Fink [CR]

CONTACT(2): [CR]

DIVISION = ICSD;

DIVISION: [CR]

Structure NOTES(1)

NOTE = The source documents are in my desk.;

: NOTE [CR]

: NOTE(2): This is a second note

: NOTE(3): [CR]

-?

Clear format

-?

dis 35 (or: /dis \$key)

KEY.SYSTEM = 35;

CONTACT(1) = Robert Fink;

DIVISION = ICSD;

Structure NOTES(1)

NOTE = The source document are in my desk.;

NOTE.DATE = July 23, 1987;

Structure NOTES(2)

NOTE = This is a second note;

NOTE.DATE = July 24, 1987;

Date.Updated = July 24, 1987;

Notice that the Contact, Notes, and Date.Updated elements have changed.

## VI.5 Adding a new record.

A complete description of adding records in SPIRES is described in the document Searching and Updating listed in Appendix G. Below, a very brief summary and sample session provides an adequate overview. See Appendix E for \$PROMPT format subcommands.

**PROMPTKS** [CR]

**ADD**

You will be prompted for the value of each element. If an *optional* element should be left blank, enter a carriage return [CR]. Also note that you will be prompted twice for each multiply occurring element. Just enter a [CR] to proceed to the next element. Please see Appendix E for the subcommands used in the \$PROMPT format.

To add several records, simply reissue the ADD command after each previous record is ADDED and DISplayed.

After ADDing a new record, always DISplay <key> to examine it for correctness.

Here's a sample session showing how to add a record (system responses in **bold**):

```
sel key system (not necessary if KEY SYSTEM is already SElected)
```

```
-?
```

```
promptks
```

```
-?
```

```
ADD
```

```
KEY.SYSTEM: 35 [CR]
```

```
CONTACT(1): Bob Fink [CR]
```

```
CONTACT(2): [CR]
```

```
DIVISION = ICSD [CR]
```

```
DIVISION: [CR]
```

```
Structure NOTES
```

```
NOTE(1) = The source documents are in my desk. [CR]
```

```
: NOTE(2): [CR]
```

```
-?
```

```
Clear format
```

```
-?
```

```
dis 35 (or: /dis $key)
```

```
KEY.SYSTEM = 35;
```

```
CONTACT(1) = Bob Fink;
```

```
DIVISION = ICSD;
```

```
Structure NOTES(1)
```

```
NOTE = The source documents are in my desk.;
```

```
NOTE.DATE = July 23, 1987;
```

```
Date.Updated = July 23, 1987;
```

## VI.6 Removing Records.

A complete description of REMoving records in SPIRES is described in the document Searching and Updating listed in Appendix G. Below, a very brief summary and sample session provides an overview.

To REMove a record from the KEY SYSTEM subfile, enter the command:

**REM < key >**

For example, to REMove record 35, enter:

**REM 35 [cr]**

Note: If you wish to restore a record that was erroneously removed, please call for assistance.

**NOTE:** To safeguard indexing searching capability, SPIRES will not let you modify or remove any POINTER elements (in fact, they are not displayed in the KEY SYSTEM subfile), *nor can you remove a record that contains occurrences of POINTER.*

## V.7 Changing the Key of a Record.

The key of a record (STATION) **cannot** be modified by editing its value and then issuing an UPDate command. To change the key of a record, enter the following commands:

1. **CLEAR FORMAT**
2. **TRAnser <old STATION key> CLR**
3. **X ACTIVE FILE**
4. edit the STATION and other elements as needed using the XEDIT editor. (APPENDIX I describes use of the editor).
5. **FILE** on the command line at the bottom of the screen.
6. **ADD**
7. **REMOve <key of the old STATION record>** (as described in Sect. VI.6 above)

Please ask for human help if you encounter any difficulties (Appendix H).

## VII. Using the INSTRUMENTS Subfile.

- 1 Description of elements in the INSTRUMENTS subfile
- 2 Displaying records
- 3 Searching in SPIRES; Searching the INSTRUMENTS subfile
- 4 Updating records
- 5 Adding new records
- 6 Removing records
- 7 Changing the key of a record

The purposes of the INSTRUMENTS subfile are: to maintain current descriptive information about each instrument type, to serve as an index to goal records in the STATIONS subfile, and to serve as a lookup table for data validation of goal records in the STATIONS subfile.

The INSTRUMENT.TYPE (note the "dot") identifier is a unique key assigned to the record.

For each record, a particular element may be required or optional, singly or multiply occurring, have controlled allowable values, be limited to a particular type of value, or be indexed for ease in searching, etc. The element listing below describes the characteristics of each element.

### VII.1 Description of elements in the KEY SYSTEM subfile.

<u>Element Name</u>	<u>Required/Opt</u>	<u>Length</u>	<u>Occurrences</u>	<u>Data Type</u>	<u>Indexed</u>
INSTRUMENT.TYPE (TYPE) (key of record)	Required	Variable	Single	Text	Immed.
INSTR.DESC (IDESC)	Optional	Variable	Single	Text	Immed word
INSTR.PRICE (PRICE)	Optional	Fixed 4	Single	dollar	
NOTE.STR	Optional	Variable	Multiple	Structure	
NOTE	Optional	Variable	Single	Character	
NOTE-DATE	Automatic	Fixed 4	Single	Date	
Date.Update (DUPD)	Automatic	Fixed 4	Single	Date	

Graphically, the hierarchical nature of a typical INSTRUMENTS record appears:

