

Research Technical Completion Report

**A SAS<sup>®</sup> BASED HYDROLOGIC  
INFORMATION STORAGE AND  
RETRIEVAL SYSTEM  
PART I**

by

Myron Molnau  
Mary Jo Bluske

Department of Agricultural Engineering



Idaho Water Resources Research Institute  
University of Idaho  
Moscow, Idaho 83843

July, 1986

*Permanently*  
PLOVER BOND

The research on which this report is based was financed in part by the United States Department of the Interior as authorized by the Water Research and Development Act of 1978 (P.L. 95-467).

Contents of this publication do not necessarily reflect the views and policies of the United States Department of the Interior nor does mention of trade names or commercial products constitute their endorsement by the U.S. Government.

Research Technical Completion Report

14-08-0001-G1014

A SAS<sup>®</sup> BASED HYDROLOGIC INFORMATION  
STORAGE AND RETRIEVAL SYSTEM

PART I

by

Myron Molnau  
Mary Jo Bluske

Department of Agricultural Engineering

Submitted to: U.S. Geological Survey  
United States Department of the Interior  
Washington, D.C. 20242

Idaho Water Resources Research Institute  
University of Idaho  
Moscow, Idaho 83843

July, 1986

TABLE OF CONTENTS

	Page
LIST OF FIGURES . . . . .	iii
ABSTRACT . . . . .	iv
I. INTRODUCTION . . . . .	1
II. BACKGROUND . . . . .	3
A. History of the Current Data Base . . . . .	3
B. Use of the Current System . . . . .	3
C. Design Choices . . . . .	4
III. SYSTEM REQUIREMENTS . . . . .	6
A. Choice of the SAS <sup>®</sup> Software System . . . . .	6
B. Human Interface . . . . .	7
C. Hardware and Software Requirements . . . . .	10
D. External Data Base Interface . . . . .	11
E. System Function Descriptions . . . . .	12
IV. FILE DESCRIPTIONS . . . . .	25
A. Justification of File Design . . . . .	25
B. Format of File Descriptions . . . . .	26
C. Detailed File Descriptions . . . . .	28
V. DATA RETRIEVAL FOR THE SAS PROGRAMMER . . . . .	44
A. Sample Data Retrieval Programs . . . . .	44
B. SAS <sup>®</sup> Programs to Generate Reports . . . . .	47
C. SAS <sup>®</sup> Program to Generate Statistics . . . . .	47
VI. IMPLEMENTATION . . . . .	57
A. User Point of View . . . . .	57
B. Programmer Point of View . . . . .	59
C. System Design Guidelines . . . . .	60
VII. FUTURE PLANS . . . . .	62
VIII. REFERENCES . . . . .	63

LIST OF FIGURES

Figure	Page
1. Data Flow Diagram . . . . .	14
2. Air Temperature Daily Data Listing . . . . .	15
3. SAS <sup>®</sup> Program for Data Retrieval by Station Number . . . . .	45
4. SAS <sup>®</sup> Program for Data Retrieval with Pointers . . . . .	46
5. SAS <sup>®</sup> Program to Generate Output with PROC PRINT . . . . .	48
6. SAS <sup>®</sup> Output with PROC PRINT . . . . .	49
7. SAS <sup>®</sup> Output with PROC PRINT . . . . .	50
8. SAS <sup>®</sup> Program to Generate Output with PUT Statements . . . . .	51
9. SAS <sup>®</sup> Output with PUT Statement . . . . .	53
10. SAS <sup>®</sup> Program to Calculate Statistics . . . . .	54
11. SAS <sup>®</sup> Output from PROC MEANS . . . . .	55
12. SAS <sup>®</sup> Output from PROC MEANS . . . . .	56
13. SAS <sup>®</sup> Program to Create a Personalized Pointer File . . . . .	58

## ABSTRACT

This paper presents a progress report on the development of a SAS<sup>®</sup> \* based information management system for the retrieval of hydrologic and climatic data. The new system will replace an existing system that has grown obsolete. A discussion is given of the current status of the project, and the remaining work to be accomplished. Descriptions of the overall design of the new system and of the SAS file structures are also presented. Also, examples are given of how to retrieve information from the new files.

\*SAS is the registered trademark of SAS Institute Inc., Cary, NC, USA.

## I. INTRODUCTION

A data management system called the Hydrologic Information Storage and Retrieval System (HISARS) is used by many individuals and organizations to retrieve data necessary in making water resource decisions. Currently maintained by the Department of Agricultural Engineering at the University of Idaho, HISARS is a PL/I based system created to give users easy access to hydrological information without requiring them to write sophisticated programs (Wiser, 1975; Molnau, 1975, 1983). However, due to the design of the system and the lack of PL/I programmers on campus, the maintenance of HISARS has become a difficult task. In particular, the addition of new data requires a significant amount of reprogramming whenever input formats are changed. Because PL/I is no longer taught on campus, student programmers cannot be found to perform such tasks, so faculty members must find time to do the maintenance programming. The result is that data entry is a slow process, often preventing users from getting the most up-to-date information available. Another problem with HISARS is the difficulty of modifying the system for new applications, such as creating new output formats or expanding the processing functions. Therefore, the purpose of this project is to replace HISARS with a SAS based storage and retrieval system called the Northwest Hydrologic Information Management System (NHIMS).

The objective of the NHIMS project is to create a hydrologic data management system that is easy to use, maintain, and modify, and at the same time simulates the actions of the existing HISARS system as much as possible. NHIMS will offer the same simple command language now used

in HISARS, which can be learned by all users, regardless of their programming skills. Furthermore, maintenance programmers will be able to easily write new input formats, speeding up data entry, as well as make enhancements to the system for new applications. A final objective, to be performed at project completion in 1987, is to conduct workshops for users on how to make use of the NHIMS system to retrieve hydrologic data.



## II. BACKGROUND

### A. History of the Current Data Base

The existing HISARS data base is organized as a group of individual ISAM files, stored on a mountable disk pack. There is a separate data file for each hydrologic element, and an index file associated with each data file. One index file contains additional information about each station in the main data file, such as the station's name, elevation, latitude, longitude, etc. For most files, one record contains one station-month of data. The record formats are either fixed or variable length; the particular format used depends on the type of hydrological data stored in that file.

Most of the HISARS files are quite large; for instance, the streamflow data file contains more than 136,000 records. Also new data are added to the files on an annual basis, so that the data base is constantly growing larger. The use of the ISAM file structure provides for efficient data retrieval with large files because records can be directly accessed by key. In HISARS, the key consists of the station identification number and the date.

### B. Use of the Current System

HISARS is used extensively by individuals, university classes, public agencies, and private companies from all over Idaho and the Pacific Northwest in order to retrieve the hydrologic data necessary in making water resource decisions. In 1985, there were 1,981 separate accesses of the system by these users, and in the previous year, 1984, over 2,000 accesses were made.

To retrieve information from HISARS, the user must issue a small number of access commands to identify the characteristics of the requested data to the system. The characteristics by which data can be retrieved are:

- hydrological element
- weather station
- county
- elevation
- drainage basin
- drainage area
- climatological region
- date (or range of dates)

Once specified to the system, the data can be simply printed out in a standard format, copied into a file for use by a program in another language, or processed in various ways. Although 17 different processing options now exist in HISARS, these options are limited in that most of them only operate on one type of hydrologic element. For instance, the HIGHEST process, which finds the highest temperature for each month in a range of data, cannot be used to find the largest monthly values in any of the other files.

### C. Design Choices

Once it had been decided to replace HISARS with another system, three main options were considered in choosing a replacement. One option would be to entirely rewrite the system in another programming language, preserving the ISAM file structure to allow direct access of the files. A second choice would be to purchase a prewritten data base management package and adapt it to satisfy the specific needs of HISARS users. Third, the SAS software system, a utility language which is already available at the University of Idaho, could be used to implement the programs necessary to manage the data base. The files

would have to be converted to permanent SAS data sets to allow efficient data retrieval by the SAS programs. This third choice is somewhat of a compromise between the other two in that all of the I/O is already present but the user interface must be programmed.

*Permanized*

PLOVER BOND

25% COTTON FIBER

U.S.A.

### III. SYSTEM REQUIREMENTS

#### A. Choice of the SAS<sup>®</sup> Software System

The SAS software system was chosen to be the basis for creating the NHIMS system because it can simplify maintenance programming, it can perform the necessary data management tasks and it is already commonly used by many departments at the University of Idaho. Studies have shown that programming in a problem-oriented language such as that provided by the SAS system can be done much more quickly than programming in a high level language such as PL/I or COBOL (Dietz and Cox, 1980). Thus, maintenance tasks can be performed more quickly on SAS programs and data sets. In addition, the SAS system provides good documentation features for the data sets, further easing the maintenance of the permanent files.

The USGS uses the SAS system as the basis of its Urban Hydrology Data Base (Driver and Mustard, 1984), which contains data similar in type to that stored in NHIMS. They chose SAS software primarily because of the ease with which it can be used to create and manage data sets (Doyle and Lorens, 1982). Built-in procedures, such as PROC DATASETS and PROC EDITOR, allow the maintenance programmer to quickly and easily update and edit files, with a minimum of programming. Also, SAS software provides for direct access of SAS data sets by use of the POINT option on the SET statement; with the POINT option, the NHIMS system will be able to simulate the current ISAM file operations for efficient data retrieval. Other examples of using the SAS system for data base management can be found in the annual proceedings of the SAS Users Group International (SUGI) (Nungesser and Olson, 1980; Finley and

Pannell, 1982).

SAS software is extensively used at the University of Idaho for both statistical analysis and data storage and retrieval. Both beginning and experienced programmers can make use of the SAS system to retrieve and analyze data. Because of the large number of SAS users on campus, it will be relatively easy to find and hire student programmers to perform many of the data entry and maintenance operations for the NHIMS system. Also, because the files are stored as permanent SAS data sets, SAS programmers will be able to access the data with their own programs, independently of the NHIMS system.

The SAS system is also widely used by organizations off campus. Such groups include the Army Corps of Engineers, the Bureau of Land Management, the Forest Service, the Soil Conservation Service, and the Agricultural Research Service. Thus, if these organizations request information from NHIMS, the data can be shipped to them, via tape, already in SAS files. The SAS data sets can be easily written to and read from the tapes with the PROC COPY procedure.

Finally, versions of the SAS system are available for many other computers besides an IBM mainframe, both for batch and timeshare use. The full language is not yet available for microcomputers, but should be within the next five years.

#### B. Human Interface

NHIMS will be a useful tool for those who need easy access to the hydrologic and climatic data stored in the system's data base. Extensive programming skills will not be needed to access the system; however, the user must learn to request information by supplying some

instructions in a specified format. Experienced SAS users will also have the ability to access the data outside of the NHIMS system, using their own SAS programs to retrieve the data and analyze it to suit their specific needs. Examples of such programs are given in Section V.

NHIMS will run on the batch system; therefore, users will have to know how to submit a batch job. Within a job, a user will have to code some job control language (JCL) instructions followed by commands to the NHIMS program. Only a few JCL statements will be necessary, with required statements to be given in the user's manual.

The NHIMS commands consist of a series of keywords and operands which will serve as input to the NHIMS system. These commands are identical with those of the current system so that a new language does not have to be learned in order to use NHIMS.

The following commands will be available with NHIMS:

ACCESS: signals the beginning of a group of commands, all of which deal with the same data.

ELEMENT: identifies the element for which data are requested.

The following elements are available:

TEMPERATURE  
DAILY PRECIPITATION  
HOURLY PRECIPITATION  
EVAPORATION  
SNOWFALL  
SNOW COURSE  
STREAMFLOW  
PEAKFLOW  
RESERVOIR STORAGE

STATION: identifies the station for which data are requested.

POINTER (new): identifies the range of observations for which data are requested.

COUNTY: retrieves data from stations in a certain county.

BASIN: retrieves data from stations in a certain drainage basin.

- REGION: retrieves data from stations in a certain climatic region.
- ELEVATION: retrieves data from stations within a given range of elevations.
- AREA: retrieves data from streamflow stations by the size of the drainage area in square miles.
- PERIOD: identifies a range of dates for which data are requested.
- LIST: directs the system to produce listings of the requested data.
- COPY: copies data to a file in another format, to be used with programming languages other than the SAS system.
- PROCESS: performs statistical analyses.

