

842 444 3

4.3
3
.24
2.2

NOAA Eastern Region Computer Programs
and Problems NWS ERCP - No. 24



WXR

Harold Opitz
Ohio River Forecast Center
Cincinnati, Ohio

Scientific Services Division
Eastern Region Headquarters
August 1984

N.O.A.A.
U. S. Dept. of Commerce

AUG 29 1984

**U.S. DEPARTMENT OF
COMMERCE**

National Oceanic and
Atmospheric Administration

National Weather
Service

NOAA TECHNICAL MEMORANDUM

National Weather Service, Eastern Region Computer Programs and Problems

The Eastern Region Computer Programs and Problems (ERCP) series is a subset of the Eastern Region Technical Memorandum series. It will serve as the vehicle for the transfer of information about fully documented AFOS application programs. The format ERCP - No. 1 will serve as the model for future issuances in this series.

- 1 An AFOS version of the Flash Flood Checklist. Cynthia M. Scott, March 1981. (PB81 211252).
- 2 An AFOS Applications Program to Compute Three-Hourly Stream Stages. Alan P. Blackburn, September 1981. (PB82 156886).
- 3 PUPPY (AFOS Hydrologic Data Reporting Program). Daniel P. Provost, December 1981. (PB82 199720).
- 4 Special Search Computer Program. Alan P. Blackburn, April 1982. (PB83 175455).
- 5 Conversion of ALEMBIC\$ Workbins. Alan P. Blackburn, October 1982. (PB83 138313).
- 6 Real-Time Quality Control of SAOs. John A. Billet, January 1983. (PB83 166082).
- 7 Automated Hourly Weather Collective from HRR Data Input. Lawrence Cedrone, January 1983 (PB83 167122).
- 8 Decoders for FRH, FTJ and FD Products. Cynthia M. Scott, February 1983. (PB83 176057).
- 9 Stability Analysis Program. Hugh M. Stone, March 1983. (PB83 197947).
- 10 Help for AFOS Message Comp. Alan P. Blackburn, May 1983. (PB83 213561).
- 11 Stability and Other Parameters from the First Transmission RAOB Data. Charles D. Little, May 1983. (PB83 220475).
- 12 TERR, PERR, and BIGC: Three Programs to Compute Verification Statistics. Matthew R. Peroutka, August 1983. (PB84 127521).
- 13 Decoder for Manually Digitized Radar Observations. Matthew R. Peroutka, June 1983. (PB84 127539).
- 14 Slick and Quick Data Entry for AFOS Era Verification (AEV) Program. Alan P. Blackburn, December 1983. (PB84 138726).
- 15 MDR--Processing Manually Digitized Radar Observations. Matthew R. Peroutka, November 1983. (PB84 161462)
- 16 RAMP: Stability Analysis Program. Hugh M. Stone, February 1984.(PB84 1614
- 17 ZONES. Gerald G. Rigdon, March 1984. (PB84 174325)
- 13 Automated Analysis of Upper Air Soundings to Specify Precipitation Type. Joseph R. Bocchieri and Gerald G. Rigdon, March 1984. (PB84 174333)

H
QC
874.3
U63
no.24

NOAA EASTERN REGION COMPUTER PROGRAMS AND PROBLEMS - No. 24

WXR

//

Harold Opitz
Ohio River Forecast Center
Cincinnati, Ohio

Scientific Services Division
Eastern Region Headquarters
August 1984

WXR

Harold Opitz
Ohio River Forecast Center
Cincinnati, OH

I. General Information

A. Summary

WXR is a hydrologic weather wire and PLT product software package which manipulates decoded SHEF hydrologic data according to runtime command line options. WXR coordinates execution with various master format parameter files. The master format parameter files contain the actual output format for specific weather wire products and are assembled during an initial setup phase. Multiple master files may be assembled for various output formats. The operator decides, via command line input, which file(s) WXR will use.

WXR will also create, at the operator's option, a hydrologic PLT product. The PMOD package can then be used to transform this into a graphic.

WXR uses three other programs: one in the initial setup phase (NET.SV), and the other two to collect and decode the input products from the database (NSEEK.SV and WXRPARSHEF.SV). NET is a dictionary/database utility program that will assemble and alphabetize station information into an RDOS disk file. The information contained in this NET.DT file is designed for use by WXR, however, it may be useful for other programs as well. NSEEK is a multi-purpose utility program that will assemble an RDOS disk file containing any specified AFOS SHEF-coded product(s) and invoke the SHEF decoder (WXRPARSHEF - this is the S/230 version of the standard SHEF decoder, see Appendix A). NSEEK also include a debug routine that will read the SHEF binary output file for display on the AFOS console. This debug routine allows the operator to examine the actual contents of the SHEF binary output file.

Each program has a complex command line with several processing options. Probably the best approach for routine use is to construct a macro.

B. Environment

All of these programs are written in Data General FORTRAN IV for the Eclipse S/230. They will run either in foreground or background (depending on the installation).

C. References

The author wishes to acknowledge Daniel P. Provost (OHRFC) for initial work performed on the structure of the format parameter files for the re-work of PUPPY to include SHEF code and user-defined weather wire product formats.

Bonin, G. M. and R. S. Cox, Jr., 1983: An Explanation of the Standard Hydrologic Exchange Format (SHEF) and its Implementation in the Central Region. NOAA Technical Memorandum NWS CR-67, NOAA, U. S. Department of Commerce.

Lee, A. B., 1981: PULL.FR. Applications Programs for AFOS No. 2, WSFO Columbia SC, National Weather Service, NOAA, Department of Commerce.

MacDonald, A. E., 1981: AFOS Graphics Creation from FORTRAN. NWS Western Region Computer Programs and Problems NWS WRCP No. 18, NOAA, U. S. Department of Commerce.

National Weather Service, 1982: Standard Hydrologic Exchange Format (SHEF) Version 1 (Working Draft). Unpublished manuscript, Northwest River Forecast Center, National Weather Service, NOAA, Department of Commerce.

National Weather Service, 1983: PMOD Plotting Package for AFOS. NOAA AFOS Operating Division Computer Programs NWS AOD CP 83-1, NOAA, U. S. Department of Commerce.

Provost, D. P., 1981: PUPPY (AFOS Hydrologic Data Reporting Program). NWS Eastern Region Computer Programs and Problems NWS ERCP No. 3, NOAA, U. S. Department of Commerce.

II. Application

A. NET

When initially setting up the WXR package, NET is run first to create a station information file called NET.DT. The file contains the station call letters, the station name, the latitude and longitude and the zoom setting. NET may be re-run subsequently to add entries to or delete entries from an existing NET.DT.

The raw station data may be input to NET either in an AFOS product or an RDOS file created with one of the editors. The operator indicates which to use via a command line switch.

The add and delete options are independent of each other: one may not add and delete simultaneously. The add function will add and alphabetize the specified station information to a currently existent NET.DT file (if one does not exist, NET will create one). The delete function will only work on a current NET.DT file.

NET will always create a NET.DT backup file, NET.BU. The operator may save the backup file in the event of anticipated add/delete/duplication errors (duplicate entries in the input cause NET to terminate). NET also creates various intermediate scratch files. All scratch and backup files may be saved via runtime options. However, by default, NET deletes all scratch and backup files when it is finished.

NET makes use of the following modules:

- INTHVG - integral halving routine
- ALBT - alphabetizing routine by numeric index
- FPREP - file preparation routine (Used extensively when an AFOS product is to be used as input, checks for parameter consistency. AFOS retrieval routine PLLDATA dumps data to this module.)
- PLLDATA - AFOS database retrieval routine (writes product to an RDOS disk file)
- GAFOS - Front end to PLLDATA
- WRADD - Add routine
- WRDEL - Delete routine

B. NSEEK

Once a NET.DT file has been created, NSEEK is run to collect specified SHEF-coded AFOS products into an RDOS file (SHEFIN) and invoke the SHEF decoder. Either current or previous versions can be accessed, and several different products can be combined into one file. NSEEK accomodates products prepared with preformats. The SHEF decoder WXRPARSHEF takes SHEFIN as input and returns the decoded data in binary form in file SHEFOUT.

NSEEK uses the following modules:

- TONUM - Converts the byte/block counters to integer numbers
- TOOCT - Converts integer numbers to unpacked ASCII
- PLLDATA - AFOS database retrieval routine
- OUTPUT - Reads the binary SHEF output file SHEFOUT and writes formatted output to disk file OUT.DT (for ADM display - debug option)

C. WXR

The last step is to run WXR to create the desired hydrologic weather wire products and/or plotfiles from the decoded data in SHEFOUT. WXR references user-created format files to properly format the weather wire products.

Execution begins by opening the program's overlay disk file and passing control to subroutine RDCLI. RDCLI reads and checks the command line options: global data filter and plot flag switches and two local arguments. The global data filter switches determine what types of SHEF data WXR will process. The WXR program currently uses five SHEF data attributes: station ID, observation time, SHEF two character physical element, element type (T of PEDTSEP) and the data value itself. The switch controls act on observation date, observation time and element type. They are optional -- if not present, WXR will default to 12Z observation time, current observation date and type-R data. The plot flags indicate whether WXR should also (or only) produce a plotfile from the input data. Local switches and arguments are used to select the proper format file(s) and optionally to indicate how many columns are desired in the output.

A file maintenance routine MAOPN is utilized next. MAOPN opens the disk format file specified in the command line. All formatted output is sequentially written to a temporary disk file, WXR.99.

Program control is subsequently passed to an executive routine, EXCTV. This routine directs control and calls the appropriate modules (PILL, WOISS, WRCOM, FORMAT, LTRS and TAFOS) according to the currently active format file. Upon completion, the temporary disk file WXR.99 is stored into the AFOS database via TAFOS. The process begins again if multiple format files have been entered via the local /F switch in the command line.

Following are descriptions of WXR modules. The main program opens the overlay file and loads module EXCTV only.