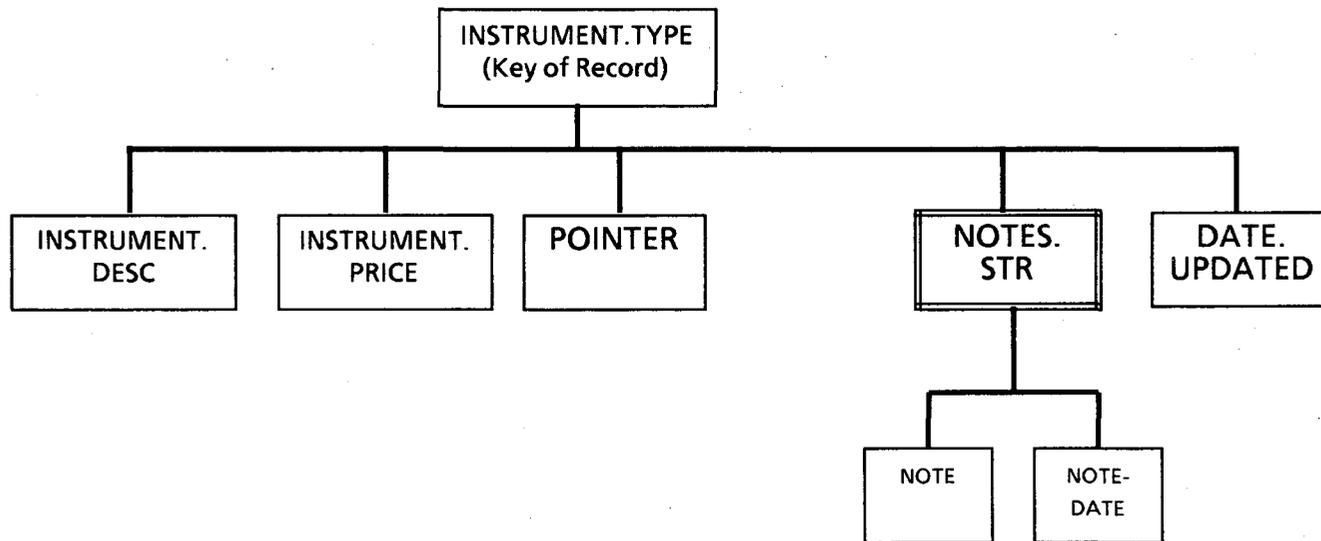


Figure 5

value  
validated  
in a table

stored  
element

structure

virtual  
element

## VII.2 Displaying Records.

A complete description of all the ways to display records in SPIRES is described in the document Searching and Updating listed in Appendix G.

To use the INSTRUMENTS subfile, you must SElect it with the command:

**SElect INSTRUMENTS**

If you select some other subfile, the you must again SElect INSTRUMENTS to display INSTRUMENTS records.

If you know the INSTRUMENT type identifier for a record which you wish to see, you may use the DISplay command to view it directly:

**DISplay <instrument.type key value>**

or, by using the FINd command, and TYPE:

**FINd <instrument.type key value>  
TYPE**

If you do not know the key (instrument type) of a INSTRUMENTS record which you wish to see, then you must search for it based upon some criteria you do know. Use the SHOW INDEXes and FINd commands to search for records in this way (Section VII.3) Then, to look at the records which are the result of a FINd command, enter the command TYPE. All of the records in the search result will then be displayed.

If you search on a non-indexed element (See Sect VI.3) i.e., using Global For, then use the DISplay <all/first/last/n/ext> command.

To put the results in your active file so that they may be printed or viewed, prefix the TYPE or DISplay commands with IN ACT CLR:

**IN ACT CLR DIS <instrument.type key value>  
IN ACT CLR TYPE**

Global for may be used to display groups of records. For example,

**FOR ADDS or FOR UPDATES  
DISplay ALL or IN ACT CLR DISplay ALL**

will display all the records added or updated that day.

Customized output formats may also be made or format \$REPORT may be used to display records in *other than* the default SPIRES output format, i.e.,

**<Element name> <equal sign> <element value> <semicolon>**

### VII.3 Searching in SPIRES; Searching the INSTRUMENTS Subfile.

You may search for INSTRUMENTS records based on any element or combination of elements. However, some elements are used as the basis of searching much more often than others. Those elements are *indexed* in the same way as selected keywords are indexed in the back of a book. Rather than searching sequentially through a book to find a particular topic, you find the topic in the index. Associated with its entry is an *address*, usually a page number. SPIRES indexes work in much the same way. Indexed elements are listed along with their "addresses". However, you never have to worry about the addresses. You simply enter a FIND command, and SPIRES fetches the addresses and then allows you to display, re-sequence, or update the records as desired.

To see a list of the elements in the subfile, enter the command **SHOW ELEMENTS**.

To see a list of indexes, enter the command **SHOW INDEXES**.

Those elements which can be searched using the FIND command are indicated in the column marked "Indexed" in the element list in section VII.1.

To search for INSTRUMENTS based on any of these elements, use the FIND command, as follows:

**FIND <index name> <relational operator> <value>**

For example, to find all the instruments that are part of a Merlin system, enter:

**FIND INSTRUMENT.DESC = Merlin**

Then use the TYPE command to see the result (Section VII.2).

If you do not include the relational operator in your search, SPIRES assumes an "equals" operator:

**FIND IDESC Merlin**

or

**FIND idesc merl#**

Then use the TYPE command to see the result (Section VII.2). The # is a truncation character or "wild card" character. In the example above, it causes the FIND command to find all records where the instrument description element value begins with the characters *merl*.

If you are not sure how the values in an index appear, you can BROWSE the index:

**BROWSE <index name> [optional value]**

For example,

**BROWSE idesc merl**

will show the values in the vicinity of MERLIN.

Indexes are updated immediately when a record containing indexed elements is added or updated.

A complete description of all the searching capabilities in SPIRES is described in the document Searching and Updating listed in Appendix G.

## VII.4 Updating Records.

A complete description of updating records in SPIRES is described in the document Searching and Updating listed in Appendix G. This brief summary provides an overview. See Appendix E for \$PROMPT format subcommands.

To update an INSTRUMENTS record, enter the following commands:

1. Determine the key of the record you wish to modify.(e.g., with FINd and TYPE commands)
2. DISplay <key> Recommended but not mandatory.
3. SET FORMAT \$PROMPT [cr]
4. MERGE <key> You will be prompted for a new values of existing elements and new occurrences of element that are multiply occurring . If you wish not to change a value, simply enter a [cr]. Otherwise, the text that you enter will become the new value.
5. DISplay <key> to verify that the record is correct.

Finally, it is always a good idea to retain your source documents after you complete any updating. In seven years of running SPIRES at LBL, no data has ever been lost, but users have forgotten why they changed some records.

After the update is finished, the current version of the record will be displayed by SPIRES. Indexes are updated immediately.

It is advisable to DISplay a record before you modify it.

**NOTE:** The key of a record (KEY.SYSTEM) cannot be modified by editing its value. To change the key of a record, please see Section VII.7.

