U.S. DEPARTMENT OF COMMERCE
NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION
NATIONAL WEATHER SERVICE
OFFICE OF SYSTEMS DEVELOPMENT
TECHNIQUES DEVELOPMENT LABORATORY

TDL OFFICE NOTE 00-1

MOS-2000

Harry R. Glahn and J. Paul Dallavalle

January 2000

TABLE OF CONTENTS

MOS-2000

Harry R. Glahn and J. Paul Dallavalle

1. INTRODUCTION

Model Output Statistics (MOS) is a technique in which a predictand is related statistically to predictors, most of which are themselves forecasts (or output) from numerical models. This technique's history reaches back to the late 1960's[1], and the forecasts produced from the Technique Development Laboratory's (TDL's) MOS system provide substantial guidance to forecasters in the National Weather Service (NWS) and other organizations (Carter et al. 1989). The last document describing the MOS system in detail is TDL Office Note 74-14, the Introduction of which is included as an appendix to this document, TDL Office Note 00-1, to furnish the reader some continuity with history. This office note was used as a loose-leaf notebook and updated through the years.

The architecture of the MOS-2000 system is basically the same as the previous MOS systems developed and used within TDL. That is, the processes that need to be carried out in the development of a statistical interpretive system are fairly constant. However, the details of the design and the software itself are completely new. Users of the system were solicited for ideas and requirements, and MOS-2000 fulfills most of those requirements. However, in some cases, practicality and ease of use of the system had to prevail over complexity. A primary goal was to develop a system and manage it in such a way that interpretive equations could be updated seasonally (e.g., a summer's worth of data could be added to the existing sample to produce an updated operational system to be used during the next summer).

The MOS system consists of extensive data archives from numerical models and of observations, a library of routines and subroutines, and rather complete documentation of all portions of the system, this office note being a fundamental part. The software is written so that it can function on either a 32-bit word workstation (e.g., Hewlett Packard[2] 755), or the 64-bit word CRAY. This has added some complexity, but was felt desirable because it is uncertain which portions of the MOS-2000 system will be most efficiently and effectively performed on the CRAY as opposed to the workstation; likely, this split will change with time as workstation capabilities continue to increase.[3]

---

[1]The first such product was disseminated to field forecasters in 1968 (Weather Bureau 1968).

[2]No endorsement of specific equipment or companies is made or implied in this document.

[3]The MOS-2000 system has been in the planning and development stages for several years. During that time, the equipment available changed. As of this writing, the NWS CRAY mainframes are being replaced by IBM machines. The 64-bit capability has been retained, although it may not be needed in the future.

While the primary use of MOS-2000 will be for the MOS technique, perfect prog and classical procedures can be accommodated as well (see Glahn 1991 for a discussion of these three techniques). The constraints imposed by the system are basically formats and identification of data. Specific modules or programs can be written to perform functions other than those available at any point in time, and can be added to the system if it is desirable to do so; the design permits and even encourages this evolution.

This office note is a "living document" and will be updated as necessary. For instance, as the National Centers for Environmental Prediction (NCEP) operations and the archival processes are moved from the CRAY to the IBM, updates will be needed (e.g., Chapter 12). The header will establish the creation or update date.

2. SYSTEM DESIGN

The top level design of the system is described below. The last section describes each of the major programs. Each program operates on input data and usually supplies data for another program. When a specific program is mentioned in the sections below, the reader can refer to that program's documentation for a more complete explanation of its capabilities.

All major MOS-2000 programs have a "driver" in which PARAMETER statements are used to set variable dimensions that are passed to the main and other subroutines (e.g., U201 has a driver DRU201).

All programs have a variety of inputs, including a ".CN" file which contains parameters for running the program, variable lists (see Chapter 4), information about the variables (see Chapter 11), and station lists and a station directory (see Chapter 10). Also, large input data sets are usually required, and the programs may output large quantities of data; in the following discussion, we are referring to these large data sets.

Most data used within the MOS-2000 system are packed in a format devised specifically for the purpose, called TDLPACK (see Chapter 5). While the packing algorithm is basically the same for gridpoint and "point" or "vector" data, minor differences exist because the information needed with each packed record is somewhat different for the two types. In particular, the point data have to have a "directory" that specifies for what points the data are valid within the record. The data other than gridpoint are generally called vector data in this document. Most large data sets are sequential, including all gridpoint data. ASCII data are used in certain cases where it may be necessary to easily view or edit them.