Some examples of how to code the NHIMS commands are given in the following paragraphs, in order to illustrate the simplicity of their use. The JCL statements have been omitted.

To obtain a printout of 1980 daily air temperature data for the Grangeville station (numeric station code 103771) the user will code the following commands:

ACCESS	TEMPERATURE
ELEMENT	103771
STATION	1/1980 TO 12/1980
PERIOD	DAILY
LIST	

If the user wants the same information for all stations in Latah County, the following statements will be needed:

ACCESS	TEMPERATURE
ELEMENT	LATAH
COUNTY	1/1980 TO 12/1980
PERIOD	DAILY
LIST	

To request an index of the information contained in the temperature data

file, the following commands are used:

ACCESS	
ELEMENT	TEMPERATURE
LIST	INDEX

The last example shows a good way to find out which stations exist, to learn the station code numbers, to identify the stations in a county, etc., in order to run subsequent jobs.

### C. Hardware and Software Requirements

Upon completion, the NHIMS system will be designed to run upon the University of Idaho's IBM 4341 mainframe computer. The data sets will be kept on a mountable disk pack to keep storage costs low; they could also be kept on tape, if necessary. With no modification, NHIMS could be used on the CMS timeshare system, but the added convenience does not seem to justify the cost.

Regarding software requirements, the NHIMS system will run as a batch job on the IBM 4341 mainframe, using the following support software:

- OS/VS1 operating system
- 82.4 version or higher of the SAS software package

It will be necessary to use the job control language (JCL) in order to communicate with OS/VS1 operating system. All of the code in NHIMS, including macros, should be compatible with future releases of the SAS software product. Furthermore, the NHIMS system will be compatible with the 82.3 version of SAS software which runs on the CMS operating system, permitting conversion of the system to CMS at a later time if desired.

Because of differences in the SAS system used on minicomputers such as Prime, Data General, or the DEC VAX, modifications would have to be



made to the NHIMS system if it were to be moved to these computers. The SAS data sets, however, should be transferable with very little, if any, changes. While it may be possible to move NHIMS to a microcomputer, the PC version of the SAS system does not contain some of the more desirable features that would make such a transfer easy at this time.

#### D. External Data Base Interface

The main purpose of the NHIMS programs is to manage several external files of hydrologic and climatic data collected from regional climatic and hydrologic stations. Initially, there will be 11 files stored as permanent SAS data sets on either a mountable disk pack, tapes, or the system disks. The data are organized so that one file corresponds to a particular hydrological element (a list of the elements is given in Section III.B). For most files, each record contains the station code number, the date, data quality flags and the data for one month; within each file, the records are sorted first by the station code number, and then the year and month.

For each of the main data files described above, there will be a file of pointers, or observation numbers, to be used within NHIMS to directly access the data in these large files. Each record of a pointer file will have:

- the station code number
- the observation number of the first record in the main data set pertaining to the station
- the observation number of the last record in the main data set pertaining to the station

Since the pointer file will specify the entire range of records which apply to each station, the NHIMS program will be able to directly

access the data for any station, eliminating the need for sequentially searching through the large files. The pointer files will have one record for each unique station that reports data for the given element.

In addition, three index files will have descriptive information on each station included in the main data files; the information in the index files includes the station name, county, longitude, latitude, elevation, hydrologic unit code, river basin, drainage area and climatic region. The index will supply textual information for the output reports, as well as allow the user to find out which stations are included in the main data files. Each station will require one record in the index file.

#### E. System Function Descriptions

The NHIMS system must read the commands supplied by the user, access the proper data, and perform the specified operations, printing the results in one or more output reports; in other words, NHIMS will perform the same data retrieval and processing functions as HISARS. The commands given by the user are of two types; first, the identifier commands, (ELEMENT, STATION, BASIN, REGION, COUNTY, ELEVATION, AREA, and PERIOD), control what and how much data are to be accessed. Second, the action commands, (LIST, COPY, and PROCESS), identify the actions to be performed on the data. Thus, NHIMS can produce a listing of either the numeric data from or information about the data collection sites, can copy the data to an external file, or can perform certain kinds of standard processing. The beginning of a single user request will be identified by an ACCESS command, followed by the identifier and action commands. Within one job, a user will be able to do multiple ACCESS requests.

This section will describe the purpose of each of the major functions to be performed by the NHIMS system. Figure 1 illustrates the functions in the form of a data flow diagram. Such a diagram is used to illustrate the flow of data through a software system, and is not to be confused with a flowchart, which depicts the flow of program control. The circles in the diagram represent transformations performed as the data flows through the system, while the arrows show the inputs and outputs to each transformation. The function numbers in the diagram correspond to the paragraphs in the subsequent sections which describe the functions. Also, an example of the type of listings currently generated by HISARS can be found in Figure 2. The output from NHIMS will produce reports in much the same format as shown in Figure 2.

#### III.E.1 ACCESS

The ACCESS function will read all of the NHIMS commands submitted by the user, and perform all initialization tasks. The action and identifier commands will be separated and stored for the duration of the job for later use by the program. If the user is only requesting a listing of one or more indexes, the ACCESS function must recognize that no data from the main files are to be accessed and pass this information on to the functions with which it interacts.

It is the responsibility of the ACCESS function to detect any errors made by the user in coding the commands and to generate appropriate error messages. If the coding errors prevent further processing, the job will be terminated at this point.

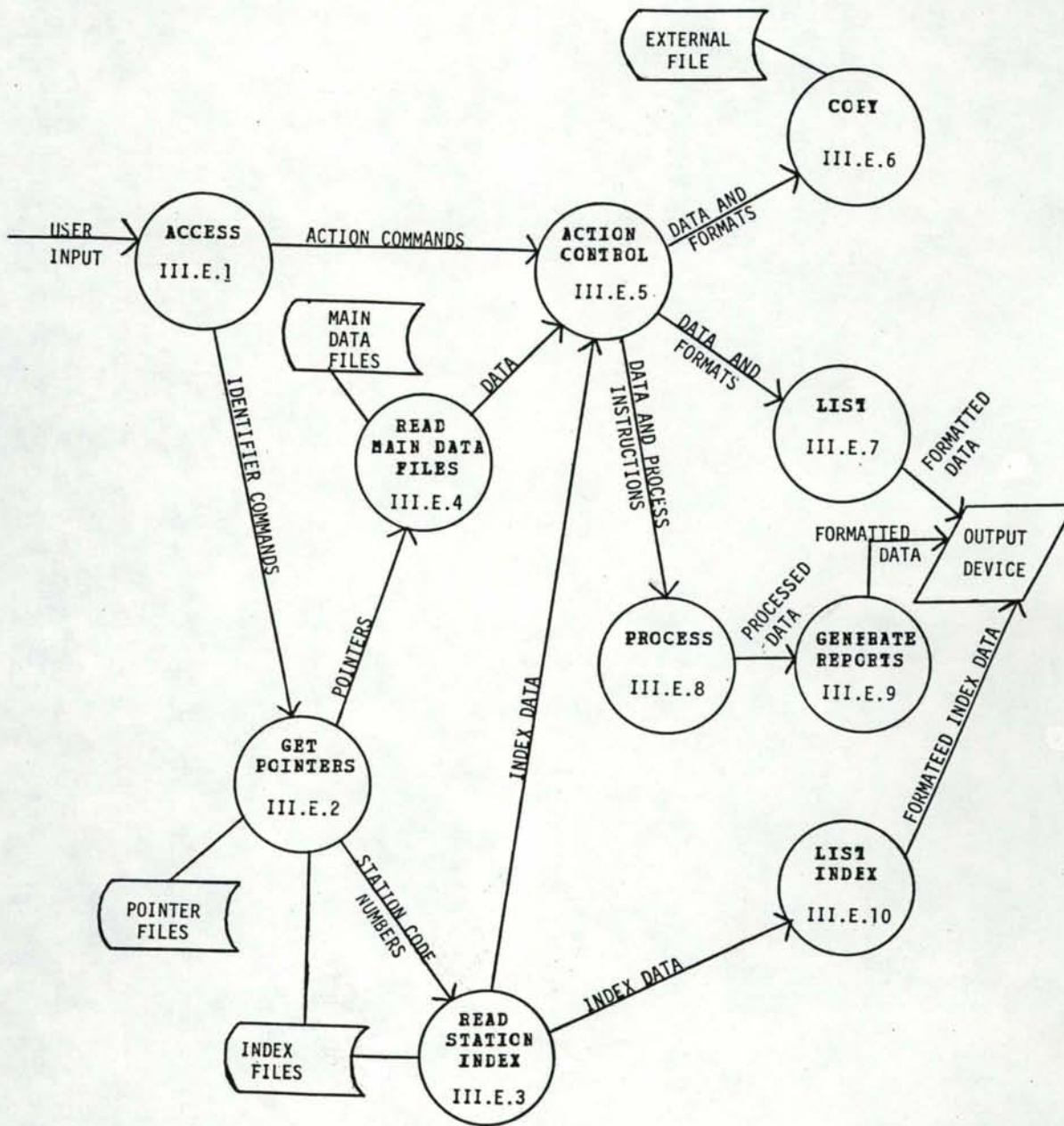


Figure 1. Data Flow Diagram

ABERLEEN EXP STATION

BINGHAM

10-0010

DAILY MAXIMUM AND MINIMUM TEMPERATURES IN DEGREES F.

1951

15

	JANUARY		FEBRUARY		MARCH		APRIL		MAY		JUNE		JULY		AUGUST		SEPTEMBER		OCTOBER		NOVEMBER		DECEMBER	
1	31	10	13	-12	32	8	51	28	49	31	58	36	88	47	91	61	78	38	73	40	38	15	46	26
2	30	24	23	6	34	18	60	25	57	30	57	33	84	49	88	61	78	39	64	44	42	7	43	27
3	37	26	37	21	32	15	64	25	68	28	63	32	94	44	82	61	78	40	59	40	48	16	30	26
4	33	17	39	22	39	22	63	29	77	42	74	33	88	55	78	61	80	40	53	39	55	22	34	20
5	30	16	40	34	38	20	65	26	80	34	63	46	91	44	81	53	83	44	59	38	50	17	36	24
6	23	7	40	32	40	13	65	27	78	47	63	34	89	42	85	52	84	48	64	29	51	13	27	16
7	20	-3	51	34	38	31	68	29	69	46	60	32	82	39	83	47	88	41	67	30	54	18	23	10
8	24	-1	47	34	38	22	64	31	57	40	67	31	83	40	80	45	82	54	73	32	55	33	13	-6
9	25	14	55	31	45	27	67	42	68	36	72	35	82	44	83	44	76	36	78	30	60	18	12	-13
10	29	10	55	31	31	11	60	31	73	35	76	37	79	50	87	45	85	37	79	33	57	24	19	0
11	33	14	57	33	32	9	58	18	72	38	80	39	72	43	93	49	77	44	69	42	54	32	23	16
12	29	2	46	30	37	16	68	22	55	31	69	54	82	41	85	49	65	29	58	38	43	32	25	19
13	31	19	38	20	37	26	74	28	55	38	75	46	97	46	83	49	75	32	58	36	40	25	24	20
14	42	25	39	18	47	29	72	32	63	27	81	44	97	50	80	44	81	34	66	31	40	30	23	-4
15	32	22	48	22	50	31	58	23	75	31	89	50	99	45	90	39	79	43	58	40	36	20	30	4
16	33	20	46	25	49	23	69	23	73	37	86	48	98	58	92	48	86	37	52	19	31	9	35	26
17	41	24	43	17	34	19	72	38	77	38	84	47	98	59	93	49	83	39	55	24	34	8	35	24
18	43	26	41	28	40	16	68	40	75	48	80	46	99	55	90	48	78	47	55	20	39	9	32	20
19	27	18	35	20	48	23	65	48	74	43	80	41	94	63	90	61	82	38	61	35	43	17	35	14
20	23	1	38	21	56	24	62	23	73	44	77	43	86	62	85	58	80	54	57	36	48	23	22	3
21	33	12	39	26	63	28	58	30	75	44	76	40	83	54	84	50	61	29	47	29	44	30	29	16
22	39	30	49	25	52	31	64	20	80	41	73	41	87	44	81	55	68	36	49	28	38	19	35	22
23	39	19	48	25	48	20	63	32	80	47	75	37	97	48	79	53	71	41	59	36	37	19	29	4
24	40	24	44	23	55	21	59	36	73	47	75	54	95	56	74	52	77	43	52	37	35	17	22	17
25	38	17	39	24	58	24	65	25	72	45	74	44	95	54	77	37	74	48	53	34	42	20	19	-5
26	42	15	38	9	55	35	63	43	82	45	82	40	95	58	83	40	68	44	55	25	43	24	25	1
27	39	30	31	21	50	28	67	36	87	54	81	41	85	58	81	44	78	27	55	24	45	27	38	23
28	30	-2	31	15	46	17	62	48	81	50	81	40	86	59	82	48	79	36	57	20	49	29	37	26
29	11	-12			54	20	49	33	75	35	79	45	82	58	72	47	75	46	60	27	54	25	37	29
30	11	-8			47	31	45	31	71	42	83	42	86	53	70	44	78	56	53	17	54	32	35	20
31	10	-8			41	30			63	39			94	53	74	41			43	18			20	10
AVEMAX	30.6		41.1		44.1		62.9		71.2		74.4		89.3		83.1		77.6		59.4		45.3		28.8	
AVEMIN	13.2		22.7		22.2		30.7		39.8		41.0		50.7		49.5		40.7		31.3		21.0		14.0	

Figure 2. Air temperature daily data listing.