WXRCOM -block data module containing all labeled common areas
INTHVG -integral halving module
RDCLI -reads and interprets the command line, checks consistency of global switches
DATIM -Calculates the day of the week and creates the unpacked ASCII equivalent.
MAOPN -file maintenance module: checks disk status for SHEFOUT and NET.DT, opens format file specified in command line and prepares the output file WXR.99 (When using multiple format files, WXR.99 is deleted after successful storage in AFOS and recreated for the next output product.)
EXCTV -directs execution path according to user format file and initiates AFOS database storage. Execution is also directed to MAOPN if multiple format files are used.
PILLL -Read, check and write a proper AFOS product header and addressee. Characters following the CCCNNNXXX and addressee in the PIL line of a format file are interpreted as a WMO header and passed unaltered to the output file. PILLL appends a date-time group to the last set of characters and writes the whole line to the output file.
WRCOM -Read and write unaltered lines to the output file
WRDAT -Same function as WRCOM except when it encounters the following independent letter combinations; then it substitutes the corresponding alphanumeric/numeric combinations for the current date. For instance, on August 7, 1983, WRDAT would output these equivalents for the letter combinations:

MO - 08	D0 - 07	Y0 - 83
M1 - 8	D1 - 7	Y1 - 1983
M2 - AUG	D2 - SAT	
M3 - AUGUST	D3 - SATURDAY	

WRTIM -Performs the same function as WRCOM/WRDAT for time and translates the following two letter combinations:

TZ - 12	(Zulu hour)
---------	-------------

TL - 7 (local hour)
TM - 36 (minute)

- FORMT - Reads and stored the output format for SHEF decoded data specified by the user. There are four buffer positions for each format element. A maximum of eight SHEF physical elements is allowed for output. The SHEF two-character physical element is held in the first position. The second position contains the actual output field width for the physical element specified in position one. The third contains an accumulative total of all previous field widths, inclusive. The fourth position is a process flag with the following possible values:
- Ø - no change to data value
 - 1 - compute 24 hour change for this element
 - 2 - extract the most significant part of a data value (drop the decimal positions)
 - 3 - perform both options 1 and 2
- Errors in the format line are fatal. Field widths/ total line length may not exceed 72 characters (exclusive of carriage return/line feed).
- CHVRT2 - Converts unpacked ASCII characters into a single precision data value.
- LTRS - Sequentially reads (left to right, then top to bottom) all contiguous character sets (call letters) and validates them against NET.DT. LTRS will match up to and including 8 characters. An operator warning is sent to the output console if the contiguous set exceeds this. A mismatch of characters produces no response, LTRS proceeds to the next set. The order of input is the order of output.
- LDNET - Reads in the NET database file (NET.DT) station ID's and also initializes the station data stack counters and pointers.
- WRMSG - Upon encountering an array of station call letters that either do not match the current NET database or are not associated with any valid physical element, WRMSG will write the message "DATA UNAVAILABLE" to the output file.
- LODYS - Loads disk file data into non-current data buffer (initiated by global /C)

- WXALPHA -Determines the number and order of valid stations for output. Each output station must match at least one physical element in the current format buffer. This routine also determines the output order for column sectoring (local /C).
- WRALPHA -Routine for fomatted output to file WXR.99. Also perfomrs 24-hour change computations (see C24HR).
- LDSHF -This module reads the SHEF binary file SHEFOUT. All characteristics of a decoded SHEF data value are read in. However, only the 8-character ID, the observation date/time group, physical element, physical element type and the actual data value are utilized for WXR's data filters. Duplicate data values are overwritten by the latest data value for all global switches. The applicable global switches are as follows:
- /C Load current date, 12Z type R data and/or change computation
- /O Load current date, any time type R data. No change computation or disk file update performed.
- /A Load any date, any time and any type data. User becomes the data filter
- Reference C24HR and LODYS routines.
- REOCT -Utility to convert a single precision number into a 16 byte unpacked character array in reverse order. Used extensively by WRALPHA.
- C24HR -Locates and matches a corresponding data value and SHEF physical elements from the non-current data buffers for a specified station (see LODYS). It will pass this value and an error condition to WRALPHA.
- TAFOS -This routine performs two functions: it adds the necessary 203 octal AFOS end-of-text to the current output file, and it uses FSTORE (BG.LB) to store that file (WXR.99) into the AFOS database.
- WXPLT -The PLT executive driver routine. This module utilizes support modules DATIM, REOCT, PLTHDR, CART, LLCVT and PLACE and performs the actual storage of the plotfile into the database.

- PLTHDR -Writes the plotfile header information and performs
PLT dummy disk file maintenance.
- LLCVT -Converts valid station coordinates (lat/lon) to pixel
values. It also checks the stack pointer for valid
data. If a particular station does not have any SHEF
data associated with it (stack pointer=0), then the
station is ignored (not written to plotfile).
- CART -Does the actual lat/lon-pixel conversion for LLCVT.
(See WRCP No. 18 in References, I.C)

D. Machine Requirements

WXR requires a minimum of 29K words of core. The program files together take up 304 blocks of disk space (NET.SV - 40 blocks - does not have to reside on the disk for routine use). Space requirements for data files will of course vary with the application. A NET.DT file for 53 stations took up 4 blocks. The format parameter files and .PM files for the plotfiles are all relatively small.

WXR's run times will also vary with the amount of data it is to process. Creating a simple weather wire product and plotfile with 12Z data for the above NET.DT file took about 1 1/2 minutes.

Only WXR uses an overlay file. NET, NSEEK and WXR allow 25 channels to be open.

E. Structure of Software

See Figures 1-3 for flowcharts.

F. Database

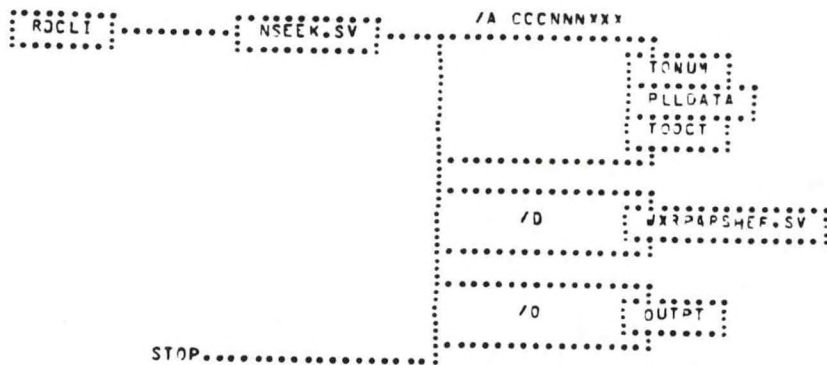
Input to NET can be stored in any RDOS file (filename up to 18 characters long) or AFOS product as long as the data is in the proper format (see Section III A). One dictionary/database file and three backup scratch files are used by NET when manipulating station information. They are:

NET.DT The end result, a database/dictionary station information file. It contains station call letters, alphanumeric station names, station latitude and longitude and a zoom level for plot purposes. Each record is 39 bytes in length terminated by a carriage return (15K). Maximum



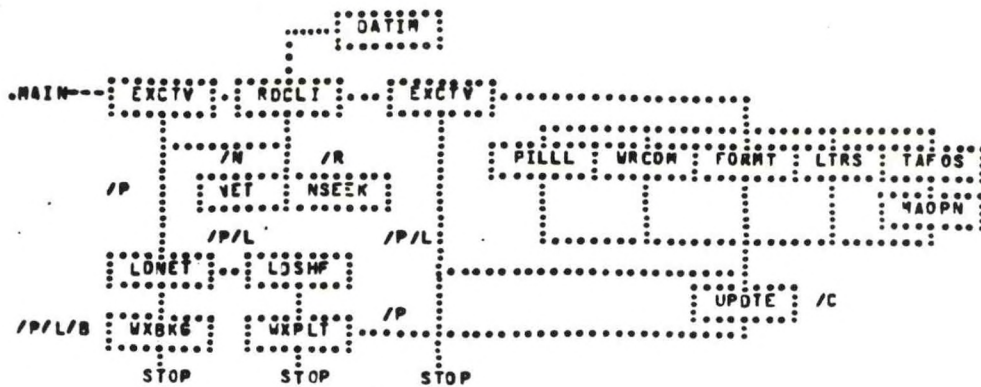
THE NET.SV EXECUTION PATH VIA WXR COMMAND LINE

Figure 1. NET Execution Path



EXECUTION PATH OF NSEEK.SV VIA THE WXR COMMAND LINE

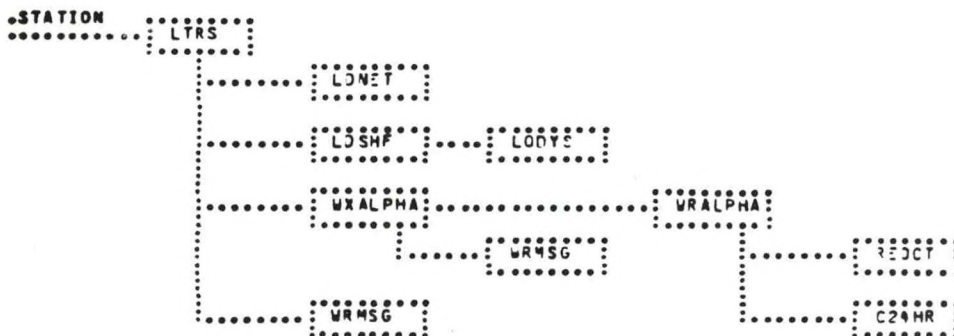
Figure 2. NSEEK Execution Path



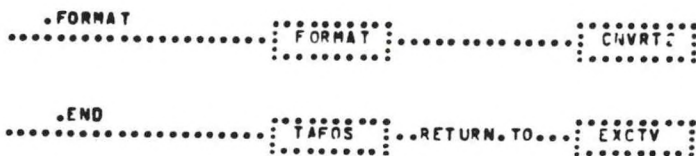
WXR FLOW CHART OF EXECUTION PATH ACCORDING TO GLOBAL SWITCH SETTINGS. EXECUTIVE ROUTINE, EXCTV, CONTROLS PROGRAM PATH AND DETERMINES THE SPECIFIC ROUTINES CALLED.



THE .PIL AND .COMMENT (DATE) COMMAND PATHS. ONCE COMPLETED, CONTROL IS RETURNED TO THE EXECUTIVE ROUTINE, EXCTV.



THE .STATION COMMAND INVOLVES A SERIES OF ROUTINES. THE WRITE ROUTINE, WRALPHA, AND TIME-CHANGE COMPUTATION ROUTINES ARE LOCATED HERE.



THE .FORMAT AND .END COMMANDS. CONTROL IS RETURNED TO EXCTV UPON COMPLETION.

Figure 3. WXR Execution path, showing responses to commands in format files.

characters allowed are:

call letters	8 (minumum 1)
alphanumeric name	20 (not contiguous characters)
latitude	4 (degrees and minutes, no decimal)
longitude	5 (" " " " ")
zoom	1 (0,1,2,3,4)
total	39 (plus carriage return 15K)

NET.BU Backup file of original NET.DT created during the add or delete process. Deleted unless the global X or global S switch is used.