Here's a sample session showing how to add a record (system responses in bold):

```
-?  
sel instruments (not necessary if INSTRUMENTS is already SElected)  
-?  
Clear format  
-?  
dis mx10  
INSTRUMENT.TYPE = MX10;  
INSTRUMENT.DESC = Merlin X10;  
INSTRUMENT.PRICE = $35.00;  
Structure NOTES(1)  
NOTE = The source document is in filed under Merlin.;  
NOTE.DATE = July 23, 1987;  
Date.Updated = July 23, 1987;  
-?
```

```

-?
set format $prompt
-?
MERGE mx10

INSTRUMENT.TYPE = MX10
INSTRUMENT.DESC = Merlin X10
INSTRUMENT.DESC: [CR]
INSTRUMENT.PRICE = $35.00
INSTRUMENT.PRICE: 42.00 [CR]
  Structure NOTES(1)
    NOTE = The source document is filed under Merlin.
:   NOTE [CR]
:   NOTE(2): This is a second note
:   NOTE(3): [CR]
-?
Clear format
-?
dis mx10 (or: /dis $key)

INSTRUMENT.TYPE = MX10;
INSTRUMENT.DESC = Merlin X10;
INSTRUMENT.PRICE = $42.00;
  Structure NOTES(1)
    NOTE = The source document is filed under Merlin.
    NOTE.DATE = July 23, 1987;
  Structure NOTES(2)
    NOTE = This is a second note;
    NOTE.DATE = July 24, 1987;
Date.Updated = July 24, 1987;

Notice that the Contact, Notes, and Date.Updated elements have changed.

```

## VII.5 Adding a new record.

A complete description of adding records in SPIRES is described in the document Searching and Updating listed in Appendix G. Below, a very brief summary and sample session provides an adequate overview. See Appendix E for \$PROMPT format subcommands.

### SET FORMAT \$PROMPT

#### ADD

You will be prompted for the value of each element. If an *optional* element should be left blank, enter a carriage return [CR]. Also note that you will be prompted twice for each multiply occurring element. Just enter a [CR] to proceed to the next element. Please see Appendix E for the subcommands used in the \$PROMPT format.

To add several records, simply reissue the ADD command after each previous record is ADDED and DISPLAYed.

After ADDing a new record, always DISPLAY <key> to examine it for correctness.

Here's a sample session showing how to add a record (system responses in **bold**):

```
sel instruments (not necessary if INSTRUMENTS is already SELECTed)
```

```
-?
```

```
set format $prompt
```

```
-?
```

```
ADD
```

```
INSTRUMENT.TYPE: mx10 [cr]
```

```
INSTRUMENT.DESC: Merlin X10 [cr]
```

```
INSTRUMENT.PRICE: 35. [cr]
```

```
Structure NOTES(1)
```

```
NOTE: The source document is filed under Merlin. [cr]
```

```
Structure NOTES(2)
```

```
NOTE: [cr]
```

```
-?
```

```
Clear format
```

```
-?
```

```
dis mx10 (or: /dis $key)
```

```
INSTRUMENT.TYPE = MX10;
```

```
INSTRUMENT.DESC = Merlin X10;
```

```
INSTRUMENT.PRICE = $35.00;
```

```
Structure NOTES(1)
```

```
NOTE = The source document is in filed under Merlin.;
```

```
NOTE.DATE = July 23, 1987;
```

```
Date.Updated = July 23, 1987;
```

## VI.6 Removing Records.

A complete description of REMoving records in SPIRES is described in the document Searching and Updating listed in Appendix G. Below, a very brief summary and sample session provides an overview.

To REMove a record from the INSTRUMENTS subfile, enter the command:

**REM <instrument.type key>**

For example, to REMove record mx10, enter:

**REM mx10 [cr]**

Note: If you wish to restore a record that was erroneously removed, please call for assistance.

**NOTE:** To safeguard indexing searching capability, SPIRES will not let you modify or remove any POINTER elements (in fact, they are not displayed in the INSTRUMENTS subfile), *nor can you remove a record that contains occurrences of POINTER.*

## VI.7 Changing the Key of a Record.

The key of a record (STATION) cannot be modified by editing its value and then issuing an UPDATE command. To change the key of a record, enter the following commands:

1. **CLEAR FORMAT**
2. **TRAnser <old INSTRUMENTS key> CLR**
3. **X ACTIVE FILE**
4. edit the INSTRUMENT.TYPE and other elements as needed using the XEDIT editor.  
(APPENDIX I describes use of the editor).
5. **FILE** on the command line at the bottom of the screen.
6. **ADD**
7. **REMOve <key of the old INSTRUMENTS record>** (as described in Sect. VII.6 above)

Please ask for human help if you encounter any difficulties (Appendix H).

## VIII. Using the BLDG Subfile

- 1 Description of elements in the BLDG subfile
- 2 Displaying records
- 3 Searching in SPIRES; Searching the BLDG subfile
- 4 Updating records
- 5 Adding new records
- 6 Removing records
- 7 Changing the key of a record

The BLDG Subfile is a goal-index record-type in the TPHHH:LBLSTAFF file. The purpose of the BLDG subfile is to maintain current descriptive information about each BLDG number, its building name, and its ICS SITE NODE identifier, as derived from personnel records in LBLSTAFF. This subfile acts as a goal record and as an index record to personnel records in LBLSTAFF.

For each record, a particular element may be required or optional, singly or multiply occurring, have controlled allowable values, be limited to a particular type of value, or be indexed for ease in searching, etc. The element listing below describes the characteristics of each element.

### VIII.1 Description of elements in the BLDG subfile

<u>Element Name</u>	<u>Required/Opt</u>	<u>Length</u>	<u>Occurrences</u>	<u>Data Type</u>	<u>Indexed</u>
BLDG (key of record)	REQ	Variable	Single	Text	
Pointer	Optional	fixed	Multiple	hex	
BLDGNAME (BN)	Optional	Variable	1	Text	
BLDGABBRV (BA)	Optional	Variable	1	Text	
ICSSITE (ICSS)	Optional	Variable	1	Text	

Controlled values:  
must be "1, 2, 3, 4, 5, or 6"

Graphically, the hierarchical nature of a typical BLDG record appears:

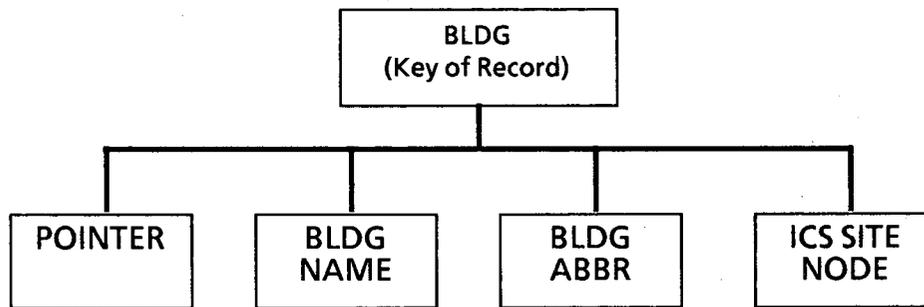


Figure 6

stored element

structure

virtual element

## VIII.2 Displaying Records.

A complete description of all the ways to display records in SPIRES is described in the document Searching and Updating listed in Appendix G.