### III.E.2 GET POINTERS

The GET POINTERS function is responsible for finding the pointers needed to directly access the information in the main data files. If station code numbers are explicitly given in the NHIMS commands, then GET POINTERS must simply reference the pointer files to get the corresponding range of pointers; if no station numbers are given, in other words, if the user requests information by county, basin, or one of the other identifier commands, then GET POINTERS must first search the index file for the proper station numbers before reading the pointer files to obtain the necessary pointers. When the user requests a listing of the index and no other information, GET POINTERS must recognize that no data from the main data files are to be accessed.

### III.E.3 READ STATION INDEX

After receiving the station code numbers from GET POINTERS, READ STATION INDEX must access the index information about those stations from the proper station index file, for use in the headings of the output reports to be generated. Only a subset of all the information for each station will be retrieved for these reports. However, if the user requests a listing of the index itself, all index information concerning the specified stations will be retrieved.

### III.E.4 READ MAIN DATA FILES

After receiving the range of observation numbers, READ MAIN DATA FILES must access the proper records from the main data files, and store them in a temporary data set.

### III.E.5 ACTION CONTROL

The ACTION CONTROL function uses the commands to decide which type of action is to be taken - COPY, LIST, or PROCESS. The inputs from both the main data and index files will then be made available to the necessary functions. User-supplied operands on the action commands will identify detailed formatting or processing instructions.

### III.E.6 COPY

The COPY function receives the data from the ACTION CONTROL function and converts it to a new format which is then written to either a permanent or temporary file, as specified in the JCL statements. The intent is to allow the user to get a copy of the data in a specific format, so that it can be accessed by programs written by the the user. The user will be able to create his own formats and design SAS data sets, as well as have the current file formats and 80-column output. There will be specific formats available for downloading to micros.

### III.E.7 LIST

The LIST function puts the data into a specific format for the output report listings. The listings can be printed in daily or monthly form. In addition, if requested by the user, the LIST function must calculate means and totals of the data by year or month, and include this information in the final report. Any missing, estimated, or accumulated data must be identified in the report by placing a symbol (character) next to the corresponding data item. Footnotes at the bottom of the report will explain the meaning of all such symbols.

An example of an output report currently produced by the LIST function in HISARS can be seen in Figure 2. The type of report produced by

the NHIMS system will be very similar to the example in Figure 2.

### III.E.8 PROCESS

The PROCESS function will perform several types of analyses using the data passed to it by the ACTION CONTROL function. The operands of the PROCESS command will serve as the processing instructions, identifying which type of analysis is to be performed. In this section, each type of analysis is described as a separate subfunction of the PROCESS function. All of the subfunctions will have the same output, which is the processed data. However, they differ slightly in the types of data that can be used as input. Therefore, the following subfunction descriptions will identify both the purpose of the function and the element files which it will process.

All of the following functions are currently available in HISARS. Numerous others will be available to the SAS programmer through the use of SAS PROCs.

#### III.E.8.a DAILY STATISTICS or MONTHLY STATISTICS

The STATISTICS function calculates the following items for each month of data:

- number of observations
- proportion of zeros
- mean
- standard deviation
- variance
- third moment
- fourth moment
- serial correlations

DAILY STATISTICS performs the analysis using daily values; MONTHLY STATISTICS performs the analysis using monthly total values, using a serial correlation between months and giving the number of pairs used for calculation.



The input files for the STATISTICS function are all element files except the peakflow file.

#### III.E.8.b DAILY FREQUENCY or MONTHLY FREQUENCY

The FREQUENCY function calculates the same items as the STATISTICS function, plus:

- maximum value
- upper and lower quartile
- upper and lower decile
- skewness
- kurtosis

Also included in the report is a plot of the cumulative frequency distribution of the data. A separate report will be generated for each month of data, as well as one for each entire year.

The input files for the FREQUENCY function are all element files except the peakflow file.

#### III.E.8.c HIGHEST or MAXIMUM

The HIGHEST function will search the data by month and find the highest value recorded for each month. For files that have two categories of data items recorded per day, the two highest (or maximum) values will be determined, one for each category. For instance, the air temperature file has both a maximum and minimum temperature recorded per day; this function will find the highest maximum and the highest minimum temperature for a given month. An option will be to have the corresponding dates of occurrence returned along with the highest values.

The HIGHEST function will operate on any of the files.

#### III.E.8.d LOWEST or MINIMUM

The LOWEST function will search the data by month and find the lowest value recorded for each month. For files that have two categories of data items recorded per day, the two lowest (or minimum) values will be determined, one for each category. For instance, the air temperature file has both a maximum and minimum temperature recorded per day; this function will find the lowest maximum and the lowest minimum temperature for a given month. A new option will be to have the corresponding dates of occurrence returned along with the lowest values.

The LOWEST function will operate on any of the files.

#### III.E.8.e EXTREME

The EXTREME function will search the data by month and find both the highest and lowest value recorded for that month. For files that have two categories of data items recorded per day, the most extreme values will be determined. For instance, the air temperature file has both a maximum and minimum temperature recorded per day; this function will find the highest maximum and the lowest minimum temperature for a given month. An option will be to have the corresponding dates of occurrence returned along with the extreme values.

The EXTREME function will operate on any of the element files.

#### III.E.8.f RANK ORDERING

The RANK ORDERING function searches the input data and finds the n largest amounts and their corresponding dates, where n is five times the number of years of data to be searched. For instance, if the user requested a rank ordering of 10 years of data, the largest 50 values in that 10-year period would be found, along with each of the dates of

occurrence. The n values are sorted into descending order, to be output from largest to smallest.

The RANK ORDERING function will operate on any of the element files.

#### III.E.8.g MONTHLY MASS FLOW ANALYSIS or MONTHLY MASS ANALYSIS

The MONTHLY MASS ANALYSIS function creates a plot of accumulated data through time. This function will use the streamflow and precipitation files as input.

#### III.E.8.h MAXIMUM FLOW ANALYSIS or MINIMUM FLOW ANALYSIS

The MAXIMUM FLOW or MINIMUM FLOW ANALYSIS function finds the average maximum or minimum streamflow for each year in the specified period of record. For maximum flow analysis, years are defined to be water years (October - September). For minimum flow analysis, years are defined to be climatic years (April - March).

The MAXIMUM or MINIMUM FLOW ANALYSIS function uses only the streamflow file as input.

#### III.E.8.i OCCURRENCES

The OCCURRENCES function calculates the number of days that data values from either the streamflow, temperature, precipitation, or snowfall files were equal to or exceeded a certain threshold amount during a single year.

For the temperature and precipitation data, the output includes the average number of days each threshold was exceeded. For streamflow data, the result is a flow duration table. For snowfall, the output contains the number of days that the snowfall was equal to or exceeded a threshold amount during the snow season (September 1 to August 31), and the dates of the first and last snowfall of the threshold amount are determined.

### III.E.8.j FLOW DURATION CURVE

The FLOW DURATION CURVE function produces a plot of the duration curve computed in Section III.E.8.i. The classes are plotted against the percent of time that the flow is equal to or greater than the lower bound of the class.

This function uses only the streamflow file.

### III.E.8.k CORRELATION

The CORRELATION function calculates the cross correlations between daily values for each pair of up to 10 stations. The results are determined separately for each month of the year.

All files can be used with CORRELATION.

### III.E.8.l HIGH OCCURRENCES

The HIGH OCCURRENCES function calculates the number of days that the maximum temperature was above a threshold for each complete year, and the first and last dates of occurrence. The user can change the default threshold.

The HIGH OCCURRENCES function uses only the air temperature file as input.

### III.E.8.m LOW OCCURRENCES

The LOW OCCURRENCES function calculates the number of days that the minimum temperature was below a threshold amount for each complete year, and the first and last dates of occurrence. The length of the period in days that the temperature was always equal to or greater than the threshold is also listed. The default threshold can be changed by the user.

The LOW OCCURRENCES function uses only the air temperature file as input.

### III.E.8.n FLOOD

The FLOOD function performs a frequency analysis of peak flow data using either the Log Pearson type III, Normal or the Log Normal statistical method. The data can be read directly from the PEAKFLOW file or data may be given by the user, bypassing the PEAKFLOW file.

This function uses the peak flow and snow course files.

### III.E.8.o CALENDAR

CALENDAR is a new function to be added to the system; it finds the necessary historical precipitation and air temperature data to be included in a monthly report in calendar form. For each calendar day of the year, the CALENDAR function must be able to find:

- the average maximum temperature
- the average minimum temperature
- the average precipitation
- the highest maximum temperature, and year it was recorded
- the lowest minimum temperature, and year it was recorded
- the highest precipitation value, and year it was recorded

The CALENDAR function will use the temperature and precipitation files for input.

### III.E.8.p SUMMARY

The SUMMARY function produces a monthly summary of the requested data patterned after the summary that appears in the Climatological Data, Annual Summary published by the National Climatic Data Center.

The SUMMARY function will use the temperature, evaporation, snowfall, and precipitation files as input.

### III.E.9 GENERATE REPORTS

GENERATE REPORTS will reformat the data obtained from the PROCESS function, in order to produce one or more output reports listing that

data. Given the data and the type of processing performed, the GENERATE REPORTS function determines what type of output report must be created and its format. As in the LIST function, any missing, estimated, or accumulated data must be identified in the report by placing a symbol (character) next to the corresponding data item. Footnotes at the bottom of the report will explain the meaning of all such symbols.

#### III.E.10 LIST INDEX

LIST INDEX receives information from the station index file, formatting the data to produce output listings. The report produced by LIST INDEX does not contain any data from the main data files; it is simply a listing of the requested index information. One paragraph of index information will be printed for each station specified; if the user requests information on multiple stations, the paragraphs will be printed out sequentially, in ascending order according to the station code number. The index information may be retrieved by county, basin, or any of the other identifier commands.

#### IV. FILE DESCRIPTIONS

##### A. Justification of File Design

In the design of the NHIMS data base, emphasis was placed on maintainability, efficiency of data retrieval, and conservation of disk space. To enhance maintainability, the individual data sets were made as uniform in structure as possible; the same variable names were used repeatedly in the different files to refer to the same types of information (STATION, YEAR, MONTH, etc). Consistency in the file structures will facilitate the use of SAS procedures for data set management, and will allow more general-purpose programs to be written. Meaningful variable names were created to represent the data elements, and extensive use was made of the LABEL statement to document the contents of the files.

Due to the large size of the files, it is important that the NHIMS programs be able to access directly the necessary data. By storing the files as permanent SAS data sets, the direct access facility provided by the SAS system can be used to retrieve data more efficiently. Creation of the pointer files, which keep track of where information on each station is stored in the main data set, allow use of the POINT option on the SET statement to simulate the efficiency of ISAM file operations. In addition, the organization of the files by element was maintained because the majority of user requests involve the retrieval of data for one particular element. With separate element files, the amount of data to be scanned in searching for the desired records is kept to a minimum, increasing the efficiency of data retrieval.