NET.SC Scratch file.

NET.TM Scratch file for CCCNNNXXX or FILENAME/F.

Input to NSEEK is the AFOS product specified in the run line. NSEEK (and WXRPARSHEF) also utilize the following files:

SHEFIN This ASCII file contains the assembled CCCNNNXXX's and is the input file for the SHEF decoder.

SHEFOUT The SHEF decoder, WXRPARSHEF.SV sends decoded SHEF data to this file. SHEFOUT is a binary data file.

SHEF.ER All data that has passed through the SHEF decoder is passed to this file including all SHEF errors. This is the actual SHEF error file.

OUT.DT This is a AFOS ADM displayable file containing the ASCII output of the binary data in SHEFOUT. Allows the operator to visually scan the contents of SHEFOUT.

SHEFPARM This is a utility file for the SHEF decoder.

The following is a listing of all the various files (mandatory and optional) utilized by WXR.

NET.DT An alphabetical master database file containing all station information.

SHEFOUT The binary output file from the SHEF decoder. WXR will read SHEFOUT to obtain all SHEF data for the current run.

WXR.* The asterisk template represents any single character, A through Z, or single number 1 through 9, i.e., WXR.A, WXR.1, or WXR.8. This is the format parameter file.

- WXR.TD Used in conjunction with global /C switch. All data values on the data stacks during execution are stored in this file prior to termination for use at a later time. The file holds a date, data pointers, and the actual SHEF two-character physical elements and its associated data values.
- WXR.YS Used in conjunction with global /C switch. Generally, data in this file will be used to calculate change computations. Same format as WXR.TD
- WXR.99 A temporary sequential data output file and is used by the FSTORE library module to store its contents into the AFOS database. This scratch file is deleted when storage is completed.

III. Procedures

A. NET

1. Initiation of program

NET will not be used routinely, only when a NET.DT file must be created or altered. NET.SV is invoked by the following command line:

WRX/N/A/D/S/X CCCNNNXXX (or) FILENAME/F

Where global switches imply:

- /N Utilize NET.SV functions (mandatory)
- /A Add the stations found in CCCNNNXXX or FILENAME to NET.DT. If NET.DT is not existent, build a NET.DT file.
- /D Delete the stations in CCCNNNXXX or FILENAME from the current NET.DT file.
- /S Do not delete the NET.BU backup file. (Used with /A or /D)
- /X Save all temporary, scratch and backup files utilized by NET.SV (used with /A or /D). This is helpful in determining any errors encountered during NET execution.

CCCNNNXXX and FILENAME are the AFOS product or RDOS file containing the raw station data (FILENAME can be up to 18 characters long).

2. Input Required

The user can enter the information for the stations whose data is to be used in the weather wire or PLT products either via message comp or one of the RDOS file editors. NET can handle up to 500 station entries. The file should conform the following format:

-the first line must begin with .START

-each station entry takes up one line (72 character max). All character fields must be entered. (If a field is not completely filled by the information, pad with blanks.) Each character field must be separated by at least one space.

The character fields are as follows:

STATION CALL LETTERS: Maximum 8 characters. No embedded blanks.

ALPHANUMERIC NAME: Maximum 20 characters and blanks may be used. It is recommended that the station's flood stage be entered here since there is no separate entry for flood stage. (i.e., Louisville, KY 23; where 23 is the flood stage and the "3" resides in the last portion of the alphanumeric field (Position 20). Refer to Figure 4.)

LATITUDE: Maximum 4 characters. Enter degree minutes with no decimal so that 45 degrees, 24 minutes would be entered 4524.

LONGITUDE: Maximum 5 characters. Also enter the same as latitude with no decimals, i.e., 101 degrees, 25 minutes is entered as 10125.

ZOOM: Maximum 1 character that is either 0,1,2,3,4. Entries 0-3 display stations at certain zoom settings at the AFOS GDM console. WXR recognizes that certain stations may not need to be plotted, for whatever reason. Therefore, entering a zoom factor greater than 3 will instruct WXR not to post that particular station to the plotfile.

-the last line must be .END.

See Figure 4 for sample input. The same format should be used when adding additional stations to an existing NET.DT file.

When deleting, however, only the station call letters are needed for the station entries. More than one set of call letters can appear on a line, as long as they are separated by a space. ID's more than 8 characters long are ignored, as well as those not matching entries in the current NET.DT. .START and .END must still be used as above. Figure 5 shows sample input for deleting.

To change station information, one must first delete and then re-enter the particular station entry.

3. Output

The output is the file NET.DT. See Figure 6. Error messages from NET appear in Appendix B.

4. Cautions or Restrictions in Program Use

It would be a good practice to always include the global S-Switch in any execution run in order to save the NET.BU file for backup purposes in the event execution is not successful. It is also better to execute NET in the same directory as the NET.DT and .BU files if /S is used, since the renaming process doesn't work well with links.

Duplicate station call letters in the input are not allowed and terminate execution (NET.BU then becomes the new NET.DT). Deleting stations not in the current NET.DT is ignored by the program.

The add and delete functions (/A and /D) may not be executed at the same time. An error will result if this is the case. To change station information for any particular parameter, the entire entry must first be deleted and subsequently re-added to the NET.DT database/dictionary file. The total number of stations may not exceed 500.

B. NSEEK

1. Initiation of Program

CRWWRK003
THIS IS A TEST FILE

.START				
ACR01	ALUM CREEK	FS	C100	85 113
ATC01	ATWOOD LAKE	FS	C100	22246
ATH01	ATHENS	FS	C000	97256
BCH01	BEACH CITY	FS	C100	48228
BHD01	HARSHA DAM	FS	C100	29202
BOL01	BOLIVAR DAM	FS	C100	98210
BJR01	BURTON	FS	C000	99199
BVL01	BOURNEVILLE	FS	C010	91189
CAK	AKRON-CANTON	FS	C0100	99299
CCL01	CEASAR CREEK	FS	C0100	45125
CCV01	CIRCLEVILLE	FS	C0100	92120
CJ101	CAMBRIDGE	FS	C0199	92221
CHC01	CHILLOCOTHE	FS	C0100	92179
CHM01	CHARLES MILL	FS	C0100	93296
CHN01	COSHOCTON	FS	C0199	95253
DAY	DAYTON AP	FS	C0100	99199
DAY01	JAYTON	FS	C0100	23174
FT101	FITCHVILLE	FS	C000	99199
FTJ01	FT JENNINGS	FS	C0099	92110
HAM01	HAMILTON	FS	C100	91184
.END				

Figure 4. Sample input for NET in AFOS product CRWWRK003. Comments may be placed after the .END command since the program stops there. This is helpful for future references.

```
.START
CIV01
CAERR5YN
INDI3
EVVI2
.END
```

Figure 5. Sample input for NET to delete station entries from NET.DT.


```

SSSSSSSSAAAAAAAAAAAAAAAAAAAAAALATILONGIZ
ACR01  ALUM CREEK          FSC100851132
ATD01  ATWOOD LAKE        FSC100222463
ATH01  ATHENS              FSC000972561
BC401  BEACH CITY           FSC0100482283
BHJ01  HARSHA DAM            FSC0100292022
BOLO1  BOLIVAR DAM          FSC0100962103
BUR01  BURTON                FSC0000991991
BVL01  BOURNEVILLE        FSC0100911992
CAK    AKRON-CANTON         FSC100992993
CCL01  CEASAR CREEK        FSC0100431252
CCV01  CIRCLEVILLE        FSC0100921202
CDIO1  CAMBRIDGE            FSC9199922213
CHC01  CHILLOCOTHE         FSC0100921792
CHM01  CHARLES MILL         FSC0100932963
CHV01  COSHOCTON           FSC9199932533
DAY    JAYTON AP            FSC0100991992
DAY01  DAYTON               FSC0100251742
FTH01  FITCHVILLE         FSC0000991991
FTJ01  FT JENNINGS         FSC9099921101
HAM01  HAMILTON            FSC0100911942

```

Figure 6. NET output file NET.DT generated from input in Figure 4 (command line RUN:WXR/N/A CRWWRK003). Notice it is in compressed in order to save disk space. The header line in record 0 specifies field length.

NSEEK will be used routinely to collect products from the database and invoke the SHEF decoder, WXRPARSHEF to decode them.

The NSEEK command line runs as follows:

```
RUN:WXR/R/A/D/O CCCNNNXXX */V
```

Global Switches:

- /R Utilize NSEEK functions (mandatory)
- /A Append CCCNNNXXX to the current SHEFIN disk file. If the /A switch is not entered, the current file is deleted and the product is written to a new one.
- /D Invoke the SHEF decoder package WXRPARSHEF.SV. The SHEF decoder will operate on the current SHEFIN file containing the CCCNNNXXX('s).
- /O Debug the current SHEF decoded binary output file for display on an AFOS ADM console (produces file OUT.DT which may be displayed on an ADM.)

All global switches are independent of each other except for the /A switch which must be used with a CCCNNNXXX. Therefore, all switches will operate on whatever the current disk file contains.

CCCNNNXXX indicates product NSEEK is to retrieve and store in SHEFIN file. (Any length up to 9 characters.)

*/V The asterisk denotes a number 1-9 and specifies the AFOS version number for CCCNNNXXX. For a current version AFOS product, enter nothing. First previous version is 1/V, second previous version is 2/V, and so on, to a maximum 9th previous version.

2. Input Required

All NSEEK requires is an AFOS product in SHEF code (see SHEF manual). Only one product is processed per NSEEK run. NSEEK assembles them into a file called SHEFIN, which WXRPARSHEF, the SHEF decoder, uses as input. (WXRPARSHEF also requires the SHEF parameter file, SHEFPARM, to operate.) See Figure 7 for an example of SHEFIN.


```

:001392
.B CHI 0203 DH1200/HG                               :FIRST DATA ROUNDUP CHICAGO
WMTW3 4.93, ALG12 1.46, GUN12 1.7                 :WILMOT/ALGONQUIN/GURNEE
RVRI2 3.3, PNT12 M, MOR12 M                       :RIVERSIDE/PONTIAC/MORRIS
.EVJ
.B CHI 0203 DH1200/PPP                               :24 HOUR PRECIPITATION TOTALS
CSNI2 0.0, WSLI2 0.0, HSBI2 0.0                  :CISNE/WEST SALEM/HARRISBURG
.EVJ
.B CHI 0203 DH1200/PPP                               :24 HOUR PRECIPITATION TOTALS
CSNI2 0.1, WSLI2 0.0, HSBI2 0.0                  :CISNE/WEST SALEM/HARRISBURG
.EVJ

```

Figure 7. Example of the SHEFIN file created by NSEEK. Notice line 1 contains block and byte pointers. These are necessary for NSEEK to append additional products to the file. The colon designates a comment to the SHEF decoder, therefore the first line is ignored.