To use the BLDG subfile, you must SElect it with the command:

**SElect BLDG**

If you select some other subfile, the you must again SElect BLDG to display BLDG records.

If you know the BLDG identifier for a record which you wish to see, you may use the DISplay command to view it directly:

**DISplay <BLDG key value>**

or, by using the FINd command, and TYPE:

**FINd <STATION key value>  
TYPE**

To put the results in your active file so that they may be printed or viewed, prefix the TYPE or DISplay commands with IN ACT CLR:

**IN ACT CLR DIS <STATION key value>  
IN ACT CLR TYPE**

Global for may be used to display groups of records. For example,

**FOR ADDS or FOR UPDATES  
DISplay ALL or IN ACT CLR DISplay ALL**

will display all the records added or updated that day.

Customized output formats may also be written or format \$REPORT may be used to display records in *other than* the default SPIRES output format, i.e.,

**<Element name> <equal sign> <element value> <semicolon>**

For example, a customized format might display all of the extensions, key systems, or instruments associated with a particular building, sorted by room number. Such a format does is not written with this version of the documentation.

## VIII.3 Searching in SPIRES; Searching the BLDG Subfile.

No indexes are defined for the BLDG subfile with this version of the documentation. Indexes could be built on ICS SITE NODE or on BLDGNAME if requested. Indirect indexes can be defined to search for EXT, STATION, or NAME (employee name) that return BLDG records as a result, although it is now possible to search those subfiles directly.

Global For processing may be used to search records in the BLDG subfile.

A complete description of all the searching capabilities in SPIRES is described in the document Searching and Updating listed in Appendix G.

#### VIII.4 Updating Records.

A complete description of updating records in SPIRES is described in the document Searching and Updating listed in Appendix G. This brief summary provides an overview. See Appendix E for \$PROMPT format subcommands.

To update a BLDG record, enter the following commands:

1. Determine the key of the record you wish to modify.(e.g., with FINd and TYPE commands)
2. DISplay <BLDG key value> Recommended but not mandatory.
3. SET FORMAT \$PROMPT -NORETURN [cr]
4. MERGE <BLDG key value> You will be prompted for a new values of existing elements and new occurrences of element that are multiply occurring . If you wish not to change a value, simply enter a [cr]. Otherwise, the text that you enter will become the new value.
5. DISplay <BLDG key value> to verify that the record is correct.

Finally, it is always a good idea to retain your source documents after you complete any updating. In seven years of running SPIRES at LBL, no data has ever been lost, but users have forgotten why they changed some records.

After the update is finished, the current version of the record will be displayed by SPIRES.

It is advisable to DISplay a record before you modify it.

**NOTE:** The key of a record (BLDG) cannot be modified by editing its value. To change the key of a record, please see Section VIII.7.

Here's a sample session showing how to add a record (system responses in bold):

```
-?  
sel bldg (not necessary if BLDG is already SElected)  
-?  
dis 1  
  
BLDG = 1;  
BLDGNAME = Donner Laboratory;  
BLDGABBRV = Donner;  
ICSSITE = 5;  
  
-?  
set format $prompt  
-?  
merge 1  
  
BLDG = 1  
BLDGNAME = Donner Laboratory  
BLDGNAME: [cr]  
BLDGABBRV = Donner  
BLDGABBRV: [cr]  
ICSSITE = 5  
ICSSITE: 4 [cr]  
-?
```

```
clear format  
-?  
dis 1 (or /dis $key)  
BLDG = 1;  
BLDGNAME = Donner Laboratory;  
BLDGABBRV = Donner;  
ICSSITE = 4;
```

### VIII.5 Adding a new record.

A complete description of adding records in SPIRES is described in the document Searching and Updating listed in Appendix G. Below, a very brief summary and sample session provides an adequate overview. See Appendix E for \$PROMPT format subcommands.

Note that SPIRES creates BLDG records whenever a new BLDG value is used in the LBLSTAFF subfile.

#### SET FORMAT \$PROMPT

#### ADD

You will be prompted for the value of each element. If an *optional* element should be left blank, enter a carriage return [CR]. Also note that you will be prompted twice for each multiply occurring element. Just enter a [CR] to proceed to the next element. Please see Appendix E for the subcommands used in the \$PROMPT format.

To add several records, simply reissue the ADD command after each previous record is ADDED and DISPLAYed.

After ADDing a new record, always DISPLAY <key> to examine it for correctness.

Here's a sample session showing how to add a record (system responses in **bold**):

```
sel BLDG (not necessary if BLDG is already SElected)
```

```
-?
```

```
set format $prompt
```

```
-?
```

```
ADD
```

```
BLDG: 1 [cr]
```

```
BLDGNAME: Donner Laboratory [cr]
```

```
BLDGABBRV: Donner [cr]
```

```
ICSSITE: 5 [cr]
```

```
-?
```

```
dis 1
```

```
BLDG = 1;
```

```
BLDGNAME = Donner Laboratory;
```

```
BLDGABBRV = Donner;
```

```
ICSSITE = 5;
```

### VIII.6 Removing Records.

A complete description of REMoving records in SPIRES is described in the document Searching and Updating listed in Appendix G. Below, a very brief summary and sample session provides an overview.

To REMove a record from the BLDG subfile, enter the command:

**REM <BLDG key>**

For example, to REMove record 522, enter:

**REM 522 [cr]**

Note: If you wish to restore a record that was erroneously removed, please call for assistance.

**NOTE:** To safeguard indexing searching capability, SPIRES will not let you modify or remove any POINTER elements (in fact, they are not displayed in the BLDG subfile), *nor can you remove a record that contains occurrences of POINTER*. It is unlikely that you will be able to remove any BLDG records unless they were added manually rather than created by SPIRES as a result of passing BLDG values from LBLSTAFF records.