The large data sets fall into one of the following four categories:

1. Sequential gridpoint data files in TDLPACK format.
2. Sequential vector data files in TDLPACK format.
3. Random access files in TDLPACK format.
4. ASCII data in a variety of formats.

The basic steps in developing guidance with the MOS-2000 System are described below; however, there are other supporting steps and programs that are not described here.

A. Archive Data

Basic forecast fields from various numerical models are saved in gridpoint form on TDL grids defined in Chapter 12 in the TDLPACK format described in Chapter 5. Current archiving programs run on the NCEP mainframe. These gridpoint data furnish the majority of predictors for MOS-2000. Usually, interpolation into the field is done (see Section B below), but the gridpoint values themselves can be used as predictors (or even predictands in the case of analyses, rather than forecasts) in a grid-oriented development.

Observations of various kinds and sources are also archived as described in Chapter 12. For instance, hourly observations (METARs) are taken from NCEP files and put into TDLPACK format. Satellite data are archived to augment the cloud amounts in the METAR observations.

B. Preparation of Predictor Data

Generally, interpolation into the model fields is done to get values at "stations" or other places where observations or other data that can be used as predictand values are available. Many times it is desired to perform computations on these data either before or after interpolation. The interpolation and computation are done by program U201. Both input and output, other than control files and ASCII print, are in the TDLPACK format described in Chapter 5.

Interpolation can be either bilinear or biquadratic, or can be done with a special algorithm designed for discontinuous fields (e.g., precipitation amount). Computations are usually performed by calling a specific computational routine through subroutine OPTION; however, smoothing or making point binary variables is done directly without calling OPTION. U201 contains a "lookback" feature which allows data from a previous time (cycle) to be mixed with data from the cycle being processed (e.g., when processing numerical model output produced by the 1200 UTC run, output from the same, or a different, model can be processed from one or more previous cycles). Data that are less time related than model output (e.g., monthly relative frequencies) can be accessed through OPTION from a TDLPACK random access file (see Chapter 7).

C. Preparation of Predictand Data

Predictand data can originate from several sources such as hourly observations, severe thunderstorm data, and lightning location data. These data may exist at station locations (vector data) or on a grid (gridpoint data). U201 can process input vector data as well as gridpoint data to produce predictand data at the locations desired.

D. Production of Equations

Many techniques can be employed to determine relationships between a predictand and predictors. The workhorse of the MOS-2000 system is the regression program U600 or an alternate U602. Predictor input is in the TDLPACK vector format described in Chapters 5 and 6 as output by U201. Predictand input is also in the TDLPACK vector format. The primary output is composed of linear regression equations in ASCII format.

E. Production of Forecasts

Forecasts can be produced from the regression equations output from U600. The input is in the same TDLPACK format as input to U600. The output is also in TDLPACK format. U700 processes historical sequential data and U900 processes real time random access data. U700 is engineered to make forecasts for a small number of variables for many cycles and to write them to a sequential file; U900 is used for making many forecasts for one cycle and to write them to an external MOS-2000 random access file (see Chapter 7).

F. Transformation of Forecasts

Many of the forecasts produced by the regression equations are processed before being disseminated to field forecasters or before verification (e,g., categorical forecasts are made from probabilities or units are changed). Program U710 transforms vector forecasts, both input and output being TDLPACK sequential data files; random access vector files are also used for constant data as input. Program U910 performs the same function, but both input and output are external MOS-2000 random access files; output can also be put on a sequential vector file.

G. Verification of Forecasts

The forecasts produced from the regression equations, or in other ways, can be verified and compared with official forecasts collected through the National Verification Program or from NCEP. To do so, the forecasts must be collated with the relevant observations. All data input to and output from the verification program U850 will be in the TDLPACK format; the output statistics are ASCII.

H. Major Programs Within the System

Considerable effort is expended in collecting the various data sets and getting them identified and put into the TDLPACK format. These programs will be documented and discussed in detail as they are developed.

Some of the major programs that deal with the various data sources, once collected and archived, are discussed above and are summarized below. Most programs accept multiple files of a particular type. For instance, U201 will accept several gridpoint TDLPACK files, several vector TDLPACK files, and more than one random access TDLPACK vector files. However, only one output file of a particular type is provided for. For instance, U201 will write to a single sequential TDLPACK vector file. No sequential file is used for both reading and writing, except to skip over records already written. A random access file can usually be used for both reading and writing.