STN ID	YYMMDD	HHMM	-YYMMDD	HHMM	PE	DJR	TSE	PROB	ACTL	VALUE	GR	SOURCE
ALG12	84	2	312	00	00	00	00	00	00	1.46	ZOCHI	0
GUN12	84	2	312	00	00	00	00	00	00	1.70	ZOCHI	0
RVRI2	84	2	312	00	00	00	00	00	00	3.30	ZOCHI	0
PNT12	84	2	312	00	00	00	00	00	00	-9999.00	ZOCHI	0
MOR12	84	2	312	00	00	00	00	00	00	-9999.00	ZOCHI	0
CSNI2	84	2	312	00	00	00	00	00	00	0.00	ZOCHI	0
WSLI2	84	2	312	00	00	00	00	00	00	0.00	ZOCHI	0
HSBI2	84	2	312	00	00	00	00	00	00	0.00	ZOCHI	0
CSNI2	84	2	312	00	00	00	00	00	00	0.00	ZOCHI	0
WSLI2	84	2	312	00	00	00	00	00	00	0.10	ZOCHI	0
HSBI2	84	2	312	00	00	00	00	00	00	0.00	ZOCHI	0

Figure 8. An OUT.DT file containing the ASCII contents of the binary SHEFOUT file. OUT.DT may be displayed on the ADM and is useful in checking for errors in the decoding process. This particular example was decoded from the SHEFIN input in Figure 7.

3. Output

WXRPARSHEF creates a binary file of decoded data called SHEFOUT. Optionally (/O switch) it also produces OUT.DT, an ADM-displayable file used for detecting any processing problems. See Figure 8 for a sample OUT.DT. Error messages from NSEEK appear in Appendix C.

4. Cautions and Restrictions

There must be a period (56K) preceded by a line feed (12K) or a carriage return (15K) within 256 bytes of the first AFOS data block for NSEEK to successfully retrieve and write the entire AFOS product to the disk file. NSEEK will write or append a maximum total of 999 full data blocks to an RDOS diskfile before beginning a disk file overwrite. Disk block usage will vary significantly from office to office since it is dependent upon the amount of AFOS products being used for SHEF material.

WXRPARSHEF.SV (and SHEFPARM) must be on disk for NSEEK to invoke the decoding process properly.

C. WXR

1. Initiation of Program

Once the collected SHEF data has been decoded, WXR is used to generate the final output products.

The WXR command line takes the form of:

```
RUN:WXR/A/C/O/P/L/B    */F    */C
```

The global switches consist of both data filters and PLT processing flags. The PLT flags instruct WXR on whether or not to create a plotfile. The data filters instruct WXR on the type of data to read into WXR's data stacks. The first argument (/F switch) is mandatory; it instructs WXR about which format file(s) to open and use for producing a weather wire product. The second argument (/C switch) is an optional column sectoring switch. These arguments may be entered in any order.

Global Switches:

The data filter switches (A,O,C) may not be used simultaneously. They instruct WXR on the type of data to load from the decoded SHEF file. The five SHEF attributes used by WXR are:

- station call letters (max of 8 contiguous characters)
- observation date and time (month,day,year,hour,minute)
- SHEF two-character Physical Element (PE of PEDTSEP)
- the "Type" code (T of PEDTSEP)
- the actual data value itself

(For further inquiries about SHEF data attributes, see SHEF Manual, Reference Tech Memo CR-57)

The data filter switches perform the following functions when loading SHEF data onto WXR's data stacks:

- /A This switch allows any type of data to be loaded onto WXR's data stacks. In this mode, all attributes are ignored and the operator becomes the data filter.
- /O The O-Switch only allows current observation date, any observation Z-time, type "R" data to be loaded onto the data stacks.
- /C Load 12Z observation time, current observation date, type "R" data. This global switch performs three important supplemental functions concerning data processing. It tells WXR to save the current data buffer in an RDOS disk file, WXR.TD, for use at any future time period. It also instructs WXR to read, if possible, the non-current RDOS disk file, WXR.YS, into the non-current data buffer (non-current implies 12Z data not equal to the current operating date). Therefore, WXR, may compute any desired period of change, whether it be a day, week, month, etc.

(Specifically, WXR is not limited to a 24-hour computation flag. In this way, the global C-switch does not mandate a 24-hour change computation; it only specifies that the current data buffer is saved in an RDOS disk file for use at any future time period. Time change computations are actually flagged by the global C-Switch in conjunction with a change computation flag located in the format parameter file format line. Hence, the disk file may be updated with all of WXR's data buffers without computing any change at all.)

The same aspects of data filtering relative to the global

/C switch are performed when no global data filter switches are entered.

(Technical Note: The default time window for WXR is 12Z +/- 1 hour or 11Z through 13Z. Any data values with SHEF observation times that are within the time window will be loaded onto WXR's data stacks. The range of the time window and the actual default Zulu hour may be modified to suit local needs. See Appendix G concerning the 12Z time window).

The PLT global switches have the following functions:

- /P This switch instructs WXR to create a plotfile after processing all weather wire products.
- /L (used with global /P) This switch will only work when used simultaneously with the global /P switch and instructs WXR to produce a plotfile only. No processing of any weather wire files will be done. If a plotfile is the only process to be executed (/P/L), the global switches /A,/C,/O will also apply when SHEF data is loaded onto WXR's data stacks.
- /B See Appendix F for specific instructions concerning this switch. It produces a special background plot.

Arguments: The two arguments (one mandatory, /F local switch and one optional, /C local switch) are used for the following tasks:

- /F The local F-Switch argument specifies to WXR which format file(s) to process. The character(s) entered in this local entry are the one-character extensions of the DOS format parameter filename(s) (WXR.*, where the * is any character between 0-9 or A-Z.)

Example: If WXR.3 is the resident format parameter file to be used by WXR, the command line would read:

```
RUN:WXR 3/F
```

This instructs WXR to follow the format command found in file WXR.3. If multiple file processing is necessary, say WXR.3, WXR.4, and WXR.D are all to be used during the same execution of WXR, the command line would read:

```
RUN:WXR 34D/F
```


This would instruct WXR to follow the formats found, in order of entry, in files WXR.3 (first), WXR.4 (second), and WXR.D (third).

A maximum of 9 single character arguments are allowed for the local /F switch. (Note: any global entry will be maintained throughout the execution period on all format parameter files.) If the /F switch is not entered, WXR terminates and displays an error message.

/C Column Sectoring: This mode instructs WXR to duplicate a format input the amount of times entered in the /C switch. In other words, WXR will columnize the weather wire product (while adhering to the maximum 72 character positions per line). If the local /C is not used, the default is one sector.

See Figure 9 for one-sector output. The format used in the figure is NM 10 PP 6 HG 6, for a total line length of 22. Figure 10 shows 3-sector output produced using the same format, but with 3/C in the command line. Note that a default of two spaces will always appear between sectors. This must be added into the total field count when computing total line length. For Figure 10, for instance, the line length total is $3*(10+6+6) + (3*2) = 66$ (3 formats) + 6 (3 two-space separators) = 72. Total field width positions, again, must not exceed 72 characters. The maximum number of sectors allowed for output is 9 (when not using the default). The same rules for valid data output, missing data output and field width/data value truncations apply to column sectoring.

The local /C switch also applies to all output if multiple format parameter files are specified in the command line.

2. Input Required

Besides the actual decoded data in the file SHEFOUT, WXR requires at least one format parameter file to indicate how you want the weather wire products formatted. These files may be created either on the ADM (M:F/) or any of the RDOS text editors. The names of the files must be WXR.*, where * is any one character A-Z or 1-9 (for example, WXR.A). Thus a number of different parameter files can be created to accommodate different weather wire products.

There are five formatting commands used in these format parameter files: .PIL, .COMMENT, .FORMAT, .STATION and .END. All formatting commands must begin with a period (56K) in column 1 followed by the appropriate command. Subsequent line(s) are information lines which contain the information to be processed. Each line in a format parameter file is allowed a maximum of 72 characters; anything beyond position 72 is ignored. Following is an explanation of each command and its associated information line(s). Refer to Figure 11 for a sample format file.

.PIL

```
.PIL
CCCNNNXXX AAA WMOHEADING (DDHHMM)
```

The .PIL line determines where the completed weather wire product should be stored (CCCNNNXXX) and the proper addressee (AAA). These two parameters are mandatory. Optional character sets following the CCCNNNXXX and AAA are considered WMO headings. After reading the last set of characters, WXR will append a date/time group (DDHHMM) to the end of the header line. A date/time group will not be appended if only CCCNNNXXX and AAA are entered. In Figure 11, the output product is CLERVACLE and the addressee is CLE. The RWRA RWUS will be used as the WMO heading with a date/time group appended.

.COMMENT

The .COMMENT command will perform a literal read and write of the subsequent information lines to the output file until it encounters another format command. The .COMMENT command is intended to allow headings and/or comments to be printed as is to the output. Hence, a blank line will be printed as a blank line, etc.

.COMMENT DATE

This command will perform exactly as the .COMMENT command, but the word DATE following it will initiate a date routine: WXR will substitute for the following character combinations in the information line(s) the appropriate form of the date or time.

M0...08	D0...05	Y0...84	TZ...12
M1...8	D1...5	Y1...1984	TL...7

```

.PIL
CLERVACLE CLE RWRA RWUS
.COMMENT DATE
NATIONAL WEATHER SERVICE CLEVELAND OHIO
PRECIPIATION AND GUAGE HEIGHT
12Z DATA FOR MO/DO/YO
STATION      PREC   STAGE
.FORMAT
NM 10 PP 7 HG 8
.STATION
NVL13 STRO1 FTJ01
FDY DEF01 DFC01 WTV01 TOL WDV01 UPS01
TI 01 FT401 H0001
COL01 CCV01 WAS01
CHC01 BVL01 PKT01 MLF01 BHDD1
.COMMENT

```

THIS IS ANOTHER COMMENT LINE USING THE SAME HEADER

```

STATION      PCPN   STAGE
.STATION
CJB01 SPRO1
DAY
EAT01
HAM01
CCL01 MAS01 CNG01 MSB01 MFD   MFD01
.END
ANYTHING AFTER THE .END COMMAND MAY BE A COMMENT OR REFERENCE

```

Figure 11. Sample format parameter file.

```

.COMMENT DATE
NATIONAL WEATHER SERVICE CINCINNATI OH
STAGE DATA FOR 03, THE 01RD OF M3
12Z DATA FOR M1/D1/YO
.FORMAT (NEXT SUBSEQUENT COMMAND)
:
:
:
ETC.