### VIII.7 Changing the Key of a Record.

The key of a record (BLDG) cannot be modified by editing its value and then issuing an UPDate command. Further, it is unlikely that such an approach is desirable since BLDG records also serve as index records to LBLSTAFF. The proper procedure is to SElect LBLSTAFF, FINd BLDG <key of desired bldg>, change the BLDG values in the result, then SElect BLDG and merge BLDGNAME, ICSSITE values etc.

Please ask for human help if you encounter any difficulties (Appendix H).

**IX. Using the SERVICE subfile**

- .1 Description of elements in the SERVICE subfile
- .2 Searching in the SERVICE subfile
- .3 Updating, adding, deleting records

**IX.1 Description of elements in the SERVICE subfile.**

The SERVICE subfile is a subset of the LBLSTAFF database. SERVICE is used primarily by the telephone operators and mailroom personnel for online realtime retrieval of employee telephone extentions and mailstops. The data is maintained by the Telephone Services Department. The SERVICE subfile default format appears:

EMPLOYEE-NAME	BUILDING	EXTENTIONS	DATE LAST CHANGED
MAILSTOP	PAYROLL ACCOUNT NUMBER		DIVISION

**IX.2 Searching in the SERVICE subfile**

To use the SERVICE subfile, enter

**SELEct SERVICE**

Normal SPIRES searching commands are unnecessary in the SERVICE subfile when searching for employee names. When the subfile is SELEcted, the system responds with the prompt:

ENTER SEARCH STRING:

Simply enter a surname alone, the first part of a surname, or all or part of the given name and all or part of a surname.

E.g., to find Ernest O. Lawrence, any of the following search strings are valid:

LAWRENCE	E O LAWRENCE
LAWRE	E O LAW
LAW	O LAWREN
E LAWRENCE	ERN O LAW
E LAWREN	ERNEST O LAWRENCE
E LAW	

To exit the prompting routing, enter an asterisk:

ENTER SEARCH STRING: \*

When in the SERVICE subfile and exited from the automatic searching facility ("ENTER SEARCH STRING"), you may use normal SPIRES search commands such as SHOW ELEMEnts, SHOW INDExes, FIND, TYPE, and DISPlay.

To turn the automatic prompting back on for name searching, SELEct SERVICE.

**3. Updating, adding, deleting records.**

Updating, adding and deleting records in the SERVICE subfile is prohibited.

## X. Generating Reports.

Further information describing \$REPORT and a complete description of all the ways to display records in SPIRES is described in the document Searching and Updating listed in Appendix G and also outlined in the SPIRES SEARCHING PRIMER.

Occasionally you may wish to use some other format than the default SPIRES format. Customized formats can be provided as required, as described in previous sections. When any particular subfile is selected, the command SHOW FORMats will list these and also indicate if either one is currently in effect by notating " - SET" after the format name.

Simple reports can be defined by the user using the SET FORMAT \$REPORT command in any subfile. For example, in the STATIONS subfile,

```
SET FORMAT $REPORT STATION(1,5) FLAG(10,3) EXT(15,5)
```

would output search result in the form of a table with STATION ID beginning in column 1, the active flag beginning in column 10, and the EXT numbers beginning in column 15. If a STATION had more than one EXT, they would be written on the following lines, beginning in column 15.

**RULE:** If any virtual element is SET, then the format will not take effect. That is, presence of a virtual element overrides a format. Thus, in order to use a format, you should enter the command CLR ELEM beforehand. The same restriction applies to generating tables with the SET FORMat \$REPORT formats. Virtual elements must be cleared by CLR ELEM before the report will take effect. The report-generating EXECs listed below perform this automatically.

## XI Printing; the LPR, LPRCC, and LABEL commands

As mentioned in Section II, nearly any CMS file that the user has created can be printed using the LPR and LPRCC commands.

Staff may occasionally wish to print files other than standard reports. These may include a file created using the Xedit editor, or created by SPIRES as the result of a FIND or DISPLAY command. (SPIRES usually places search result displays and other output in the CMS file named ACTIVE FILE A or on the CRT or both.) Therefore, it will often be useful to be able to print files directly. There are two EXECs that will send files to the TALERIS printers on the first floor of Bldg. 50B in the Central Computing Facility machine room area. These EXECs are:

### LPR and LPRCC

The syntax of these commands is:

```
LPR <filename> <filetype> <filemode>
LPRCC <filename> <filetype> <filemode>
```

For example, to print the CMS file, ACTIVE FILE A, enter the command:

**LPR ACTIVE FILE A**

The distinction between the two is that LPRCC interprets any characters in the first column of the file (at the left margin) as carriage control (hence the CC; LPR is an acronym for line printer). Generally, users will not insert carriage control characters in a file, and so LPR is the appropriate command to use. However, the SPIRES facility **FORMAT\$REPORT** automatically reserve column 1 for carriage control characters, with data beginning in column 2. For files generated by these utilities, LPRCC should be used.

A file probably includes carriage control if most of the text begins in column two and column one contains characters such as: 1, 0, and +. For example, it may look something like:

```
1
MARY HAD A LITTLE LAMB
ITS FLEECE WAS WHITE AS SNOW
+
  WHITE AS SNOW
0
AND EVERY WHERE THAT MARY WENT
THE LAMB WAS SURE TO GO.
```

The LABEL command is used to print an existing file in label format. label format is:

Column one blank except for a "1" on the first line.  
No more than 40 characters per line.  
No more than eight lines per label.

It is advisable to check the label printer prior to sending a file to the label printer.  
So send a file do:

**LABEL filename filetype filemode**

E.g.,

**LABEL LABEL FILE A**

assuming that the file LABEL FILE A contains your label output.

Both label and laser printers are on the first floor of 50B in the User Area.