Conservation of disk space is essential when dealing with large data sets. To make the files smaller, unnecessary items were omitted when creating the new versions for the NHIMS system. For example, the value for the number of days per record was not included in any of the new files. Also, in order to minimize the amount of storage for individual data items, the LENGTH statement was used to decrease the default length of 8 bytes for SAS numeric values; 8-byte precision was not needed, since most of the numbers are, or can be stored as, integers. Furthermore, in some of the files, real numbers are adjusted by a factor of 10 or 100 so that the data could be stored as integers, which use a smaller number of bytes. These numbers must be readjusted to produce the correct data values.

#### B. Format of File Descriptions

The external files to be managed by NHIMS have been converted to permanent SAS data sets and are now ready for use. Each file contains data about a single climatic or hydrologic element; the one exception is the monthly file, which has monthly summaries of several elements. For most files, one record contains the data for a single month; the exceptions are the snow course and peak flow files, which have one year per record, and the hourly precipitation file, which has one day per record. In addition, there are three index files which have textual information on each station in the data base. All records are of fixed length, and 4 bytes of each record are reserved for internal use by the SAS system itself.

In the following file descriptions, each subsection begins with the name of a SAS data library, which contains all data sets relating



to a particular element. The library name is used as the operand for the DSN parameter in the JCL DD statement. For example, to access streamflow data, first identify the proper library to the operating system:

```
//ddname DD DSN=NHIMS.STREAM,DISP=SHR
```

After each library name, all corresponding SAS data sets are named and described. The SAS data set name is used in conjunction with the ddname to refer to a particular file within the SAS program. For example, to access the streamflow data, use the data set name MAINDATA in a SET statement:

```
SET ddname.MAINDATA;
```

Most of the libraries contain two data sets: one with the main climatic or hydrologic data (MAINDATA), and one with observation numbers (POINTERS) which can be used with the POINT option of the SET statement to directly access values from the main file. (For more information on using the POINT option, refer to the SAS User's Guide: Basics.) For each station in the main file, the corresponding POINTERS data set identifies the range of observation numbers containing data from that station; the number of observations in the POINTER file equals the number of unique stations in the main data file. Once a data set has been named in a SET statement, all values are accessed simply by using the individual variable names, as given in the file descriptions.

Each data set description has additional notes which name the units of all numerical values, explain any codes, and identify which numbers must be adjusted on input, because some numerical values are

scaled up in order to be stored as integers. All illegal end-of-month values, such as February 31, are stored as missing. The PROC CONTENTS procedure can also be used to produce data set descriptions.

Descriptions of the hourly precipitation file and the index files have been omitted because they are still in the process of being converted to SAS data sets. File descriptions will be available as soon as the conversion process has been completed.

### C. Detailed File Descriptions

#### IV.C.1 NHIMS.STREAM

##### IV.C.1.a SAS Data Set Name - MAINDATA.

Number of Observations - 136,828.

Number of Bytes/Observation - 146.

Total Length of File in Bytes - 19,976,888.

#### RECORD DESCRIPTION

<u>FIELD</u>	<u>VARIABLE NAME</u>	<u>TYPE</u>	<u>LENGTH</u>
Station identification	STATION	CHAR	8
Calendar year	YEAR	NUM	3
Month	MONTH	NUM	2
Monthly total discharge X 100	MONTHTOTAL	NUM	5
31 daily values: mean discharge X 100	STREAM1-STREAM31	NUM	4 each

#### NOTES

1. The 31 daily values and the monthly total are adjusted in order to be stored as integers; to get the true data values, each number must be multiplied by 0.01.
2. All missing data are stored as a single period (.).
3. The units for discharge are cubic feet per second (cfs).

IV.C.1.b SAS Data Set Name - POINTERS.

Number of Observations (Stations) - 569.

Number of Bytes/Observation - 20.

Total Length of File in Bytes - 11,380.

RECORD DESCRIPTION

<u>FIELD</u>	<u>VARIABLE NAME</u>	<u>TYPE</u>	<u>LENGTH</u>
Station identification	STATION	CHAR	8
First observation	FOBS_PTR	NUM	4
Last observation	LOBS_PTR	NUM	4

IV.C.2 NHIMS.PEAKS

IV.C.2.a SAS Data Set Name - MAINDATA.

Number of Observations - 17,249.

Number of Bytes/Observation - 31.

Total Length of File in Bytes - 534,719.

RECORD DESCRIPTION

<u>FIELD</u>	<u>VARIABLE NAME</u>	<u>TYPE</u>	<u>LENGTH</u>
Station identification	STATION	CHAR	8
Water Year of Peak Flow	WAT_YEAR	NUM	3
Month of Peak Flow	MONTH	NUM	2
Day of Peak Flow	DAY	NUM	2
Calendar Year of Peak Flow	YEAR	NUM	2
Gage Height X 100	GAGEHT	NUM	3
Annual Peak Flow	PEAKFLOW	NUM	4
Remarks Code for Gage Height	GH_CODE	CHAR	1
Remarks Code for Peak Flow	PK_CODE	CHAR	1
Remarks Code for Regulation or Diversion	RD_CODE	CHAR	1

NOTES

1. The values for gage height are adjusted in order to be stored as integers; to get the true data values, each number must be multiplied by 0.01.

2. Missing numeric data are stored as a single period (.); missing or non-applicable character data are stored as blanks.
3. The units for peak flow are cubic feet per second; the units for gage height are feet.
4. For the gage height remarks code:
  - a '1' indicates that the gage height was due to backwater; a 'BW' message should be printed.
  - a '2' indicates that the gage height was not the maximum for the water year; an 'NM' message should be printed. Also, the record immediately following contains information on the maximum gage height value for the same water year.
5. For the peak flow remarks code:
  - a '1' indicates that the value given is a maximum daily; an 'MD' message should be printed.
  - a '2' indicates that the discharge is estimated from information at another site; an 'ES' message should be printed.
  - a '3' indicates that the maximum was due to dam failure; a 'DF' message should be printed.
  - a '4' indicates that the actual discharge is less than the indicated value; a 'LT' message should be printed.
6. For the regulation and diversion remarks code:
  - a '1' indicates an unknown effect of regulation or diversion; a 'UR' message should be printed.
  - a '2' indicates a known significant effect of regulation or diversion; a 'KR' message should be printed.

IV.C.2.b SAS Data Set Name - POINTERS.

Number of Observations(Stations) - 730.

Number of Bytes/Observation - 18.

Total Length of File in Bytes - 13,140.

RECORD DESCRIPTION

<u>FIELD</u>	<u>VARIABLE NAME</u>	<u>TYPE</u>	<u>LENGTH</u>
Station identification	STATION	CHAR	8
First observation	FOBS_PTR	NUM	3
Last observation	LOBS_PTR	NUM	3

### IV.C.3 NHIMS.RESVOIR

#### IV.C.3.a SAS Data Set Name - MAINDATA.

Number of Observations - 7,618.

Number of Bytes/Observation - 145.

Total Length of File in Bytes - 1,104,610.

#### RECORD DESCRIPTION

<u>FIELD</u>	<u>VARIABLE NAME</u>	<u>TYPE</u>	<u>LENGTH</u>
Station identification	STATION	CHAR	8
Calendar year	YEAR	NUM	3
Month	MONTH	NUM	2
Code for time of day	TIMECODE	NUM	3
Code for data units	UNITCODE	CHAR	1
31 values: daily reservoir storage	STORGE1-STORGE31	NUM	4 each

#### NOTES

1. The 31 daily values may or may not need to be adjusted, depending on the value of the units code:
  - If UNITCODE = 'A', then the units are in acre-feet, and no adjustment of the daily values is necessary.
  - If UNITCODE = 'E', then the units are in feet of elevation above sea level, and the daily values must be multiplied by 0.01 on input.
  - If UNITCODE = 'S', then the units are in feet above datum, and the daily values must be multiplied by 0.01 on input.
2. All missing data are stored as a single period (.).
3. The units for the time code variable are hours in 24-hour time and in the time of observation.

#### IV.C.3.b SAS Data Set Name - POINTERS.

Number of Observations(Stations) - 56.

Number of Bytes/Observation - 18.

Total Length of File in Bytes - 1,008.

RECORD DESCRIPTION

<u>FIELD</u>	<u>VARIABLE NAME</u>	<u>TYPE</u>	<u>LENGTH</u>
Station identification	STATION	CHAR	8
First observation	FOBS_PTR	NUM	3
Last observation	LOBS_PTR	NUM	3

IV.C.4 NHIMS.EVAP

IV.C.4.a SAS Data Set Name - MAINDATA

Number of Observations - 2,167.

Number of Bytes/Observation - 271.

Total Length of File in Bytes - 587,257.

RECORD DESCRIPTION

<u>FIELD</u>	<u>VARIABLE NAME</u>	<u>TYPE</u>	<u>LENGTH</u>
Station identification	STATION	CHAR	6
Calendar year	YEAR	NUM	3
Month	MONTH	NUM	2
Total monthly wind movement	WIND_TOT	NUM	3
Total monthly evaporation X 100	EVAP_TOT	NUM	3
Monthly summary of codes	MONTHSUM	NUM	2
31 daily flags for estimated or accumulated wind values	W_CODE1-W_CODE31	CHAR	1 each
31 daily flags for estimated or accumulated evaporation values	E_CODE1-E_CODE31	CHAR	1 each
31 daily wind movement values	WIND1-WIND31	NUM	3 each
31 daily evaporation values X 100	EVAP1-EVAP31	NUM	3 each

NOTES

1. The 31 daily evaporation values and the total monthly evaporation are adjusted to be stored as integers; to get the true data values, each number must be multiplied by 0.01.
2. Missing numeric data are stored as a single period (.).
3. The units for evaporation are inches; the units for wind movement are miles.

4. If non-blank, the daily codes for wind and evaporation identify the following special conditions:
  - An 'A' means the corresponding daily value is accumulated
  - An 'E' means the corresponding daily value is estimated
  - A 'B' means the corresponding daily value is both estimated and accumulated
  
5. The monthly summary variable summarizes the information in the daily code variables. The summary is stored as an integer number, which is used to flag special conditions during the month. The meanings of the flags are:
  - 128: Wind movement values missing during month
  - 32: Wind movement values accumulated during month
  - 16: Wind movement values estimated during month
  - 8: Evaporation values missing during month
  - 2: Evaporation values accumulated during month
  - 1: Evaporation values estimated during month

If more than one condition is flagged, the value of MONTHSUM will be the sum of all the applicable conditions.

IV.C.4.b SAS Data Set Name - POINTERS.

Number of Observations(Stations) - 16.

Number of Bytes/Observation - 16.

Total Length of File in Bytes - 256.

RECORD DESCRIPTION

<u>FIELD</u>	<u>VARIABLE NAME</u>	<u>TYPE</u>	<u>LENGTH</u>
Station identification	STATION	CHAR	6
First observation	FOBS_PTR	NUM	3
Last observation	LOBS_PTR	NUM	3

IV.C.5 NHIMS.AIRTEMP

IV.C.5.a SAS Data Set Name - MAINDATA.

Number of Observations - 70,204.

Number of Bytes/Observation - 140.

Total Length of File in Bytes - 9,828,560.

RECORD DESCRIPTION

<u>FIELD</u>	<u>VARIABLE NAME</u>	<u>TYPE</u>	<u>LENGTH</u>
Station identification	STATION	CHAR	6
Calendar year	YEAR	NUM	3
Month	MONTH	NUM	2
Flag for missing/estimated data during month	MONTHSUM	CHAR	1
31 maximum daily temperatures	MAXTMP1-MAXTMP31	NUM	2 each
31 minimum daily temperatures	MINTMP1-MINTMP31	NUM	2 each

NOTES

1. Missing numeric data are stored as a single period (.).
2. The units for temperature are degrees Fahrenheit.
3. The monthly flag identifies special conditions for that month's data. If non-blank, its possible values are:
  - An 'A' means there were estimated data as well as missing maximums and minimums (ALL)
  - An 'M' means there were missing maximums and minimums
  - An 'X' means there were missing maximums
  - An 'N' means there were missing minimums
  - An 'E' means there were estimated data
  - An 'H' means there were estimated data and missing maximums
  - An 'L' means there were estimated data and missing minimums
4. Daily values that are estimates are stored as out-of-range negative numbers; each one is scaled by a factor of 200. If the monthly flag shows there are estimates for a month, each daily value should be checked to see if it is less than -80; if so, it is an estimate and should be scaled up by 200.