```

WOULD OUTPUT THE FOLLOWING:

```

(BLANK LINE)
NATIONAL WEATHER SERVICE CINCINNATI OH
STAGE DATA FOR FRIDAY, THE 3RD OF FEBRUARY
(BLANK LINE)
12Z DATA FOR 2/ 3/84

```

Figure 12. Example of .COMMENT DATE command and its results. All the two-character letter combinations have been converted to data representations. (BLANK LINE for illustrative purposes only.)

M2...AUG D2...THU
M3...AUGUST D3...THURSDAY

TM...36

(Given that the current date and time is Thursday, August 8, 1984 at 7:36 local time or 1236Z zulu time.) Figure 12 shows how this command works.

.FORMAT

```
.FORMAT  
NM 15 PP 10 HG 6 SD 5
```

The .FORMAT command specifies the placement of the station names and data values to the output file. The .FORMAT command allows only one line for information. There are two elements associated with each field. The first parameter is the SHEF two-character physical element, the second, the field width specifier. Currently a maximum of eight SHEF physical elements and their field widths may be used. (Remember that a total of 72 characters is allowed per line. Total all the field widths to calculate the maximum line length. In the example above, for instance, there are 36 positions allotted for data (15+10+6+5=36)). In Figure 11, 10 spaces have been allotted for the station name, 7 for precipitation (PP) and 8 for the stage height (HG).

Note that the first two-character element must be NM. This specifies the number of characters printed for a station name. The station name will be printed, left to right, with a maximum allowable total of 20 characters. If a station name contains 20 characters and only 12 are allotted, only the first 12 characters of the name will be printed. However, subsequent SHEF elements in the line may be allotted any field width as long as the line total does not exceed 72.

Within each format line, there are a series of options that directly affect data output. Consider this format line:

```
.FORMAT  
NM 14 PP 10 HG/C 10 QR 12 SD/D 5 HW/B 8
```

/C This switch instructs WXR to perform a time change computation for the corresponding SHEF element. Specifically, for HG/C WXR will compute the change in stage using previous data in WXR.YS and output that only. Field width is 10 positions. This switch should be coordinated with the global /C.

/D The /D switch instructs WXR to perform data value

truncation. A value of 24.45 will be written as 24. This switch is not associated with any global switches.

/B The /B switch instructs XR to perform both /C and /D options. Naturally, this switch must also be coordinated with the global /C.

By default, all data values are written to the hundredths position (i. e., 2 becomes 2.00). One may want to drop the decimal positions when they are insignificant or unnecessary. Missing values are decoded by SHEF as -9999.00 but are written as a single character "M" by WXR. Therefore, any values which are considered missing (-9999.00 to SHEF) are not affected by the format switch options.

Data values greater than the specified field width are flagged to the operator via the output console. Program execution will continue. WXR will place as many characters possible into the allotted field width. Unintentional character truncation due to a field width that is too small may be avoided by specifying a width large enough for the largest anticipated data value.

.STATION

.STATION

ALBR3 NMRF5 INDI3 EVVI3 YU199 EVVMO OHI34 I88 CVG IND
SSS67 CAERR56 STLI4 CAEPR123 ...

The .STATION command determines the order of station output for the weather wire product according to the last .FORMAT command. Only the station call letters corresponding to the appropriate station in the NET.DT file need be entered. Station sequence is determined sequentially from left to right, then top to bottom. This type of sequencing allows for changes/additions/deletions to station order rather easily. A minimum of 1 character and a maximum of 8 are allowed for the station call letters.

Call letters entered under a .STATION command not identified in the NET.DT file are ignored. Character sets over 8 characters in length are also ignored and flagged to the operator at the console while execution continues. A total of 500 sets of call letters may be entered per .STATION command.

Entering all possible stations under a .STATION command does not imply output for every station. WXR will only write

the wanted station information and data values under the following conditions:

- the station call letters are in the current NET.DT file
- there is at least 1 valid decoded data value for the station in SHEFOUT that matches at least one valid element in the last encountered .FORMAT line (Subsequent allocated data fields which are unmatched are left blank. There is no padding of unmatched data fields and unnecessary characters in the output data line are eliminated.)

The words "DATA UNAVAILABLE" are printed to the output file if WXR cannot match at least one valid station with at least one valid element (from the last .FORMAT line) in any .STATION listing.

.END

The .END command terminates data processing for a particular format parameter file. This command also initiates AFOS storage. However, program execution is terminated if an .END command is not encountered.

Points on constructing a format parameter file:

- Format files must be named WXR.* ,where * is any character A-Z or 0-9.
- WXR will ignore any lines in any format parameter file until it encounters the first format command line (a period in the first column followed by one of the five commands previously mentioned).
- The first format command should be a .PIL.
- The last format command should be a .END. This command initiates AFOS database storage and terminates any further reading of a format parameter file.
- At least one .FORMAT command should be entered before encountering the first .STATION command. When processing multiple format parameter files, each file must contain at least one .FORMAT command. Otherwise, WXR will not know how to place the data values to the output file.

-Multiple .FORMAT commands may be entered into any one format parameter file. WXR will use and maintain the latest specified .FORMAT line throughout the entire execution process unless changed in the file by another .FORMAT command.

-Remember, the two-character "NM" and an associated field width must be the first entry in the information line of a .FORMAT command. This "NM" informs WXR how many characters to write for a station name. The maximum field width for "NM" is 20.

-A maximum of eight SHEF parameters and their associated field widths are allowed in a .FORMAT command.

-The "NM" field is left-justified. All subsequent fields are right-justified.

3. Output

Figure 13 shows a sample weather wire product produced according to the format parameter file in Figure 11.

If the user directs WXR to create a plotfile (with the /P and/or /L switch), a plotfile is produced according to the following default parameters (these will have to be tailored to individual sites, see Appendix E for instructions):

-default key and address: CRWPLTCIN, 000

-default SHEF elements to be plotted (in order)

1. PP (REAL)
2. HG (REAL)
3. SD (INTEGER) (10 IS THE MAXIMUM ALLOWED)
4. SW (INTEGER)
5. TA (INTEGER)

-default call letters: written as they appear in NET.DT.

-default station symbol: overcast

A sample plotfile appears in Figure 14 along with an explanation of each entry. The PMOD plotting package (see References, I C) can be used to convert the WXR plotfile to graphic form. Two plot model offset files are supplied with the WXR package: WXL.PM and WXP.PM. (WXL.PM plots in line mode and is more economical in terms of graphic storage space -- see PMOD manual in References, I C). Figure 15 shows the station model these .PM files produce along with actual PMOD


```

F1 (PP) (SD) F3
F5 (TA) 0
F2 (HG) (SW) F4
NAMEXXX

```

```

LINE MODE ONLY CHAR/FIELD
<--- LINE 1, 6+6
<--- LINE 2, 5+SYMBOL
<--- LINE 3, 5+5
<--- LINE 4, MAX 8

```

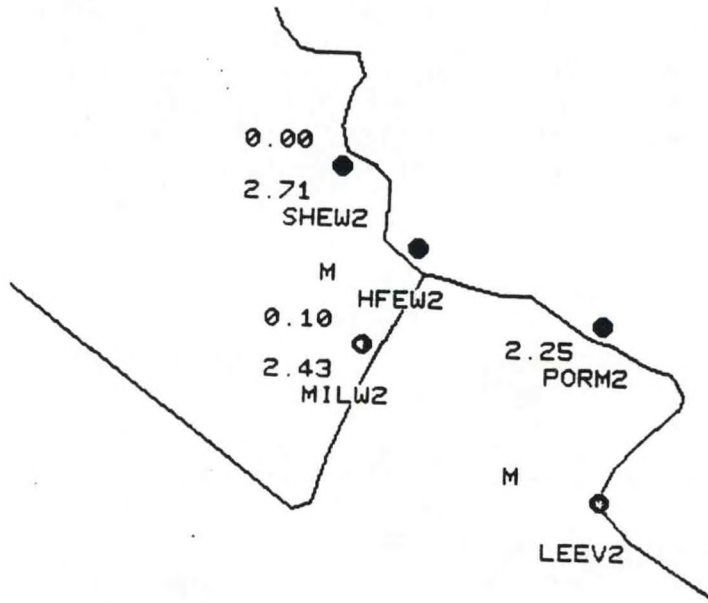


Figure 15. Station model produced by WXL.PM and WXP.PM, and some actual data (WBC area) plotted with WXL.PM.

graphic. Note that the .PM files only accomodate the 5 default SHEF elements. If others are added, new .PM files will have to be generated.

4. Cautions and Restrictions

Extensive error checking facilities have been incorporated into WXR. Processing errors are corrected during execution unless proven to be fatal. Non-fatal errors will pass an operator warning message to the output console (Dasher) while data processing continues.

The non-current data buffer reads its data from the RDOS disk file WXR.YS through routine LODYS. Non-current implies 12Z data that is not equal to the current date of operation and that the data buffer may contain SHEF data older than 24 hours. Therefore WXR may compute any desired period of change, whether it be a day, week, month, etc. Specifically, it is not limited to a 24-hour change computation. The /C global switch doesn't mandate such a computation; it only specifies that the current data buffer is saved in an RDOS file for use at any future time period. Change computations are actually flagged by the global /C in conjunction with the change computation flag in the format parameter file .FORMAT line. Thus the disk file may be updated without computing any change at all.

If the FSTORE routine returns an unsuccessful storage flag, WXR will invoke a 15 second wait period (warning to output console) and then try again. A total of four attempts will be executed (1+ minute) before proceeding to the next file (if specified).

D. Source Listings

Because of the volume of documentation, source file listings will not be included in this CP. Source files are available from Eastern Region SSD and from the AFOS Applications Library maintained by TDL.

APPENDIX A

WXPARSHEF.SV/THE SHEF DECODER DRIVER ROUTINE

All references to SHEF and the SHEF decoded software should be made to NOAA Technical Memorandum NWS CR-67 by Geoffrey M. Bonin and Robert S. Cox. The following explanation of WXPARSHEF.FR concerns the changes implemented to the FORTRAN IV PARSHEF.FR source driver routine of the SHEF decoder.