## APPENDIX A

### DIP-SWITCH SETTINGS FOR ADM-3A TERMINALS FOR USE ON SERIES I 3270 EMULATOR

INTERNAL					
	ON	OFF		ON	OFF
none			7	X	
6	X		6		X
5		X	5	X	
4	X		4		X
3		X	3		X
2		X	2		X
1		X	1		X

EXTERNAL		
	ON	OFF
Bit 8-0	X	
Parity	X	
STOP	X	
Data 7	X	
Parity		X
LC	X	
Auto NL	X	
RS232	X	
HDX		X
All speeds but 9600		X
9600	X	

## APPENDIX B

### Logging ON using an IBM PC and YTERM.

1. Turn PC, disk drive, and printer on and wait for the PC to complete booting.

2. Enter the command: **CD \YTERM**

**CONFIRM THAT YOU ARE USING THE CAMPUS VERSION OF YTERM AND NOT THE LBL ADMINISTRATION DIVISION VERSION.**

3. Enter the command: **X 7**

4. Enter the command: **T 9600 K**  
System responds: "PLEASE ENTER THE NAME OF THE KEYBOARD TABLE"

5. Enter: **UCBCAD**  
System responds by clearing the screen and displaying DISCONN in the lower left corner

6. Make sure the blue TSB box displays either a green or red light.

7. If red light is illuminated, press the blue button and wait for green light.

8. When green light is illuminated, enter carriage return [CR].

The following dialogue should occur. The system response is in **bold**. The user response in modern font.

9. Request: **ccdb** [CR].

10. System will respond with a bell, and cursor and will jump to next line. Enter carriage return [CR].

11. **YALE ASCII TERMINAL COMMUNICATIONS SYSTEM V2.1**  
enter terminal type: **YTERM** [CR].

12. System will respond with a pseudo-three-dimensional display CFO over the letters VM. Enter another [CR].

13. The screen will clear. Enter:  
**L** [name of your virtual machine] [CR].

14. **ENTER PASSWORD:**  
enter your password. It is not a good idea to write your password in this set of instructions. If you write it down, do so elsewhere.

Note: If your previous session ended "abnormally", e.g., by simply pushing the blue button on the TSB box to obtain a red light, you will have to enter, at this point in the logon procedure, the command: **IPL CMS** and then a [CR]. This should always be done when a paragraph beginning with the word "**RECONNECTED . . .**" appears.

15. Enter yet another [CR]. This causes your PROFILE EXEC to execute. The system will then perform the following tasks automatically. See Step 10 in Section II.1 of your Users Guide.

## APPENDIX C

### Using the UC 227.14 KERMIT with UCBCMSA

The UC 227.14 version of KERMIT can be used with IBM PC/XT's and must be used for IBM PC AT's that wish to do uploading and downloading of files. YTERM does not work with AT's.

#### To initiate a KERMIT session on the PC:

1. Change directory to this version of KERMIT. Distinguish it from other versions of KERMIT you might have on your PC. If it is in a directory called UCKERMIT, do: `cd \UCKERMIT` [CR].
2. Enter: `MSKERMIT` [CR].
3. Enter: `DO CMS` [CR].
4. Enter: `STAT` and confirm that `PARITY` is set to `EVEN` and `BAUD RATE` is 9600. If not, do: `SET PARITY EVEN` and/or `SET BAUD RATE 9600` as needed.
5. Enter: `C` [CR].
6. Make sure the blue TSB box displays either a green or red light.
7. If red light is illuminated, press the blue button and wait for green light.
8. When green light is illuminated, enter carriage return [CR].
9. Request: `ccdb` [CR].
10. System will respond with a bell, and cursor and will jump to next line. Enter carriage return [CR].
11. YALE ASCII TERMINAL COMMUNICATIONS SYSTEM V2.1  
enter terminal type: `KERMIT` [CR].
12. System will respond with a pseudo-three-dimensional display CFC over the letters VM. Enter another [CR].
13. The screen will clear. Enter:  
`L [name of your virtual machine]` [CR].
14. ENTER PASSWORD:  
enter your password. It is not a good idea to write your password in this set of instructions. If you write it down, do so elsewhere.

Note: If your previous session ended "abnormally", e.g., by simply pushing the blue button on the TSB box to obtain a red light, you will have to enter, at this point in the logon procedure, the command: `IPL CMS` and then a [CR]. This should always be done when a paragraph beginning with the word "RECONNECTED..." appears.

15. Enter yet another [CR]. This causes your PROFILE EXEC to execute. The system will then automatically enter SPIRES.

See Step 10 of section II.1 of your User's Guide.

To terminate a KERMIT session on the PC:

1. If you have one of the SPIRES prompts (-?, +?, ->, +>), enter: EXIT
2. The system will respond: Leaving SPIRES.
3. Enter: **LOG** [CR]. The TSB light will change from green to red.
4. Enter: **<control> - ]** (i.e., depress the CONTROL key, and while holding, depress the right-square bracket key).
5. Enter: **C**
6. Enter: **QUIT** [CR].

To download a file while logged on to UCBCMSA using KERMIT:

1. Enter: **KERMIT** [CR].
2. Enter: **SEND** fn ft fm [CR], e.g., **SEND ACTIVE FILE A** [CR].
3. Enter: **<control> - ]** (i.e., depress the CONTROL key, and while holding, depress the right-square bracket key).
4. Enter: **C**
5. Enter: **RECEIVE** [CR]. The file will then be received at the PC. When it is finished,
6. Enter: **<control> - ]** (i.e., depress the CONTROL key, and while holding, depress the right-square bracket key).
7. Enter: **CONNECT** [CR].
8. Enter: **<control> - G**
9. Enter: **QUIT** [CR]. You will now be in CMS/SPIRES with the normal prompt.

## APPENDIX D

### TERMINAL CONTROL

The SERIES/1 terminal controller commands are summarized in the document "Key Definitions for IBM 3277 Terminal Emulation", section, "ADM-3A Key Definitions for IBM 3277 Terminal Emulation" available from the Electronics Shop in 50B-2259 (see Allan Konrad). Other ASCII terminals such as the VT100 may be used as well. Each has its own key definitions which are summarized in the same document.

Occasionally, the system will not accept characters typed on the keyboard, but rather sound the "bell". To clear this keyboard lock, depress the CONTROL key and, while depressed, enter the letter sequence: RTXQV. This is notated

#### CNTL-RTXQV

When the system is displaying output on the CRT screen, it will stop after 22 or 23 lines, depending on the kind of terminal. The message **MORE** will be displayed at the lower right. At this point, one has four options:

1. Do nothing. After 50 seconds, the bell will sound. After an additional 10 seconds, the system will clear the screen and display the next page.
2. Enter **CNTL-Z**. This causes the next 23 lines to be displayed immediately.
3. Enter a [CR]. This causes the message in the lower right portion of the screen to change from **MORE** to **HOLDING**. The timer holds, and the screen will not change. Another [CR] causes the message in the lower right to return to **MORE** and the timer is reset.
4. Enter **HT** [CR], then **CNTL-Z**. The **HT** halts typing, preventing the rest of the lines from being displayed. The **CNTL-Z** then clears the screen.