IV.C.5.b SAS Data Set Name - POINTERS.

Number of Observations(Stations) - 249.

Number of Bytes/Observation - 18.

Total Length of File in Bytes - 4,482.

RECORD DESCRIPTION

<u>FIELD</u>	<u>VARIABLE NAME</u>	<u>TYPE</u>	<u>LENGTH</u>
Station identification	STATION	CHAR	6
First observation	FOBS_PTR	NUM	4
Last observation	LOBS_PTR	NUM	4

IV.C.6 NHIMS.PRECIP.DAILY

IV.C.6.a SAS Data Set Name - MAINDATA.

Number of Observations - 78,305.

Number of Bytes/Observation - 144.

Total Length of File in Bytes - 11,275,920.

RECORD DESCRIPTION

<u>FIELD</u>	<u>VARIABLE NAME</u>	<u>TYPE</u>	<u>LENGTH</u>
Station identification	STATION	CHAR	6
Calendar year	YEAR	NUM	3
Month	MONTH	NUM	2
Monthly summary of codes	MONTHSUM	NUM	2
Monthly total rainfall X 100	MONTOTAL	NUM	3
Remarks code for daily rainfall amount	P_CODE1-P_CODE31	CHAR	1 each
Daily rainfall amount X 100	PRECIP1-PRECIP31	NUM	3 each

NOTES

1. Missing numeric data are stored as a single period (.).
2. The units for daily rainfall amounts and the monthly total rainfall are inches.
3. The monthly total and the 31 daily rainfall amounts are adjusted in order to be stored as integers; to get the true data values, each number must be multiplied by 0.01.

4. The daily codes for rainfall amounts are blank if there are no special conditions. Otherwise, the possible values for the codes are:
  - A 'T' means the corresponding daily value was a trace value
  - An 'A' means the corresponding daily value was accumulated
  - An 'E' means the corresponding daily value was estimated
  - A 'B' means the corresponding daily value was both estimated and accumulated
  
5. The monthly summary variable summarizes the information in the daily code variables. The summary is stored as an integer number, which is used to flag special conditions during the month. The meanings of the flags are:
  - 8: Missing values during month
  - 4: Accumulated values during month
  - 2: Estimated values during month
  - 1: Trace amounts during month

If more than one condition is flagged, then value of MONTHSUM will be the sum of all the applicable conditions.

IV.C.6.b SAS Data Set Name - POINTERS.

Number of Observations(Stations) - 293.

Number of Bytes/Observation - 18.

Total Length of File in Bytes - 5,274.

RECORD DESCRIPTION

<u>FIELD</u>	<u>VARIABLE NAME</u>	<u>TYPE</u>	<u>LENGTH</u>
Station identification	STATION	CHAR	6
First observation	FOBS_PTR	NUM	4
Last observation	LOBS_PTR	NUM	4

IV.C.7 NHIMS.SNOWFALL

IV.C.7.a SAS Data Set Name - MAINDATA.

Number of Observations - 32,324.

Number of Bytes/Observation - 237.

Total Length of File in Bytes - 7,660,788.

RECORD DESCRIPTION

<u>FIELD</u>	<u>VARIABLE NAME</u>	<u>TYPE</u>	<u>LENGTH</u>
Station identification	STATION	CHAR	6
Calendar year	YEAR	NUM	3
Month	MONTH	NUM	2
Monthly summary of codes	MONTHSUM	NUM	2
Monthly total snowfall X 10	MONTHTOTAL	NUM	3
Remarks code for daily snowfall amount	S_CODE1-S_CODE31	CHAR	1 each
Remarks code for daily snow depth	D_CODE1-D_CODE31	CHAR	1 each
Daily snowfall amount X 10	SNOW1-SNOW31	NUM	3 each
Daily snow depth	DEPTH1-DEPTH31	NUM	2 each

NOTES

1. Missing numeric data are stored as a single period (.).
2. The units for snowfall amounts, snow depths, and the monthly total are inches.
3. The monthly total and the 31 daily snowfall amounts are adjusted in order to be stored as integers; to get the true data values, each number must be multiplied by 0.1.
4. The daily codes for snowfall amounts and snow depths are blank if there are no special conditions. Otherwise, the possible values for the codes are:
  - A 'T' means the corresponding daily value was a trace value
  - An 'A' means the corresponding daily value was accumulated
  - An 'E' means the corresponding daily value was estimated
  - A 'B' means the corresponding daily value was both estimated and accumulated

5. The monthly summary variable summarizes the information in the daily code variables. The summary is stored as an integer number, which is used to flag special conditions during the month. The meanings of the flags are:

- 128: Snowfall amounts missing during month
- 64: Snowfall amounts accumulated during month
- 32: Snowfall amounts estimated during month
- 16: Trace snowfall amounts during month
- 8: Snow depths missing during month
- 4: Snow depths accumulated during month
- 2: Snow depths estimated during month
- 1: Trace snow depths during month

If more than one condition is flagged, the value of MONTHSUM will be the sum of all the applicable conditions.

IV.C.7.b SAS Data Set Name - POINTERS.

Number of Observations(Stations) - 208.

Number of Bytes/Observation - 18.

Total Length of File in Bytes - 3,744.

RECORD DESCRIPTION

<u>FIELD</u>	<u>VARIABLE NAME</u>	<u>TYPE</u>	<u>LENGTH</u>
Station identification	STATION	CHAR	6
First observation	FOBS_PTR	NUM	4
Last observation	LOBS_PTR	NUM	4

IV.C.8 NHIMS.MONTHLY

IV.C.8.a SAS Data Set Name - MAINDATA.

Number of Observations - 95,821.

Number of Bytes/Observation - 82.

Total Length of File in Bytes - 1,149,852.

RECORD DESCRIPTION

<u>FIELD</u>	<u>VARIABLE NAME</u>	<u>TYPE</u>	<u>LENGTH</u>
Station number from NCDC	STATION	NUM	4
Calendar year	YEAR	NUM	3
Month	MONTH	NUM	2
Mean maximum monthly temperature	MAXTEMP	NUM	4
Mean minimum monthly temperature	MINTEMP	NUM	4
Monthly mean temperature	TMEAN	NUM	4
Departure from normal temperature	TDEPART	NUM	4
Heating degree days	DEGDAYS	NUM	3
Highest maximum temperature during month	HIGHEST	NUM	2
Day of occurrence of highest temperature	HIGHDAY	NUM	2
Lowest minimum temperature during month	LOWEST	NUM	2
Day of occurrence of lowest temperature	LOWDAY	NUM	2
Total precipitation during month	PRECIP	NUM	4
Precipitation departure from normal	PDEPART	NUM	4
Maximum daily precipitation during month	PPTMAX	NUM	4
Date of maximum daily precipitation	PMAXDAY	NUM	2
Total monthly snowfall	SNOFALL	NUM	4
Maximum daily snowfall for month	MAXSNO	NUM	4
Day of occurrence of maximum snowfall	MAXSDAY	NUM	2
Number of days with snowfall greater than 1 inch	SNODAYS	NUM	2
Total wind run for month	WIND	NUM	3
Total monthly evaporation	EVAP	NUM	4
Cooling degree days	COOLDAYS	NUM	3
Flag showing if maximum temperature occurred on more than one day	PLUS1	CHAR	1
Flag showing if minimum temperature occurred on more than one day	PLUS2	CHAR	1
Flag showing if maximum precipitation occurred on more than one day	PLUS3	CHAR	1
Flag showing if maximum snowfall occurred on more than one day	PLUS4	CHAR	1

<u>FIELD</u>	<u>VARIABLE NAME</u>	<u>TYPE</u>	<u>LENGTH</u>
Flag showing if total monthly precipitation is trace or missing	PLUS5	CHAR	1
Flag showing if total monthly snowfall is trace or missing	PLUS6	CHAR	1

NOTES

1. The PLUS flags will have a '+' character value if the condition is true, and will be blank otherwise.
2. The units are inches for snowfall and precipitation, degrees Fahrenheit for temperature, and miles for wind movement.

IV.C.8.b SAS Set Name - POINTERS.

Number of Observations(Stations) - 258.

Number of Bytes/Observation - 29.

Total Length of File in Bytes - 7,482.

RECORD DESCRIPTION

<u>FIELD</u>	<u>VARIABLE NAME</u>	<u>TYPE</u>	<u>LENGTH</u>
Station identification	STATION	NUM	4
First observation	FOBS_PTR	NUM	4
Last observation	LOBS_PTR	NUM	4

IV.C.9 NHIMS.SNOW.COURSE

IV.C.9.a SAS Data Set Name - SNOWWY.

Number of Observations - 2,354.

Number of Bytes/Observation - 79.

Total Length of File in Bytes - 185,966.

\*\*\* NOTE \*\*\*

The following record description shows the snow course file as it actually exists. The snow course information was stored in a SAS data set before the NHIMS project began; in order to make its structure more consistent with the other files, changes will be made before the full release of NHIMS.

RECORD DESCRIPTION

<u>FIELD</u>	<u>VARIABLE NAME</u>	<u>TYPE</u>	<u>LENGTH</u>
FIPS code for state	STATE	NUM	2
Code showing when monthly measurement occurred	CARDNO	NUM	2
Monthly snow depths	DEPTH1-DEPTH6	NUM	2 each
Month of measurement	MONTH1-MONTH6	NUM	2 each
Day of measurement	DAY1-DAY6	NUM	2 each
Water Year	WAT_YEAR	NUM	4
Snow water equivalent	WATER1-WATER6	NUM	4 each
Station code from SCS	STATION	CHAR	5
Station type	TYPE	CHAR	2

NOTES

1. This file structure is subject to revision.
2. The units for snow depth and snow water equivalent are inches.
3. The possible values for the CARDNO variable are:
  - 1: first-of-month measurement
  - 2: mid-month measurement
  - 3-6: special measurements
4. The station type variable (TYPE) has the following possible values:
  - A: aerial station
  - R: radio isotope gage station
  - S: snotel site
  - E: estimated values
  - M: soil moisture station

IV.C.9.b SAS Data Set Name - SNOWID.

Number of Observations - 8,698.

Number of Bytes/Observation - 79.

Total Length of File in Bytes - 687,142.

RECORD DESCRIPTION

<u>FIELD</u>	<u>VARIABLE NAME</u>	<u>TYPE</u>	<u>LENGTH</u>
FIPS code for state	STATE	NUM	2
Code showing when monthly measurement occurred	CARDNO	NUM	2
Monthly snow depths	DEPTH1-DEPTH6	NUM	2 each
Month of measurement	MONTH1-MONTH6	NUM	2 each
Day of measurement	DAY1-DAY6	NUM	2 each
Water Year	WAT_YEAR	NUM	4
Snow water equivalent	WATER1-WATER6	NUM	4 each
Station code from SCS	STATION	CHAR	5
Station type	TYPE	CHAR	2

NOTES

1. This file structure is subject to revision.
2. The units for snow depth and snow water equivalent are inches.
3. The possible values for the CARDNO variable are:
  - 1: first-of-month measurement
  - 2: mid-month measurement
  - 3-6: special measurements
4. The station type variable (TYPE) has the following possible values:
  - A: aerial station
  - R: radio isotope gage station
  - S: snotel site
  - E: estimated values
  - M: soil moisture station

IV.C.9.c SAS Data Set Name - PTR\_SWY0.

Number of Observations(Stations) - 55.

Number of Bytes/Observations - 23.

Total Length of File in Bytes - 1,265.



RECORD DESCRIPTION

<u>FIELD</u>	<u>VARIABLE NAME</u>	<u>TYPE</u>	<u>LENGTH</u>
SCS station code	STATION	CHAR	5
First observation	FOBS_PTR	NUM	4
Last observation	LOBS_PTR	NUM	4

IV.C.9.d SAS Data Set Name - SNOWNAME.