File maintenance changes have been incorporated in the FORTRAN IV WXRPARSHEF.FR only for WXR applications to facilitate file input/output. Absolutely no other changes have been made to the FORTRAN VI SHEF decoder. These input/output file maintenance changes are supported by the Ohio River Forecast Center for use by WXR applications.

FILENAME

PARSHEF.FR
TESTFILE
SHEFOUT
SHEFPARM
DOTBTEMP
\$LPT.DU

WXR APPLICATION FILENAME

WXRPARSHEF.FR
SHEFIN
(SAME - NO CHANGE)
(SAME - NO CHANGE)
(SAME - NO CHANGE)
SHEF.ER

APPENDIX B

NET Error Messages

Errors in NET execution generally occur with the add/delete function. However, there are other program errors that may occur. System errors begin with the acronym "SYS" and may be due to software or hardware failure.

NET operator error messages:

ERR: YOU CANT DO THAT	You can't add and delete simultaneously
ERR: FILNAM/CCCNNNXXX	No input filename or product key given in command line
ERR: CCCNNNXXX	Invalid AFOS product key
ERR: /F	Invalid RDOS filename
ERR: /F STAT	Specified RDOS file does not exist
ERR: /F RENAM	Error renaming RDOS file
FPREP: .START/A	No .START statement in input <u>file</u>
FPREP: DATA LINE/.END	Invalid data line entry or no .END
FPREP: FIELD LENGTH EXCE	Field width exceeded or data field missing
ADD: DUPLICATION	Call letters duplicated in input (recheck data)
DEL: .START/D	No .START statement in input file
AFOS: KSRCF	Specified product key not in database
AFOS: RDBKF	Can't get first AFOS data block
AFOS: SOF/56	No .START statement in input <u>product</u>
AFOS: EOF203	No AFOS end-of-text
AFOS: NXBKF	Subsequent AFOS data blocks unavailable

APPENDIX C

NSEEK Error Messages

Errors in program execution generally occur due to missing and/or invalid AFOS product keys or products unavailable in the AFOS database. System errors may be due to software or hardware failure and will begin with the acronym "SYS". AFOS product and file errors produce the following operator messages:

ERR: CCCNNXXX	Invalid CCCNNXXX in command line
ERR: VERSION	Invalid version entry in command line
ERR: VERSION >1	Invalid version number
ERR: VER #	Requested version unavailable
ERR: SOF -56	Period (56K) not found in first data block of product
ERR: KSRCF -1/0	AFOS product key not in local database
ERR: RDBKF -1/0	Can't get first AFOS data block
ERR: NXBKF -1/0	Subsequent AFOS data blocks unavailable
ERR: EOF -203	AFOS end-of-text missing in last data block
TONUM: CHR	Illegal character in file's byte/block pointers
TOOCT: CHR	Illegal character in file's byte/block pointers
STAT: SHEFOUT	Attempt to read a non-existent SHEFOUT file
STAT: SHEFIN	Attempt to decode a non-existent SHEFIN file
STAT: SHEFPARM	SHEF parameter file not on disk
SYS: SWAP/DCODR	WXRPARSHEF.SV not on disk; swap fails

APPENDIX D

WXR Error Messages

Extensive error checking facilities have been incorporated into WXR. Process errors are flagged to the operator via the output console. Fatal errors will terminate execution. WXR has been programmed to respond with a very short message when an error has occurred, whether it be fatal or non-fatal. The message form for fatal errors is

ROUTINE:PROBLEM (or) SYS:ROUTINE-PROBLEM

where "ROUTINE" is the module name detecting the error and "PROBLEM" is a brief message about the error condition. System errors begin with the acronym "SYS" followed by a one or two word code. System errors generally are fatal in nature and may be due to hardware or software failure in the system. Non-fatal errors will pass a warning message to the output console and execution will continue. Non-fatal error messages has the form of

WARN:MESSAGE

where "MESSAGE" is a deliberately brief message describing the error encountered. All errors and warnings should be addressed immediately.

Command Line Errors

WXR checks for consistency in the command line switches and arguments.

RDCLI: YOU CANT DO THAT	Only one global data filter switch is permitted
RDCLI:COM	Unknown local switch entered
RDCLI: /C	Unknown character in local /C argument
RDCLI: /F NO WAY	Too many format parameter files specified
RDCLI: /F	No format parameter file specified in command line

Processing Errors/Operator Warnings

These messages may result from errors in file input/output, character truncations, invalid or unknown format parameter file specification or AFOS database storage problems.

EXCTV: E/TERM	Abnormal WXR termination/end of parameter file?
PIL: FRE	Problem reading format parameter file
PIL: CCCNNNXXX	Illegal CCCNNNXXX in format par. file
PIL: ADDRESSEE	Illegal addressee in format par. file
WRCOM: FRE	Problem reading format parameter file
WRDAT: FRE	Problem reading format parameter file
WRCOM: OVRFLO	Line length exceeded in .COMMENT DATE
FORMT: FRE	Problem reading format parameter file
FORMT: /C-/B	/C or /B in format line but not in global area
FORMT: FIELD/LINE	Field/total line length greater than 72
FORMT: NM	First format parameter is not "NM"
FORMT: JIBBER	Unknown character in field width
LDSHF: SHEF INPUT	No match of SHEF data with NET.DT stations
NET: OUT/BOUNDS	Program pointer to invalid record in NET.DT
WRALP: FRE/NET.DT	Problem reading station name in NET.DT
WRALP: REOCT	Invalid conversion character in data value
LTRS: FRE	File read error

Database/File Maintenance/AFOS Storage Error

These are problems arising from faulty file to non-existent files or empty files. A file status check is first performed for NET.DT (database file) and SHEFOUT (decoded SHEF data binary file) before WXR executes. WXR cannot execute properly without these two important files. The AFOS storage module will perform 4 15-second wait periods if WXR senses AFOS is unavailable. WXR will either terminate (processing one format parameter file) or continue onto the next

format parameter file (multiple format parameter files) after four attempts (60+ seconds) to store the dummy file WXR.99 into the AFOS database.

MAOPN: SHEFOUT	Decoded SHEF file SHEFOUT missing
MAOPN: C/FILE	Cannot create file WXR.99
MAOPN: NET.DT	Database file NET.DT missing
MAOPN: FILE	Runtime specified format file (WXR.*) missing
LDNET: MT/NET.DT	Existing NET.DT file empty or unreadable
LDNET: FRE/NET.DT	Problem reading NET.DT file, execution continues
TAFOS: WAIT	AFOS storage problem...will try 4 times
WARN: VAL>FW	Data value larger than allotted field width
WARN: STN ID	Station ID longer than 8 characters
WARN: 90%	Runtime data stacks approaching capacity
WARN: 100% FULL	Runtime data stacks at full capacity
WARN: P/PIX	Incorrect lat/lon conversion to pixels
WARN: P/ID	Problem writing station call letters to plotfile
WARN: PB/PIX	Incorrect lat/lon conversion in BG mode
WARN: PB/ID	Problem writing station call letter to plotfile in BG mode
CART: LON	Longitude out of bounds/no conversion to pixels
CART: LAT	Latitude out of bounds/no conversion to pixels
CART: MAP	Incorrect map label

APPENDIX E

Changing the PLT Default Parameters

WXR may be tailored to suit local plot output needs. This requires using OEDIT.SV, the RDOS octal editor, to change any or all of six areas in WXR.SV. This is not difficult, but should be left to a person who is experienced in using the octal editor. Caution should be exercised throughout the whole procedure. Make your changes to a copy of the original file.

1. Changing the output PLT product key and the addressee

Locations 626 through 633 contain the AFOS keyname for the plotfile and the addressee in the form of "CCCNXXAAAA". The original defaults are "CRWPLTCIN000". If these locations are changed you must also change locations 640 to 644 to the new AFOS keyname only (no addressee here). The location 644 must contain the last letter of the keyname in the left byte and an ASCII zero in the right byte.

2. Changing the plotted SHEF elements

Locations 566-577 determine which SHEF elements are to be included in the plotfile and their order in the PLT entries (this in turn affects how they are plotted around the station model by PMOD). A maximum of 10 elements can be entered. The original defaults are

LOCATION	SHEF ELEMENT
566	PP
567	HG
570	SD
571	SW
572	TA
573	Ø
574	Ø (for expansion)
575	Ø
576	Ø
577	Ø

Elements currently in these locations can be deleted by changing the location to Ø. For example, SD and SW can be omitted by changing locations 570 and 571 to Ø. PP, HG and TA will still be plotted in their same locations.

The (octal) number of SHEF elements to be plotted is in location 615. For example, currently there are five SHEF elements listed in locations 566-577, so location 615 should hold the number five (octal).

3. Changing the data value type (real or integer)

One must decide whether the data value output for each SHEF element entered in locations 566-577 is a real or integer number. Each SHEF element has a corresponding data type flag which appears in locations 600-611. Enter a "0" for a real value or a "1" for an integer. The current data flags (corresponding to the current SHEF elements given in 2. above are:

LOCATION	DATA TYPE
600	0 (real)
601	0 (real)
602	1 (integer)
603	1 (integer)
604	1 (integer)
605	0 (real)
606	0 (real)
607	0 (real)
610	0 (real)
611	0 (real)

4. Changing the product storage flag

The WXR plotfile is initially written to the dummy disk file WXR.99. A flag in location 612 indicates what WXR should do with the dummy file.

VALUE	RESULT
-1	Do nothing. Plotfile output is left in file WXR.99.
0	Rename the file to CCCNNNXXX (the AFOS keyname as specified in section 1 of this Appendix)
1	Store the file into the AFOS database as CCCNNNXXX (the AFOS keyname as specified in section 1 of this Appendix)

5. Changing the station call letter output

This value (location 613) determines what WXR will do with the station call letter in the plotfile.

VALUE	RESULT
-1	WXR determines how many call letters there are per station ID and writes those letters (no padding) to the

plotfile (maximum 8, minimum 1)

- 0 WXR will not write any call letters to the plotfile; this field will be skipped
- 1 WXR writes a specified number of call letters. The octal number must appear in location 614. Maximum is still 8 and minimum is 1.

6. Changing the plot symbol and symbol type

A value in location 616 determines whether WXR will write a station plot symbol to the plotfile. Subsequently the value in location 617 determines the type of plot symbol to be used.