Several helpful CMS terminal commands are available:

The pound sign (#) acts as a **LINEND** character (line end).

The double-quote (") acts as an **ESCAPE** character

The (@) acts as a **CHARDEL** (character delete) character.

The (ø) acts as a **LINEDL** character (line delete)

The (#) and the (") have been disabled as CMS control characters since they conflict with often-used SPIRES characters.

Series/1 - ADM3A terminal control commands (The complete list can be found in "ADM-3A Key Definitions for IBM 3277 Terminal Emulation" available from the Computer Center Library.):

<b>CNTL-N</b>	go to next line
<b>CNTL-H</b> (or left-arrow key)	move cursor to the left
<b>CNTL-L</b> (or right-arrow key)	move cursor to the right
<b>CNTL-K</b> (or up-arrow key)	move cursor up
<b>CNTL-J</b> (or down-arrow key)	move cursor down
<b>CNTL-D</b>	deletes a character (On PCs, use Delete key)
<b>CNTL-E</b>	deletes to the end of a line
<b>ESC-spacebar</b>	enter or leave <i>character</i> insert mode (on PCs, use Insert key)

These sequences work in the editor as well as outside the editor.

### Program Function (PF) keys

In some utilities, such as FLIST and Xedit, PF keys are assigned specific functions. When using an ADM-3A terminal, the PF keys are implemented as a sequence of two keys: the ESC key followed by some other key. For PF1 through PF9, use ESC 1 through ESC 9. ESC-: (colon) is equivalent to EXC-11, and deletes to the end of line in the FLIST facility. ESC-3 usually means "quit". ESC-1 usually calls a CMS help screen. Often a menu of valid PF keys will be displayed in utilities where they are recognized.

## APPENDIX E

### SPIRES FORMAT \$PROMPT Subcommands

The following commands are recognized by SPIRES when adding new records (or modifying existing records) using SET FORMAT \$PROMPT (formerly SET INPUT FORMAT):

[CR] (carriage return)	Continue to next prompt
//	Puts in a null-length value if legal, otherwise you are reprompted for a legal value.
/N	Skip to the next element of the current structure for input
/S	Skip to the next structure for input (first element of next structure)
/ <b>&lt;value&gt;</b>	Retains leading blanks (blanks in front of the value)
<b>&lt;value&gt; //</b>	Continue value on next line (for long values, e.g., paragraphs)
/E	End input for the current structure, and retain input thus far
/X	Abort input, and do not retain any input
/R	Removes occurrence of existing element value
/X	Removes occurrence of existing structure

Example of //:                   to enter a null value in a structure without exiting the structure, for example in the ALIEN structure:

```
STRUCTURE ALIEN
  ANAME: //
  JOB.TITLE president
  (other elements)
```

This prevents the other elements in the ALIEN structure from being skipped merely because there was no value entered for ANAME.

The full set of subcommands can be found in the SPIRES manual Searching and Updating.

## APPENDIX F

### Looking at your CMS files

The CMS FLIST facility provides a listing of your permanent files and several capabilities to browse, edit, copy, rename, and delete them. To use the FLIST facility, enter the command FLIST and your files will be displayed, with the cursor at the top of the list. You may move the cursor up and down to select any file. You may use the ESC commands on the menu at the bottom to perform various operations, e.g., ESC-4 or an X will invoke the editor on the selected file, an EXC-2 will allow you to browse the file, and ESC-8 will allow you to see the next screenful of files on your list if you have more files than can be listed on one screen, and ESC-3 will exit FLIST. All the terminal control keys work in FLIST.

There are other file listing facilities besides FLIST. FLIST currently provides the most functionality. For assistance with FLIST, please see Appendix H for human help.

## **APPENDIX G**

### **Documentation**

A complete set of SPIRES documentation is available by entering PERFORM PUBLISH. Those items most likely to be of interest are:

1. A Guide to Searching -- A SPIRES Primer.
2. Searching and Updating.
3. Sequential Record Processing: Global FOR Reference Manual.
4. SPIRES Keyterm Index -- An index of all SPIRES terms.

A complete set of CMS documentation is available from the Computer Center library. The following are most likely to be of interest to users of the AWARDS database system.:

1. System Product Editor User's Guide (SC24-5220-1)
2. System Product Editor Command and Macro Reference (SC24-5221-1)

The RTSG Electronics Shop in Bldg. 50B-2259 has copies of the following documents:

1. ADM-3A Key Definitions for IBM 3277 Terminal Emulation
2. VT100 Key Definitions for IBM 3277 Terminal Emulation

## **APPENDIX H**

### **Human Help**

For assistance, call:

Allan Konrad

x 5458

## APPENDIX I

### Using Xedit

The following describes use of Xedit with an ADM-3A terminal. For other terminals, please see Appendix D.

(Note: If you are using the Xedit editor and SPIRES, be aware that it is helpful to be in the same case mode in the editor as in SPIRES. That is, it is possible to be in SPIRES in upper-and-lower case, while in Xedit in upper only, or vice-versa. If you have problems with case, call for human help (Appendix H).)

Files in the VM/CMS system have three-part names:

filename filetype filemode

usually abbreviated

fn ft fm

The filemode is generally assumed to be A, referring to you "A-disk", 191. This 191 A disk is your private disk.

To edit a file, issue the command

**X fn ft**

For example, to edit the CMS file ACTIVE FILE A, enter

**X ACTIVE FILE A**

The document will then appear ready to edit. Case is not significant on this command. You could also enter:

**x active file a**

If the file ACTIVE FILE did not exist on your A disk, the editor would create a new empty file, with only a top-of-file and a bottom-of-file marker.

Once in the editor, you can:

Use the CNTL-D and CNTL-E keys (see Appendix D)

Use the "cursor" keys to move the cursor around on the screen. On an ADM3A terminal, depress the CONTROL key, and while holding it, press either H, J, K, or L depending on which direction you wish to move the cursor. After you release CONTROL, whatever characters you type will replace the text in your file, if any.