Number of Observations - 365.

Number of Bytes/Observation - 112.

Total Length of File in Bytes - 40,880.

\*\*\* NOTE \*\*\*

This file is the index for the snow course main data files.

RECORD DESCRIPTION

<u>FIELD</u>	<u>VARIABLE NAME</u>	<u>TYPE</u>	<u>LENGTH</u>
FIPS code for state	STATE	NUM	2
Unknown value from SCS	WHAT	NUM	2
Section number	SECTION	NUM	2
Elevation	ELEV	NUM	3
Latitude	LAT	NUM	3
Longitude	LONG	NUM	3
SCS station code	STATION	CHAR	5
Station type	TYPE	CHAR	2
Township	TOWNSHIP	CHAR	3
Range	RANGE	CHAR	3
Station name	NAME	CHAR	23
Location narrative	LOCATION	CHAR	57

## V. DATA RETRIEVAL FOR THE SAS<sup>®</sup> PROGRAMMER

### A. Sample Data Retrieval Programs

An example of a simple data retrieval program for a NHIMS data set is given in Figure 3. The file being accessed is the evaporation file, and data are being requested for one station, whose number is given in the DATA step named REQUEST. Note the use of the evaporation pointer file, which is first searched to find the range of observation numbers for each station requested; the first and last observation numbers are then used as the range of values for the POINT option on the SET statement to directly access the data. Also, the SAS macro facility is used to create a routine called ECONSTERT, which adjusts the daily evaporation numbers from integer form to their correct values. The format of the output is taken care of by PROC PRINT.

Figure 4 contains another example program for retrieving evaporation data from NHIMS, but with a slightly different strategy. Suppose the programmer wants information on station 104598, and from a listing of the contents of the pointer file knows that this station ranges from observations 820 through 859, inclusive. By supplying those observation numbers explicitly in the DO statement of the DATA step SUBSET, the program can directly access those observations without having to look them up in the pointer file. Thus, the information is retrieved more efficiently and less expensively. Note, however, the importance of the STOP statement in the program in Figure 4. Because the SET statement is located within the DO loop, termination of the DATA step will not occur when the data set runs out of observations; therefore, the STOP is necessary to avoid getting into an infinite loop.

```

//JEVAP   JOB   (XXXXXX,XXX-XX-XXXX),XXXXXX,CLASS=A
/*       DEST=TS
//READ   EXEC   SAS,OPTIONS='MACRO'
//EDATA  DD   DISP=(OLD,SHR),DSN=NHIMS.EVAP
//SYSIN  DD   *

%MACRO ECONST; /* MACRO TO SCALE EVAPORATION VALUES */
  FORMAT EVAP1-EVAP31 5.2;
  %DO I = 1 %TO 31;
    EVAP&I = EVAP&I * ADJUST;
  %END;
%MEND ECONST;

DATA REQUEST; /* THIS IS THE USER INPUT REQUEST */
  INPUT STATION $;
  CARDS;
    104598
;

PROC SORT DATA = REQUEST;
  BY STATION;

DATA PTRS; /* MATCH STATION REQUESTS WITH POINTERS */

  MERGE EDATA.POINTERS REQUEST(IN=COMMON);
  BY STATION;
  IF FOBS_PTR EQ . THEN
    DO;
      PUT ' STATION NUMBER ' STATION @;
      PUT ' IS AN ILLEGAL REQUEST.';
    END;

  /* KEEP ONLY STATIONS AND POINTERS REQUESTED */
  IF COMMON AND FOBS_PTR NE .;

DATA SUBSET; /* FIND THE REQUESTED DATA */
  SET PTRS;
  RETAIN ADJUST 0.01;
  FORMAT EVAP_TOT 6.2;
  DO I = FOBS_PTR TO LOBS_PTR;
    SET EDATA.MAINDATA POINT = I;
    %ECONST
    EVAP_TOT = EVAP_TOT * ADJUST;
    OUTPUT;
  END;
  DROP FOBS_PTR LOBS_PTR ADJUST;

PROC PRINT   DATA=SUBSET;

```

Figure 3. SAS® Program for Data Retrieval by Station Number.

```

//JEVAP   JOB (XXXXX,XXX-XX-XXXX),XXXXXX,CLASS=A
/*       DEST=TS
//READ   EXEC SAS,OPTIONS='MACRO'
//EDATA  DD DISP=(OLD,SHR),DSN=NHIMS.EVAP
//SYSIN  DD *
          OPTIONS NOCENTER;

          %MACRO ECONVERT;
            FORMAT EVAP1-EVAP31 5.2;
            %DO I = 1 %TO 31;
              EVAP&I = EVAP&I * ADJUST;
            %END;
          %MEND ECONVERT;

DATA SUBSET; /* FIND THE REQUESTED DATA */
  RETAIN ADJUST 0.01;
  FORMAT EVAP_TOT 6.2;
  DO I = 820 TO 859; /* GET DATA FROM THIS */
                    /* OBSERVATION RANGE */
    SET EDATA.MAINDATA POINT = I;
    %ECONVERT
    EVAP_TOT = EVAP_TOT * ADJUST;
    /* SUBSETTING IF STATEMENTS CAN BE PLACED HERE */
    OUTPUT;
  END;
  DROP ADJUST;
  STOP; /* BE SURE TO INCLUDE THIS STOP STATEMENT */

PROC PRINT DATA=SUBSET;
/*

```

Figure 4. SAS® Program for Data Retrieval with Pointers.

Because the programs in Figures 3 and 4 are used to retrieve the same data values, a comparison was made in the run times of the two retrieval methods. Even though only a few observations, 40, were retrieved, the method presented in Figure 4 required a full second less CPU time.

### B. SAS<sup>®</sup> Programs to Generate Reports

An example program for accessing the NHIMS temperature data set is shown in Figure 5; this example illustrates the use of PROC PRINT and its various options to generate a listing of the requested data. Also featured in Figure 5 is the use of the subsetting IF statement to retrieve only the observations for 1982 for the two requested stations. The output from the program is shown in Figures 6 and 7. Although this is a simple example, PROC PRINT offers many other options for reformatting the data listing. (For more information see the SAS User's Guide: Basics.) The advantage of using PROC PRINT is that the formatting is taken care of internally by the SAS system.

Figure 8 contains an example of accessing NHIMS streamflow data, and using PUT statements to generate an output listing that is very similar to the current HISARS output. The JCL statements have been omitted. The results of the program are shown in Figure 9.

### C. SAS<sup>®</sup> Program to Generate Statistics

Figure 10 shows a program that retrieves precipitation data from NHIMS and performs statistical computations with PROC MEANS. The calculations are performed using the monthly totals. The JCL statements are omitted. The results of the analyses are shown in Figures 11 and 12.

```

//JTEMP   JOB (XXXXXX,XXX-XX-XXXX),XXXXXX,CLASS=A
/*       DEST=TS
//OUT     EXEC SAS
//TDATA   DD DISP=(OLD,SHR),DSN=NHIMS.AIRTEMP
//SYSIN   DD *
          OPTIONS NOCENTER;

          DATA REQUEST; /* THIS IS THE USER INPUT REQUEST */
            INPUT STATION $;
            CARDS;
              106152
              100010
            ;

          PROC SORT DATA = REQUEST;
            BY STATION;

          DATA PTRS; /* MATCH STATION REQUESTS WITH POINTERS */
            MERGE TDATA.POINTERS REQUEST(IN=COMMON);
            BY STATION;
            IF FOBS_PTR EQ . THEN
              DO;
                PUT 'STATION NUMBER ' STATION @;
                PUT ' IS AN ILLEGAL REQUEST.';
              END;
            /* KEEP ONLY STATIONS AND POINTERS REQUESTED */
            IF COMMON AND FOBS_PTR NE . ;

          DATA SUBSET; /* FIND THE REQUESTED DATA */
            SET PTRS;
            DO I = FOBS_PTR TO LOBS_PTR;
              SET TDATA.MAINDATA POINT = I;
              DROP FOBS_PTR LOBS_PTR ;
              IF YEAR = 1982 THEN OUTPUT;
            END;

          PROC PRINT DATA = SUBSET;
            TITLE 1982 AIR TEMPERATURES;
            BY STATION;
            ID YEAR MONTH;
/*

```

Figure 5. SAS® Program to Generate Output with PROC PRINT.



1982 AIR TEMPERATURES

STATION CODE NUMBER=106152

Y E A R	M					M				M												M										
	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	
1982	1	28	23	28	28	18	7	20	29	31	28	28	30	35	37	35	35	35	34	33	30	30	25	43	38	35	42	38	34	36	36	36
1982	2	35	36	23	15	15	20	23	21	22	21	31	33	37	46	49	49	45	40	53	52	49	32	31	35	38	40	42	48	.	.	.
1982	3	49	44	42	42	40	45	47	48	51	52	36	38	43	44	38	43	45	39	33	37	44	49	53	51	59	53	45	49	39	38	41
1982	4	40	40	42	39	44	34	43	44	50	57	56	49	52	47	41	48	48	40	44	50	62	71	67	52	57	59	64	51	52	67	.
1982	5	66	62	46	49	55	61	56	54	56	58	64	62	69	69	60	70	59	55	62	69	75	64	63	73	76	64	53	54	59	63	72
1982	6	65	63	60	57	51	55	60	68	74	78	82	68	60	73	81	82	84	82	88	87	78	79	82	84	75	85	69	60	69	74	.
1982	7	61	72	63	64	66	71	70	72	76	81	85	81	85	64	55	64	74	81	82	85	79	73	79	83	89	91	94	92	94	98	85
1982	8	72	72	69	76	82	89	96	74	90	70	69	74	78	67	75	83	83	88	95	93	90	86	83	87	90	86	83	78	68	68	73
1982	9	82	86	93	72	77	81	81	82	79	58	48	58	58	58	64	73	78	82	78	69	66	74	79	62	76	59	53	53	57	60	.
1982	10	61	58	53	49	53	52	49	49	54	61	63	66	67	69	69	69	43	43	48	52	56	62	53	61	56	49	47	47	43	41	43
1982	11	43	47	47	48	42	42	41	41	33	37	36	35	32	33	33	39	49	38	38	30	29	27	23	27	28	33	36	40	40	42	.
1982	12	33	36	48	48	38	34	37	24	35	28	30	32	32	33	38	47	42	37	38	38	42	39	34	32	30	31	26	24	22	20	22

Y E A R	M					M				M												M										
	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	
1982	1	18	12	17	20	3	-23	3	20	21	19	22	23	26	26	31	29	27	23	21	16	16	19	24	33	27	32	31	27	30	30	32
1982	2	28	31	19	2	2	8	15	15	8	5	16	22	30	35	37	41	37	32	38	41	30	12	25	22	28	29	34	33	.	.	.
1982	3	39	33	32	32	29	30	30	33	41	39	31	28	33	37	29	28	32	32	29	28	29	30	31	30	33	37	35	35	29	30	33
1982	4	29	29	31	29	30	31	26	27	31	34	43	41	42	32	28	29	37	28	26	28	33	39	43	35	27	31	33	43	31	37	.
1982	5	38	36	34	31	27	33	47	33	38	39	28	37	35	47	49	43	48	41	34	38	41	46	37	39	50	48	31	41	37	36	42
1982	6	39	43	39	38	44	43	37	35	44	46	45	49	51	50	51	52	57	49	50	53	54	51	54	53	64	55	52	53	53	.	
1982	7	55	45	49	39	47	43	54	49	52	50	52	51	49	54	37	45	38	52	50	51	48	36	37	42	51	55	51	56	56	55	54
1982	8	44	42	51	53	45	45	58	58	54	55	52	42	50	54	48	48	49	46	59	65	60	52	44	46	51	47	44	45	43	50	40
1982	9	43	52	60	52	44	45	41	45	49	44	34	43	39	41	34	34	45	48	54	50	46	39	44	54	50	41	42	44	36	32	.
1982	10	29	40	41	38	31	37	37	38	29	34	38	40	38	41	47	48	35	30	29	30	34	47	45	43	48	41	36	31	32	29	29
1982	11	38	27	34	40	35	35	28	28	29	31	30	25	21	29	22	31	33	33	31	27	7	9	15	15	20	24	30	33	33	33	.
1982	12	30	32	33	35	29	27	14	8	18	10	8	16	21	25	32	35	33	27	34	31	34	30	29	25	21	23	19	17	20	6	9