VALUE	RESULT
1	WXR will write the value in location 617 to the plotfile (location 617 currently holds the value for a cloudy symbol - 70 octal).
0	WXR will skip the station plot symbol field and continue onto the next field. No station symbol will be plotted.

APPENIDX F

WXR PLT Background Option

WXR can produce a separate station name/station symbol plotfile that can be used for constructing a station overlay map. The operator can overlay this station call letter and/or station symbol plot on another graphic containing the data values in order to identify and locate stations. This is useful in reducing CLS memory overflow problems that occur when graphics become larger than the allowable CLS memory partition.

The PLT options for station call letters and station symbols are applicable to the background option (see Appendix E, options 5 and 6 for specific instructions on these two options). A station call letter background does not necessarily need station symbols and vice versa.

The first step is to produce a plotfile with the following WXR command:

```
RUN:WXR/P/L/B
```

this command line will produce a plotfile and store it into the AFOS database (under the CCCNNNXXX in locations 626-633, see Appendix E). The plotfile will contain station call letters and/or station plot symbols according to the options previously selected.

The second step is to produce the overlay graphic using the PMOD plotting software (see I C, References) with an appropriate plot model offset file. It is important to remember to store the graphic in the AFOS database under a separate AFOS key. Do not store the station call letter/station symbol map under the identical AFOS key for regularly plotted data; this will defeat the purpose of an overlay graphic. It is only necessary to produce the overlay map once, when a station network has been established (unless changes are needed later). Figure F-1 shows such an overlay map.

The last step is to modify WXR.SV to eliminate station call letters and/or station symbols from routinely produced plotfiles (again see Appendix E, options 5 and 6). There is no reason to include these fields if an overlay graphic with them already exists. However, it is not mandatory that they be removed.

APPENDIX G

WXR Special Data Value/Special Character Options

This Appendix addresses locations in WXR.SV that contain special values and characters for certain aspects of data handling and processing. WXR is initially loaded with default values. If necessary, these special values and characters may be changed to suit local needs. Changes to WXR.SV must be done with the octal editor OEDIT.

1. Sector spacing

Location 3037 contains the octal number of spaces between sectors used when columnizing weather wire products (local /C). Default is 2.

2. Z-time

Location 3040 holds the local Zulu hour (octal) for data collection (used with global /C or no global switches). The default is 12Z.

3. Local Z-time window

Locations 3041 and 3042 contain the positive/negative time windows in (octal) hours for the Z time given in location 3040. These are used only when the global /C is used or if no global switches are used. The time window allows a flexible range (may differ in either direction) for data filtering. The default values are 1 and 1, that is, +/- 1 hour from the default Z time (if default Z-time of 12Z is used, this window runs from 11Z to 13Z).

4. Tendency characters

Locations 3043 and 3044 contain the positive and negative time-change computation characters (octal ASCII equivalents). They are only used when WXR computes and time change value (such as a 24 hour change computation). The default characters are a plus and a minus sign.

5. Local time

Location 3110 contains the (octal) value it takes to translate

Zulu time to local 24-hour clock time. For Eastern Standard Time, the value is 5; for Eastern Daylight Time, the value is 4.

<u>TIME ZONE</u>	<u>VALUE</u>
ATLANTIC STANDARD, AST	4
ATLANTIC DAYLIGHT, ADT	3
EASTERN STANDARD, EST	5
EASTERN DAYLIGHT, EDT	4
CENTRAL STANDARD, CST	6
CENTRAL DAYLIGHT, CDT	5
MOUNTAIN STANDARD, MST	7
MOUNTAIN DAYLIGHT, MDT	6
PACIFIC STANDARD, PST	8
PACIFIC DAYLIGHT, PDT	7

WXR -- WEATHER WIRE REFORMATTING PACKAGE (VER 2.5)
(NET)

PART A: PROGRAM INFORMATION AND INSTALLATION PROCEDURE

PROGRAM NAME: NET

AAL ID: XXXXXXXX
REVISION NO.: 2.50

PURPOSE: NET is a dictionary/database utility program that will assemble and alphabetize station information into an RDOS disk file, NET.DT. The information contained in the NET.DT file is primarily utilized by WXR, but may have other uses.

PROGRAM INFORMATION

DEVELOPMENT PROGRAMMER(S):	MAINTENANCE PROGRAMMER(S):
H. OPITZ	H. OPITZ
LOCATION: RFC CIN	LOCATION: RFC CIN
PHONE: FTS 684-2371	PHONE: FTS 684-2371
LANGUAGE: DG FORTRAN IV/5.20	TYPE: STANDARD (CHAINED TO BY WXR)
DATE: 03/08/84	REVISION DATE: 08/06/84

RUNNING TIME: VARIES WITH NUMBER OF STATIONS PROCESSED

DISK SPACE:	PROGRAM FILES	40 RDOS BLOCKS (NET ONLY)
	DATA FILES	VARIABLE

PROGRAM REQUIREMENTS

PROGRAM FILES:

<u>NAME</u>	<u>COMMENTS</u>
NET.SV	
WXR.SV,.OL	NET is run via WXR

DATA FILES:

<u>NAME</u>	<u>DP LOCAT.</u>	<u>READ/WRITE</u>	<u>COMMENTS</u>
NET.DT	DP0*	WRITE	Output file of station information
NET.BU	DP0*	WRITE	NET.DT backup file, deleted by default

NET.SC	DPØ	WRITE/READ	Scratch file, deleted by default
NET.TM	DPØ	WRITE/READ	Scratch file for input product or file, deleted by default

* These files may be placed in other directories for normal use, but when NET is being used to add/delete stations, make sure they are on DPØ.

AFOS PRODUCTS:

<u>ID</u>	<u>ACTION</u>	<u>COMMENTS</u>
NONE		

LOAD LINE

RLDR/P NET.SV/S NET<MAIN IH 1 2 3 4 5 6> <UTIL BG FORT>.LB 25/C

PROGRAM INSTALLATION

1- Put WXR.SV and .OL and NET.SV on DPØF, link from DPØ.

WXR -- WEATHER WIRE REFORMATTING PACKAGE (VER 2.50)
(NSEEK)

PART A: PROGRAM INFORMATION AND INSTALLATION PROCEDURE

PROGRAM NAME: NSEEK

AAL ID: XXXXXXXX

REVISION NO.: 2.50

PURPOSE: NSEEK is a multi-purpose utility program that will assemble an RDOS disk file containing any specified AFOS SHEF product(s) and invoke the SHEF decoder package. NSEEK also includes a debug routine that will redo the SHEF binary output file (SHEFOUT) for display on an AFOS console. This debug routine allows the operator to examine the actual contents of SHEFOUT.

PROGRAM INFORMATION

DEVELOPMENT PROGRAMMER(S):

H. OPITZ

LOCATION: RFC CIN

PHONE: FTS 684-2371

LANGUAGE: DG FORTRAN IV/5.20

DATE: 03/08/84

MAINTENANCE PROGRAMMER(S):

H. OPITZ

LOCATION: RFC CIN

PHONE: FTS 684-2371

TYPE: CHAIN

REVISION DATE: 08/06/84

RUNNING TIME: VARIES WITH NUMBER AND SIZE OF PRODUCTS ASSEMBLED

DISK SPACE: PROGRAM FILES

108 RDOS BLOCKS (NSEEK, WXR-
PARSHEF)

DATA FILES

35 RDOS BLOCKS (SHEFPARM)

PROGRAM REQUIREMENTS

PROGRAM FILES:

NAME

NSEEK.SV

WXRPARSHEF.SV

WXR.SV, .OL

COMMENTS

The SHEF decoder adapted for WXR.
NSEEK may be run through WXR.

DATA FILES:

<u>NAME</u>	<u>DP LOCAT.</u>	<u>READ/WRITE</u>	<u>COMMENTS</u>
SHEFIN	DPØ	WRITE/READ	The assembled AFOS products; input for SHEF decoder.
SHEFOUT	DPØ	WRITE	Decoded SHEF data; binary file.
SHEFPARM	DPØF	READ	SHEF parameter file
SHEF.ER	DPØ	WRITE	SHEF error file
OUT.DT	DPØ	WRITE	An ADM-displayable version of SHEFOUT; allows operator to easily check data.

AFOS PRODUCTS:

<u>ID</u>	<u>ACTION</u>	<u>COMMENTS</u>
VARIOUS	READ	Specified in run line.

LOAD LINE

RLDR/P NSEEK NSEEKCOM NSEEK<1 2 3 4 5> <UTIL BG FORT>.LB 25/C

PROGRAM INSTALLATION

- 1- Put NSEEK.SV, SHEFPARM and WXRPARSHEF.SV on DPØF linked to DPØ. WXR.SV and .OL should also exist on DPØF, linked to DPØ.
- 2- The output files may be linked to DPØF if there is more space there.

WXR -- WEATHER WIRE REFORMATTING PACKAGE (VER 2.50)
(WXR)

PART A: PROGRAM INFORMATION AND INSTALLATION PROCEDURE

PROGRAM NAME: WXR

AAL ID: XXXXXXXX
REVISION NO.: 2.50

PURPOSE: WXR is a software package that manipulates decoded SHEF data according to a specified output format and command line runtime options to create products for the weather wire and plotfiles for PMOD plotting. WXR uses various master format files (assembled during the initial setup) that contain the actual output format for specific weather wire products. Multiple master files may be assembled for various output formats. The operator indicates which file(s) WXR will use via the command line.

PROGRAM INFORMATION

DEVELOPMENT PROGRAMMER(S):
H. OPITZ
LOCATION: RFC CIN
PHONE: FTS 684-2371
LANGUAGE: DG FORTRAN IV/5.20
DATE: 03/05/84

MAINTENANCE PROGRAMMER(S):
H. OPITZ
LOCATION: RFC CIN
PHONE: FTS 684-2371
TYPE: OVERLAY, CHAIN
REVISION DATE: 08/06/84

RUNNING TIME: VARIES WITH AMOUNT OF DATA, NUMBER OF PRODUCTS,
PLOTFILE OPTION

DISK SPACE: PROGRAM FILES
DATA FILES

120 RDOS BLOCKS (WXR.SV,.OL)
VARIES WITH NUMBER OF PARA-
METER FILES

PROGRAM REQUIREMENTS

PROGRAM FILES:

NAME
WXR.SV
WXR.OL

COMMENTS

DATA FILES:

<u>NAME</u>	<u>DP LOCAT.</u>	<u>READ/WRITE</u>	<u>COMMENTS</u>
NET.DT	DPØF	READ	Generated by NET; contains station information.
SHEFOUT	DPØ	READ	Decoded SHEF data from WXRPARSHEF.
WXR.*	DPØF	READ	Output format files; * is any single charac- ter.
WXR.TD	DPØ*	WRITE/READ	If global /C is used, current run's data will be stored in this file for later use.
WXR.YS	DPØ*	WRITE/READ	Also used with global /C - used for change com- putations.
WXR.99	DPØ	WRITE	Temporary output file, deleted after storage in AFOS data base.