Use the prefix field on the left side of the screen (the five columns of equal signs) to copy, delete or move whole lines or groups of lines.

## Often-used Prefix-field Commands.

### D (delete)

To delete one line, place a **d** anywhere in the prefix field to the left of the line you wish to delete. Then hit [CR]. E.g.,

```
==== This is line one
==d== This is line two
==== This is line three
```

results in:

```
==== This is line one
==== This is line three
```

To delete a known number of contiguous lines, enter **d** and the number of lines to be deleted.

```
==== This is line one
==d2= This is line two
==== This is line three
==== This is line four
```

results in:

```
==== This is line one
==== This is line four
```

To delete an *unknown* number of contiguous lines, that is, a "block" of lines enter **dd** on the first line to be deleted and on the last line to be deleted. E.g.,

```
==== This is line one
==dd= This is line two
==== This is line three
dd== This is line four
==== This is line five
```

results in:

```
==== This is line one
==== This is line five
```

## I (insert)

To insert a new blank line that can be edited, place an **i** in the prefix field on the line which you want the new line to follow. E.g,

```
==== This is line one
==i== This is line two
==== This is line three
==== This is line four
```

results in:

```
==== This is line one
==== This is line two
====
==== This is line three
==== This is line four
```

The new blank line can now be edited by moving the cursor to anywhere to the right of the prefix field and the first blank column following it.

To insert a specified number of new blank lines that can be edited, place an **i** and the number of blank lines needed in the prefix field on the line which you want the new line to follow. E.g,

```
==== This is line one
==i3= This is line two
==== This is line three
==== This is line four
```

results in:

```
==== This is line one
==== This is line two
====
====
====
==== This is line three
==== This is line four
```

It is also possible to insert lines by entering the command **i** on the command line at the bottom of the screen. This will clear the screen below the column-counter line. You can then enter text and use **CNTL-N** to go to the next line. When you hit a **[CR]**, your text will be shifted up above the column-counter line and the lower part of the screen will be available for more input. Two consecutive **[CR]**'s will return you to normal edit mode.

## C (copy)

To copy one line, place a **c** anywhere in the prefix field to the left of the line you wish to copy and a **p** on the line before which the newly created line should be placed. E.g,

```
==== This is line one
==c== This is line two
====p This is line three
```

results in:

```
==== This is line one
==== This is line two
==== This is line two
==== This is line three
```

the **p** stands for *prior* and instructs the system to put the new copy of the line prior to the line with the **p**. You can use the **f** instead, which means *following*:

```
==== This is line one
==c== This is line two
==== This is line three
==f= This is line four
```

results in:

```
==== This is line one
==== This is line two
==== This is line three
==== This is line four
==== This is line two
```

To copy a known number of contiguous lines, enter **c** and the number of lines to be copied on the first line to be copied, and an **f** or a **p** to mark where the copied lines should be placed:

```
==== This is line one
==c2= This is line two
==== This is line three
==f= This is line four
```

results in:

```
==== This is line one
==== This is line two
==== This is line three
==== This is line four
==== This is line two
==== This is line three
```

To copy a *unknown* number of contiguous lines, that is, a "block" of lines, enter **cc** on the first line to be copied and on the last line to be copied, and an **f** or a **p** to mark where the copies should be placed:

```
=p= This is line one  
=cc= This is line two  
===== This is line three  
cc= This is line four  
===== This is line five
```

results in:

```
===== This is line two  
===== This is line three  
===== This is line four  
===== This is line one  
===== This is line two  
===== This is line three  
===== This is line four  
===== This is line five
```

**M** (move)

the move command, **m**, works similarly to copy:

```
===== This is line one  
=m= This is line two  
====f This is line three
```

results in:

```
===== This is line one  
===== This is line three  
===== This is line two
```

and,

```
=p= This is line one  
==mm= This is line two  
===== This is line three  
==mm This is line four
```

results in:

```
===== This is line two  
===== This is line three  
===== This is line four  
===== This is line one
```

Most terminals can only display about 22 lines of text. Therefore, if the file you are editing is longer than 22 lines, not all of them can be displayed simultaneously.

Think of your file as if it were a very tall building. The building is a strange building however, because its floors are numbered from top to bottom rather than from bottom to top! So the first floor is at the top of the building.

Our building has a rather unique elevator. Unquestionably the oddest thing of all is that the elevator doesn't move, the building does! The elevator is fixed, but the building moves up and down, into and out of the ground.

But that's not all! First, its doors are always open, so you can always see out as the building moves up and down in front of you. Furthermore, your elevator is 21 stories high! Stranger yet is that half-way up this tall elevator is a platform on which you stand. Thus, you can see the floor that is level with yourself, the 10 floors lower, and the 10 floors higher.

This peculiar building is like your file and your terminal is like its elevator which provides you with a view of some portion of of the building. Imagine standing in the fixed elevator as the building moves up and down in front of you. This is exactly the phenomenon you experience using the editor.

When you first enter the editor, it automatically gives you a view of the top 10 lines of your file. This is like standing in your elevator at the top of the building, with a view of the 10 floors beneath you and 10 stories of thin air above you.

If you wish to look at lower floors of the building, what would you do? You would command the building to shift up (which is equivalent to the elevator going down). This is exactly what you do in the editor. The following is a brief summary of the commands that you can use to move around in your file. They are entered on the command line at the bottom of your screen when you're in the editor.

**+5** shifts the file up 5 lines so that your view is the next 5 lines down. The "+" is optional. Just a 5 or any number is acceptable.

To adjust your view in the opposite direction, i.e., towards the top of the file, use a minus sign preceding the number of lines you want to shift, e.g., **-20** will display the portion of the file 20 lines above your current position.

The command **top** will go the the top of the file. The command **bot** will go to the bottom of the file.

When a number is preceeded with a colon, the editor will go directly to that absolute line number. E.g, **:104** would display lines 93 through 115, with line 104 exactly in the middle of the screen.

To locate a string of characters, enter a slash (/) and the character string to be searched for. It will locate the first instance of that string. If you want to search for later occurrences, continue entering equal signs (=) until you find the occurrence you desire.

Finally, the insert command, **i**, discussed above, is entered from the command line and allows you to insert a virtually infinite number of new lines at that point in the file.

It would not be useful to give every detail of the editor here. See Appendix G for a list of documents which describe how to use the editor. If you need assistance, please see Appendix H for human help.

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

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

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