Figure 7. SAS® Output with PROC PRINT.



```

OPTIONS NOCENTER ;

%MACRO SCONVERT;
  FORMAT STREAM1-STREAM31 9.2;
  %DO I = 1 %TO 31;
    STREAM&I = STREAM&I * ADJUST;
  %END;
%MEND SCONVERT;

%MACRO MONLIST;
  %DO I = 1 %TO 31;
    CURLINE = &I + 6;
    PUT #CURLINE @COL STREAM&I 10.2;
    %IF &I = 31 %THEN
      %DO;
        PUT #38;
        PUT #39 @COL MONTOTAL 10.2;
      %END;
    %END;
%MEND MONLIST;

%MACRO DAYSPEC;
  %DO D = 1 %TO 31;
    CURLINE = &D + 6;
    WHATDAY = &D;
    PUT #CURLINE @3 WHATDAY 2.;
    %IF &D = 31 %THEN
      %DO;
        PUT #38;
        PUT #39 'TOTAL';
      %END;
    %END;
%MEND DAYSPEC;

%MACRO HEAD;
  PUT #1 @100 STATION;
  PUT #2;
  WATYR = YEAR + 1;
  PUT #4 YEAR 1-4 @5 '-' WATYR 6-9;
  PUT #5 @13 'OCTOBER' @23 'NOVEMBER'
    @33 'DECEMBER' @43 'JANUARY'
    @52 'FEBRUARY' @64 'MARCH'
    @74 'APRIL' @84 'MAY'
    @95 'JUNE' @105 'JULY'
    @114 'AUGUST' @121 'SEPTEMBER';
  PUT #6;
%MEND HEAD;

```

Figure 8(a). SAS® Program to Generate Output with PUT Statements.

```

DATA REQUEST; /* THIS IS THE USER INPUT REQUEST */
  INPUT STATION $;
  CARDS;
    10038000
;
DATA PTRS; /* MATCH STATION REQUESTS WITH POINTERS */
  MERGE SDATA.POINTERS REQUEST(IN=COMMON);
  BY STATION;
  IF FOBS_PTR EQ . THEN
    DO;
      PUT 'STATION NUMBER ' STATION @;
      PUT ' IS AN ILLEGAL REQUEST.';
    END;
  /* KEEP ONLY STATIONS AND POINTERS REQUESTED */
  IF COMMON AND FOBS_PTR NE . ;

DATA SUBSET; /* FIND THE REQUESTED DATA */
  FORMAT MONTOTAL 10.2;
  SET PTRS;
  RETAIN ADJUST 0.01;
  DO I = FOBS_PTR TO LOBS_PTR;
    SET SDATA.MAINDATA POINT = I;
    %SCONVERT
    MONTOTAL = MONTOTAL * ADJUST;
    DROP FOBS_PTR LOBS_PTR ADJUST;
    OUTPUT;
  END;

DATA _NULL_ ;
  FILE PRINT N=PS;
  TITLE3 MEAN DAILY STREAMFLOW IN CFS;
  %DAYSPEC
  COL=0;
  DO M = 1 TO 12;
    SET SUBSET END=EOF;
    BY STATION;
    IF M = 1 THEN
      DO;
        %HEAD
      END;
    COL = COL+10;
    %MONLIST
    IF EOF THEN STOP;
    IF LAST.STATION THEN
      DO;
        PUT _PAGE_;
        RETURN;
      END;
  END;
  PUT _PAGE_;

```

Figure 8(b). SAS® Program to Generate Output with PUT Statements.

SAS

MEAN DAILY STREAMFLOW IN CFS

10038000

1979-1980

53

	OCTOBER	NOVEMBER	DECEMBER	JANUARY	FEBRUARY	MARCH	APRIL	MAY	JUNE	JULY	AUGUST	SEPTEMBER
1	99.00	145.00	90.00	114.00	85.00	394.00	271.00	2200.00	1390.00	793.00	267.00	228.00
2	100.00	140.00	90.00	126.00	100.00	388.00	258.00	2290.00	1420.00	849.00	279.00	227.00
3	101.00	148.00	100.00	120.00	130.00	372.00	253.00	2260.00	1590.00	880.00	286.00	230.00
4	100.00	150.00	110.00	118.00	150.00	361.00	256.00	2250.00	1640.00	889.00	274.00	218.00
5	101.00	155.00	120.00	124.00	150.00	350.00	271.00	2270.00	1660.00	919.00	285.00	209.00
6	100.00	145.00	130.00	120.00	140.00	351.00	329.00	2290.00	1690.00	1130.00	279.00	195.00
7	100.00	141.00	148.00	110.00	166.00	387.00	360.00	2320.00	1690.00	968.00	305.00	199.00
8	99.00	141.00	154.00	109.00	130.00	385.00	315.00	2360.00	1650.00	927.00	274.00	196.00
9	95.00	141.00	154.00	121.00	110.00	375.00	318.00	2370.00	1640.00	911.00	266.00	200.00
10	84.00	130.00	130.00	130.00	95.00	358.00	306.00	2380.00	1660.00	891.00	267.00	201.00
11	85.00	150.00	110.00	110.00	95.00	346.00	355.00	2350.00	1700.00	876.00	258.00	216.00
12	86.00	125.00	100.00	100.00	95.00	333.00	385.00	2320.00	1710.00	831.00	259.00	232.00
13	88.00	135.00	95.00	109.00	110.00	354.00	416.00	2270.00	1710.00	761.00	255.00	200.00
14	87.00	142.00	100.00	154.00	120.00	336.00	537.00	2190.00	1690.00	732.00	245.00	184.00
15	87.00	149.00	110.00	186.00	140.00	354.00	826.00	2160.00	1650.00	804.00	244.00	178.00
16	93.00	153.00	110.00	498.00	148.00	471.00	954.00	2170.00	1600.00	730.00	266.00	172.00
17	93.00	149.00	120.00	824.00	148.00	480.00	1080.00	2230.00	1550.00	638.00	280.00	177.00
18	89.00	177.00	120.00	800.00	154.00	422.00	1190.00	2200.00	1490.00	613.00	264.00	174.00
19	121.00	159.00	120.00	752.00	166.00	384.00	1320.00	2180.00	1460.00	562.00	275.00	166.00
20	177.00	141.00	122.00	400.00	202.00	362.00	1400.00	2180.00	1460.00	601.00	289.00	161.00
21	156.00	145.00	120.00	250.00	293.00	344.00	1510.00	2160.00	1440.00	557.00	270.00	160.00
22	201.00	150.00	131.00	230.00	439.00	315.00	1630.00	2120.00	1390.00	532.00	256.00	160.00
23	189.00	140.00	120.00	230.00	524.00	316.00	1840.00	2000.00	1360.00	534.00	247.00	158.00
24	161.00	140.00	110.00	240.00	600.00	303.00	2070.00	1940.00	1270.00	492.00	229.00	157.00
25	157.00	150.00	110.00	265.00	678.00	366.00	2130.00	1890.00	1150.00	452.00	222.00	152.00
26	155.00	162.00	120.00	281.00	784.00	334.00	2250.00	1800.00	974.00	421.00	235.00	145.00
27	153.00	120.00	110.00	190.00	664.00	298.00	2210.00	1730.00	916.00	393.00	228.00	144.00
28	149.00	100.00	100.00	140.00	479.00	281.00	2130.00	1650.00	838.00	342.00	226.00	141.00
29	147.00	90.00	95.00	120.00	401.00	274.00	2080.00	1500.00	812.00	339.00	221.00	140.00
30	148.00	90.00	95.00	100.00	-	279.00	2060.00	1480.00	767.00	329.00	222.00	137.00
31	147.00	.	95.00	85.00	-	286.00	-	1430.00	-	308.00	226.00	.
TOTAL	3748.00	4203.00	3539.00	7256.00	7496.00	10959.00	31310.00	64940.00	42967.00	21004.00	7999.00	5457.00

Figure 9. SAS® Output with PUT Statements.

```

OPTIONS NOCENTER;

%MACRO PCONVERT;
%*
%* MACRO TO CONVERT THE DAILY RAINFALL AMOUNTS ;
%* AND MONTH TOTAL INTO THE TRUE NUMERIC VALUE ;
%*
FORMAT PRECIP1-PRECIP31 9.2;
%DO I = 1 %TO 31;
PRECIP&I = PRECIP&I * ADJUST;
%END;
%MEND PCONVERT;

DATA REQUEST; /* THIS IS THE USER INPUT REQUEST */
INPUT STATION $;
CARDS;
106152
;

DATA PTRS; /* MATCH STATION REQUESTS WITH POINTERS */
MERGE RDATA.POINTERS REQUEST(IN=COMMON);
BY STATION;
IF FOBS_PTR EQ . THEN
DO;
PUT ' STATION NUMBER ' STATION @;
PUT ' IS AN ILLEGAL REQUEST.';
END;
/* KEEP ONLY STATIONS AND POINTERS REQUESTED */
IF COMMON AND FOBS_PTR NE .;

DATA SUBSET (KEEP = STATION YEAR MONTH MONTOTAL);

/* FIND THE REQUESTED DATA */
FORMAT MONTOTAL 10.2;
SET PTRS;
RETAIN ADJUST 0.01;

/* READ DATA FOR ONE STATION */
DO I = FOBS_PTR TO LOBS_PTR;
SET RDATA.MAINDATA POINT = I;
%PCONVERT
MONTOTAL = MONTOTAL * ADJUST;
IF YEAR > 1959 THEN OUTPUT;
END;

PROC MEANS DATA=SUBSET;
TITLE MONTHLY PRECIPITATION STATISTICS;
VAR MONTOTAL;
BY YEAR;

```

Figure 10. SAS® Program to Calculate Statistics.

MONTHLY PRECIPITATION STATISTICS

VARIABLE	LABEL	N	MEAN	STANDARD DEVIATION	MINIMUM VALUE	MAXIMUM VALUE	STD ERROP OF MEAN	SUM
YEAR THAT DATA WAS RECORDED	(NUMERIC=1960 -----)							
MONTH TOTAL	TOTAL RAINFALL AMOUNT FOR THE MONTH	12	1.78666667	1.00011211	0	3.91000000	0.28870750	21.44000000
YEAR THAT DATA WAS RECORDED	(NUMERIC=1961 -----)							
MONTH TOTAL	TOTAL RAINFALL AMOUNT FOR THE MONTH	12	1.99333333	1.45467480	0.22000000	5.38000000	0.41992845	23.92000000
YEAR THAT DATA WAS RECORDED	(NUMERIC=1962 -----)							
MONTH TOTAL	TOTAL RAINFALL AMOUNT FOR THE MONTH	12	1.78666667	1.06014865	0.19000000	3.35000000	0.30603855	21.44000000
YEAR THAT DATA WAS RECORDED	(NUMERIC=1963 -----)							
MONTH TOTAL	TOTAL RAINFALL AMOUNT FOR THE MONTH	12	1.69166667	0.96991877	0.38000000	3.66000000	0.27999143	20.30000000
YEAR THAT DATA WAS RECORDED	(NUMERIC=1964 -----)							
MONTH TOTAL	TOTAL RAINFALL AMOUNT FOR THE MONTH	12	2.63000000	2.36086271	0.89000000	7.25000000	0.68152236	31.56000000
YEAR THAT DATA WAS RECORDED	(NUMERIC=1965 -----)							
MONTH TOTAL	TOTAL RAINFALL AMOUNT FOR THE MONTH	12	1.38666667	0.99589460	0.35000000	3.48000000	0.28749001	16.64000000
YEAR THAT DATA WAS RECORDED	(NUMERIC=1966 -----)							
MONTH TOTAL	TOTAL RAINFALL AMOUNT FOR THE MONTH	12	1.57833333	1.32290200	0.06000000	3.74000000	0.38188891	18.94000000
YEAR THAT DATA WAS RECORDED	(NUMERIC=1967 -----)							
MONTH TOTAL	TOTAL RAINFALL AMOUNT FOR THE MONTH	12	1.68500000	1.34119689	0	3.48000000	0.38717019	20.22000000
YEAR THAT DATA WAS RECORDED	(NUMERIC=1968 -----)							
MONTH TOTAL	TOTAL RAINFALL AMOUNT FOR THE MONTH	12	2.26916667	1.29842883	0.65000000	4.34000000	0.37482412	27.23000000
YEAR THAT DATA WAS RECORDED	(NUMERIC=1969 -----)							
MONTH TOTAL	TOTAL RAINFALL AMOUNT FOR THE MONTH	12	1.66750000	1.58494121	0	4.83000000	0.45753312	20.01000000
YEAR THAT DATA WAS RECORDED	(NUMERIC=1970 -----)							
MONTH TOTAL	TOTAL RAINFALL AMOUNT FOR THE MONTH	12	2.56333333	1.81014481	0.16000000	7.67000000	0.52254380	30.76000000
YEAR THAT DATA WAS RECORDED	(NUMERIC=1971 -----)							
MONTH TOTAL	TOTAL RAINFALL AMOUNT FOR THE MONTH	12	2.52500000	0.99752876	0.96000000	4.81000000	0.28796175	30.30000000
YEAR THAT DATA WAS RECORDED	(NUMERIC=1972 -----)							
MONTH TOTAL	TOTAL RAINFALL AMOUNT FOR THE MONTH	12	2.56833333	1.70268683	0.59000000	5.86000000	0.49152335	30.82000000

55

Figure 11. SAS® Output from PROC MEANS.