* These two files should remain on DPØ (i. e., no links!). The other files may be linked to other directories.

AFOS PRODUCTS:

<u>ID</u>	<u>ACTION</u>	<u>COMMENTS</u>
SPECIFIED BY FORMAT FILE	STORE	A variety of products is possible.

LOAD LINE

```
RLDR/R/G/P WXR.SV/S WXR<MAIN COM IH> [EXCTV] [LTRS WRMSG,REOCT
C24HR,LLCVT CART,PILL,PLTHDR,MAOPN TAFOS UPDTE,WRCOM WRTIM]
[WRDAT,LDSHF LODYS,WXALPHA WRALPHA,WXPLT PLACE WXFIL,WXBKG WXBPL
LDNET,FORMT CNVRT2,RDCLI DATIM] <BG UTIL FORT>.LB
```

PROGRAM INSTALLATION

1- WXR.SV and .OL should exist on DPØF with links to DPØ.

WXR -- WEATHER WIRE REFORMATTING PACKAGE (VER 2.50)
(NET)

PART B: PROGRAM EXECUTION AND ERROR CONDITIONS

PROGRAM NAME: NET

AAL ID: XXXXXXXX
REVISION NO.: 2.50

PROGRAM EXECUTION

1. The station information NET will add to or delete from NET.DT must exist either in an AFOS product created with message comp or an RDOS file created with a text editor (filename may be up to 18 characters long). Up to 500 station entries are allowed. Information on the formats required appears in ERCP #24, Section III A 2.

2. After the information has been stored in the product or file, run NET.

RUN:WXR/N/A/D/S/X ccnnnxxx (or) filename/F

where ccnnnxxx or filename contain the raw station data and the global switches imply:

- /N Utilize NET.SV functions (mandatory)
- /A Add the input station information to NET.DT. If no NET.DT exists, create one.
- /D Delete the input stations from NET.DT.
- /S Do not delete NET.BU (use with /A or /D)
- /X Save all temporary, scratch and backup files (used with /A or /D). This is useful in determining any errors encountered during execution.

(NOTE: /A and /D cannot be used in the same run.)

ERROR CONDITIONS

ADM MESSAGES
NONE

MEANING

DASHER MESSAGES	MEANING
ERR: YOU CANT DO THAT	You can't add and delete simultaneously
ERR: FILNAM/CCCNNNXXX	No input filename or product key given in command line
ERR: CCCNNNXXX	Invalid AFOS product key
ERR: /F	Invalid RDOS filename
ERR: /F STAT	Specified RDOS file does not exist
ERR: /F RENAM	Error renaming RDOS file
FPREP: .START/A	No .START statement in input <u>file</u>
FPREP: DATA LINE/.END	Invalid data line entry or no .END
FPREP: FIELD LENGTH EXCE	Field width exceeded or data field missing
ADD: DUPLICATION	Call letters duplicated in input (recheck data)
DEL: .START/D	No .START statement in input file
AFOS: KSRCF	Specified product key not in database
AFOS: RDBKF	Can't get first AFOS data block
AFOS: SOF/56	No .START statement in input <u>product</u>
AFOS: EOF203	No AFOS end-of-text
AFOS: NXBKF	Subsequent AFOS data blocks unavailable

WXR -- WEATHER WIRE REFORMATTING PACKAGE (VER 2.50)
(NSEEK)

PART B: PROGRAM EXECUTION AND ERROR CONDITIONS

PROGRAM NAME: NSEEK

AAL ID: XXXXXXXX

REVISION NO.: 2.50

PROGRAM EXECUTION

1. NSEEK is run using the following command line:

RUN:WXR/R/A/D/O cccnnxxx */V

cccnnxxx is the AFOS SHEF product to be assembled and/or decoded and the global switches have the following functions:

- | | |
|----|---|
| /R | Use NSEEK functions (mandatory) |
| /A | Append the current SHEFIN file to the AFOS product in the run line. If /A is not used, the current file is deleted and a new one containing the product is created. |
| /D | Invoke the SHEF decoder, which will act on the current SHEFIN file. |
| /O | Debug the current SHEFOUT file for display on an ADM (produces file OUT.DT). |

If /A is used, cccnnxxx must appear in the command line. The local /V switch can be used to extract previous versions of cccnnxxx: 1/V for the first previous version, 2/V for the next, and so on up to 9 versions.

ERROR CONDITIONS

ADM MESSAGES	MEANING
NONE	
DASHER MESSAGES	MEANING
ERR: CCCNNXXX	Invalid CCCNNXXX in command line

ERR: VERSION	Invalid version entry in command line
ERR: VERSION >1	Invalid version number
ERR: VER #	Requested version unavailable
ERR: SOF -56	Period (56K) not found in first data block of product
ERR: KSRCF -1/0	AFOS product key not in local database
ERR: RDBKF -1/0	Can't get first AFOS data block
ERR: NXBKF -1/0	Subsequent AFOS data blocks unavailable
ERR: EOF -203	AFOS end-of-text missing in last data block
TONUM: CHR	Illegal character in file's byte/block pointers
TOOCT: CHR	Illegal character in file's byte/block pointers
STAT: SHEFOUT	Attempt to read a non-existent SHEFOUT file
STAT: SHEFIN	Attempt to decode a non-existent SHEFIN file
STAT: SHEFPARM	SHEF parameter file not on disk
SYS: SWAP/DCODR	WXRPARSHEF.SV not on disk; swap fails

WXR -- WEATHER WIRE REFORMATTING PACKAGE (VER 2.50)
(WXR)

PART B: PROGRAM EXECUTION AND ERROR CONDITIONS

PROGRAM NAME: WXR

AAL ID: XXXXXXXX
REVISION NO.: 2.50

PROGRAM EXECUTION

1. WXR will create weather wire products and plotfiles: 1) using data assembled and decoded in the file SHEFOUT via NSEEK, 2) for the stations in the NET.DT file created with NET, 3) according to specified parameter format files. Directions for creating these files appear in ERCP #24, Section III C 2. Be sure these steps have been taken.
2. WXR may then be run using the following command line:

RUN:WXR/C/O/A/P/L/B * /F X/C

where the global switches have the following functions:

- /C Load 12Z type R data for current date and/or do change computation.
- /O Load any hour's type R data for the current date. No change computation or file update.
- /A Load any type data for any hour and/or day. Operator becomes data filter.
- /P After producing weather wire products, create a plotfile. Default output is CRWPLTCIN, may be changed with octal editor (see ERCP #24, Appendix E)
- /L Produce plotfile only (used with /P, e. g. /P/L)
- /B Produce a special background plot (used with /P/L, e. g. /P/L/B. See ERCP #24, Appendix F)

(NOTE: /C, /O and /A are independent and cannot be used in the same run.)

The local /F indicates the parameter format files to be used. The single-character extension(s) of the file(s) is replaces the asterisk. Up to three files can be specified, for example if one wanted to use the format files WXR.1, WXR.3, and WXR.D, 13D/F should appear in the command line. This switch is mandatory.

The /C is optional and indicates how many columns should appear in the weather wire products.

ERROR CONDITIONS

ADM MESSAGES NONE	MEANING
DASHER MESSAGES RDCLI: YOU CANT DO THAT	MEANING Only one global data filter switch (/A/O/C) allowed
RDCLI: /F	No format parameter file specified in run line
FORMT: /C-/B	/C or /B in format line but no global /C in run line
MAOPN: SHEFOUT	Decoded SHEF file SHEFOUT missing
MAOPN: NET.DT	Database file NET.DT missing
MAOPN: FILE	Format file missing (WXR.*)
TAFOS: WAIT	AFOS storage problem - will try four times

Other WXR error messages appear in APPENDIX D of ERCP #24.

Eastern Region Computer Programs and Problems (Continued)

- 19 Verification of Asynchronous Transmissions. Lawrence Cedrone, March 1984. (PB84 189885)
- 20 AFOS Hurricane Plotter. Charles Little, May 1984. (PB84 199629)
- 21 WARN - A Warning Formatter. Gerald G. Rigdon, June 1984. (PB84 204551)
- 22 Plotting TDL Coastal Wind Forecasts, Paula Severe, June 1984 (Revised) (PB84-220789)
- 23 Severe Weather Statistics STADTS Decoder (SWX) and Plotter (SWY), Hugh M. Stone, June 1984. (PB84-213693)

NOAA SCIENTIFIC AND TECHNICAL PUBLICATIONS

The National Oceanic and Atmospheric Administration was established as part of the Department of Commerce on October 3, 1970. The mission responsibilities of NOAA are to assess the socioeconomic impact of natural and technological changes in the environment and to monitor and predict the state of the oceans and their living resources, the atmosphere, and the space environment of the Earth.

The major components of NOAA regularly produce various types of scientific and technical publications in the following kinds of publications:

PROFESSIONAL PAPERS — Important definitive research results, major techniques, and special investigations.

CONTRACT AND GRANT REPORTS — Reports prepared by contractors or grantees under NOAA sponsorship.

ATLAS — Presentation of analyzed data generally in the form of maps showing distribution of rainfall, chemical and physical conditions of oceans and atmosphere, distribution of fishes and marine mammals, ionospheric conditions, etc.

TECHNICAL SERVICE PUBLICATIONS — Reports containing data, observations, instructions, etc. A partial listing includes data serials; prediction and outlook periodicals; technical manuals, training papers, planning reports, and information serials; and miscellaneous technical publications.

TECHNICAL REPORTS — Journal quality with extensive details, mathematical developments, or data listings.

TECHNICAL MEMORANDUMS — Reports of preliminary, partial, or negative research or technology results, interim instructions, and the like.



Information on availability of NOAA publications can be obtained from:

**ENVIRONMENTAL SCIENCE INFORMATION CENTER (D822)
ENVIRONMENTAL DATA AND INFORMATION SERVICE
NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION
U.S. DEPARTMENT OF COMMERCE**

**6009 Executive Boulevard
Rockville, MD 20852**