MONTHLY PRECIPITATION STATISTICS

VARIABLE	LABEL	N	MEAN	STANDARD DEVIATION	MINIMUM VALUE	MAXIMUM VALUE	STD ERROR OF MEAN	SUM
YEAR THAT DATA WAS RECORDED (NUMERIC=1973 -----)								
MONTH TOTAL	TOTAL RAINFALL AMOUNT FOR THE MONTH	12	2.29500000	2.42016716	0.02000000	7.32000000	0.69864208	27.54000000
YEAR THAT DATA WAS RECORDED (NUMERIC=1974 -----)								
MONTH TOTAL	TOTAL RAINFALL AMOUNT FOR THE MONTH	12	2.16000000	1.82620321	0.05000000	6.70000000	0.52717946	25.92000000
YEAR THAT DATA WAS RECORDED (NUMERIC=1975 -----)								
MONTH TOTAL	TOTAL RAINFALL AMOUNT FOR THE MONTH	12	2.75583333	1.23732452	0	4.96000000	0.35718482	33.07000000
YEAR THAT DATA WAS RECORDED (NUMERIC=1976 -----)								
MONTH TOTAL	TOTAL RAINFALL AMOUNT FOR THE MONTH	12	1.86250000	0.93886900	0.06000000	2.90000000	0.27102814	22.35000000
YEAR THAT DATA WAS RECORDED (NUMERIC=1977 -----)								
MONTH TOTAL	TOTAL RAINFALL AMOUNT FOR THE MONTH	12	1.90500000	1.43560250	0.47000000	4.71000000	0.41442275	22.86000000
YEAR THAT DATA WAS RECORDED (NUMERIC=1978 -----)								
MONTH TOTAL	TOTAL RAINFALL AMOUNT FOR THE MONTH	12	2.02500000	1.17260782	0.09000000	4.74000000	0.33850272	24.30000000
YEAR THAT DATA WAS RECORDED (NUMERIC=1979 -----)								
MONTH TOTAL	TOTAL RAINFALL AMOUNT FOR THE MONTH	12	2.11166667	1.26432039	0.41000000	4.12000000	0.36497786	25.34000000
YEAR THAT DATA WAS RECORDED (NUMERIC=1980 -----)								
MONTH TOTAL	TOTAL RAINFALL AMOUNT FOR THE MONTH	12	2.33833333	1.39119852	0.75000000	4.80000000	0.40160442	28.06000000
YEAR THAT DATA WAS RECORDED (NUMERIC=1981 -----)								
MONTH TOTAL	TOTAL RAINFALL AMOUNT FOR THE MONTH	12	2.40083333	1.27076540	0.01000000	4.63000000	0.36683837	28.81000000
YEAR THAT DATA WAS RECORDED (NUMERIC=1982 -----)								
MONTH TOTAL	TOTAL RAINFALL AMOUNT FOR THE MONTH	12	2.30583333	0.86177890	0.67000000	3.75000000	0.24877414	27.67000000
YEAR THAT DATA WAS RECORDED (NUMERIC=1983 -----)								
MONTH TOTAL	TOTAL RAINFALL AMOUNT FOR THE MONTH	12	2.65833333	1.41024713	0.77000000	5.95000000	0.40710328	31.90000000
YEAR THAT DATA WAS RECORDED (NUMERIC=1984 -----)								
MONTH TOTAL	TOTAL RAINFALL AMOUNT FOR THE MONTH	12	2.19666667	1.20505249	0.66000000	5.02000000	0.34786869	26.36000000
YEAR THAT DATA WAS RECORDED (NUMERIC=1985 -----)								
MONTH TOTAL	TOTAL RAINFALL AMOUNT FOR THE MONTH	1	0.34000000	.	0.34000000	0.34000000	.	0.34000000

56

Figure 12. SAS® Output from PROC MEANS.

## VI. IMPLEMENTATION

### A. User Point of View

The main purpose of creating a data management system such as NHIMS is to provide users with a tool with which they can easily and quickly retrieve hydrologic information. Current users of HISARS, who are already accustomed to the access command language, will be able to use the same commands to retrieve data with the NHIMS system; only the name of the EXEC called in the JCL statements will have to be changed. However, more options will be available to users with the new system, allowing more versatility in data set retrieval. For instance, one option will be to allow users to receive the output in metric units. In addition, the current processing functions will be expanded so that most of them will be able to analyze data with or using more than one type of element, unlike the single-element processing which now exists in HISARS. Also, the ability to readily add new processing options in the future will allow the system to expand to new application needs.

Other options will exist to allow the user to more efficiently utilize the NHIMS programs, minimizing processing time and thus reducing costs. For example, if a user has a listing of the pointer file, and from that obtains the observation numbers for a particular station, those numbers can be specified in a POINTER command, allowing the system to directly access the requested data without having to search the pointer file for those observation numbers. Also, a user who regularly wants data from one station can make up a personalized pointer file in order to obtain a map of the observation numbers by year for that station or create a specialized data file. Figure 13

```

OPTIONS NOCENTER;

DATA TEMP.POINTERS (KEEP =STATION FIRST_YR LAST_YR);
  LABEL  STATION = 'STATION CODE NUMBER'
        FIRST_YR = 'FIRST OBS. OF STATION IN DATA FILE'
        LAST_YR  = 'LAST OBS. OF STATION IN DATA FILE';
DO I = 38361 TO 39380;
  SET TEMP.MAINDATA;
  BY STATION YEAR;
  LENGTH FIRST_YR LAST_YR 4;
  RETAIN FIRST_YR;
  IF FIRST.YEAR THEN FIRST_YR = _N_;
  IF LAST.YEAR THEN
    DO;
      LAST_YR = _N_;
      OUTPUT TEMP.POINTERS;
    END;
END;
STOP;

PROC PRINT DATA=TEMP.POINTERS;

```

Figure 13. SAS® Program to Create a Personalized Pointer File.



illustrates a sample program for such a personalized pointer file. In this example, the user knows that the information for the Moscow weather station is located from observations 38,361 to 39,380, inclusive, in the temperature data set; the program will create a map of those observations by year, so that the user can choose the observation numbers for the exact years for which data are required.

Because the NHIMS files are stored as permanent SAS data sets, SAS programmers will have the ability to access the hydrologic information directly with their own programs, or with the help of the NHIMS macro library. This feature will make the data available to a large number of users who wish to perform different data retrievals and analyses than those offered by the NHIMS system. SAS programmers can then make use of the wide variety of SAS procedures for doing statistical analysis, and can also design their own personalized output formats.

#### B. Programmer Point of View

SAS programmers will be able to perform routine maintenance tasks, especially data entry, quite easily with the NHIMS system. When input formats change, the maintenance programmer will simply have to write a small program to read the data into a temporary or working data set, using the variable names that correspond with those in the permanent data files. Then, PROC UPDATE or the MERGE statement in a DATA step can be used to add the new information to the permanent file. Because of the abundance of SAS programmers on campus, and the straightforward nature of the tasks, the entry of each new year's data can be performed by student programmers as soon as the new information is received by the Agricultural Engineering Department.

Programmers working on the NHIMS system will have the ability to add new files to the data base, and also to add or modify the processing functions available to the users. The data retrieval portion of the system will be general enough that files containing different elements or files with information from other states can be added to the system, as long as they contain the same type of information as the current data sets. Similarly, the processing functions will be designed to work on files of a general structure, so that any data set with the same characteristics will qualify as input.

### C. System Design Guidelines

Modularity is the key to designing a maintainable system. If the functional elements of a software system are separated into modules, with strict limits on intermodule communication, then changes can be made to the individual modules without affecting the operation of the entire system. Within the SAS software system, the macro language facility can be used to create a modular system. The user commands, such as ACCESS, ELEMENT, STATION, etc., can be implemented as individual macros which perform the necessary data retrieval and processing functions. Thus, the same user command language can be utilized, but with the feature that new commands can be easily added to the system via macros.

To improve the readability of the code, certain programming conventions will be followed. Meaningful variable and data set names will be used, as meaningful as possible within the constraint of 8 characters. A standard indentation scheme will be implemented and consistently followed. There will be only one source statement per

physical line, and blank lines will be inserted to separate the different code segments. Meaningful documentation can be obtained by inserting comments at the beginning of each DATA and PROC step, and at appropriate locations within the steps. LABEL statements already have been used to describe all variables in the permanent data sets, allowing the descriptions to be retrieved at any time with PROC CONTENTS. Above all, the programming style must be consistent and clear, and the programming logic must be easy to understand.

## VII. FUTURE PLANS

At this point in the project, the HISARS data files have been converted to permanent SAS data sets and are ready for use, with full documentation available. Programs have been written to perform sample data retrievals, proving the ability of the POINT option to do direct access. A small-scale simulation has successfully shown that users will be able to employ the existing command language to retrieve information, even though SAS software is now performing the internal operations.

The final steps of the overall project, which will be completed in the coming year, are to finish the design of the NHIMS programs and to implement the program code. A detailed design document will be written, which will completely outline how the programs are to be coded. Once the programs have been written, a thorough testing of the system will be conducted; both unit testing of individual modules, and integration testing of the entire system will be performed. Two types of manuals will be prepared to show how to use the new system: one directed to the general user, and another directed to SAS programmers. In addition, upon project completion, workshops will be held to instruct users on how to retrieve hydrological data from the NHIMS system.

## VIII. REFERENCES

- Deitz, Cynthia and C. William Cox, Jr. 1980. Efficiency comparison of SAS with PL/I for programming a client monitoring system in a community mental health center. Proceedings SUGI, pp. 122-128.
- Doyle, W. Harry, Jr. and Joy A. Lorens. 1982. Data management system for USGS/USEPA urban hydrology studies program. USGS Open File Report OF 82-0442.
- Driver, Nancy E. and Martha H. Mustard. 1984. U.S. Geological Survey urban-hydrology data base for 22 metropolitan areas throughout the United States (abstract). EOS 65(45): 885.
- Finley, Wayne B. and William C. Pannell. 1982. A SAS based management information system. Proceedings SUGI, pp. 266-270.
- Molnau, Myron. 1975. A guide to the use of HISARS. Idaho Agriculture Experiment Station Misc. Series No. 32.
- Molnau, Myron. 1983. Climate and hydrology data for Idaho. Idaho Agriculture Experiment Station Misc. Series No. 32(rev).
- Nungesser, Martha K. and Richard J. Olson. 1980. Regional ecological analysis and data base applications. Proceedings SUGI, pp. 45-51.
- SAS Institute Inc. SAS User's Guide: Basics, 1982 Edition. Cary, NC: SAS Institute Inc., 1982.
- Wiser, Edward H. 1975. HISARS, Hydrologic information storage and retrieval system. North Carolina Agriculture Experiment Station Technical Bulletin No. 215.