U170 computes "constants" for use in such programs as U600 and U700, such as means and relative frequencies; these are put into random access files with U351 or U352.

INPUT: Sequential TDLPACK vector.
OUTPUT: Sequential TDLPACK vector.
ASCII.

U520 quality controls and archives hourly observations.

INPUT: ASCII.
OUTPUT: Sequential TDLPACK vector.

U350 creates one or more random access files.

OUTPUT: Random access file, sans data.

U351 puts ASCII data into random access TDLPACK vector file.

INPUT: ASCII.
OUTPUT: Random access TDLPACK vector.

U352 puts data into random access TDLPACK vector file.

INPUT: Sequential TDLPACK vector.
OUTPUT: Random access TDLPACK vector.

U201 performs the following functions:

- Accepts basic gridpoint data from various sources, computes variables, and interpolates basic and computed fields to station (or other) locations.
- Accepts "constant" vector data (e.g., at station locations) stored in the MOS-2000 external random access file system, performs calculations as needed, and collates them with the interpolated values.
- Accepts TDLPACK vector format data and performs computations as needed, combining data from the various sources.
- Packs all output data in TDLPACK vector format.

INPUT: Sequential TDLPACK gridpoint.
Sequential TDLPACK vector.
Random access TDLPACK vector.
OUTPUT: Sequential TDLPACK vector.

U600 performs screening regression analysis and provides the equations that are used to make the guidance forecasts. (See Chapter 15 for the output equation format.)

INPUT: Sequential TDLPACK vector.
Random access TDLPACK vector.
OUTPUT: ASCII Equations.

U602 performs basically the same functions as U600, but provides for a slightly different screening algorithm and a "simplified" set of inputs. (See Chapter 15 for the output equation format.)

INPUT: Sequential TDLPACK vector.
Random access TDLPACK vector.
OUTPUT: ASCII Equations.

U700 makes forecasts using the equations produced by U600 or U602, station data produced by U201, and constant data, and outputs the forecasts in TDLPACK format. Equation input is in the format produced by U600.

INPUT: Sequential TDLPACK vector.
Random access TDLPACK vector.
ASCII Equations.
OUTPUT: Sequential TDLPACK vector.

U710 transforms variables, usually forecasts, into forms needed for guidance to forecasters or for verification. Input and output are sequential files.

INPUT: Sequential TDLPACK vector.
Random access TDLPACK vector.
OUTPUT: Sequential TDLPACK vector.

U660 accepts TDLPACK vector data, collates the data from all sources, and repacks them for the stations desired. It also writes for viewing and/or printing (a portion of) the data with a format provided as input.

INPUT: Sequential TDLPACK vector.
Random access TDLPACK vector.
OUTPUT: Sequential TDLPACK vector.

U830 accepts TDLPACK vector data and computes thresholds for making categorical forecasts. The ASCII output can be put into random access vector files with U351.

INPUT: Sequential TDLPACK vector.
OUTPUT: Random access TDLPACK vector.
ASCII.

U850 is used to collate various forecasts and observations and to comparatively verify the forecasts. Some transformation of data, like that done in U710, can also be done in U850 prior to verification.

INPUT: Sequential TDLPACK vector.
Random access TDLPACK vector.
OUTPUT: ASCII statistics.

U900 is the operational counterpart of U700; it uses the regression equations from U600, data prepared by U201, and constant data to produce the operational guidance forecasts. The primary output is a TDLPACK MOS-2000 random access file. Note that U201 is used as a preprocessor for U900 to produce sequential TDLPACK vector data.

INPUT: Sequential TDLPACK vector.
Random access TDLPACK vector.
ASCII Equations.
OUTPUT: Sequential TDLPACK vector.
Random access TDLPACK vector.

U910 is the operational counterpart of U710 for transforming forecasts. Input and output are MOS-2000 random access files.

INPUT: Random access TDLPACK vector.
OUTPUT: Random access TDLPACK vector.
Sequential TDLPACK vector.

It is expected that almost all calculations necessary before performing regression analysis with U600 or making forecasts with U900 will be done in

U201. A fully functional "OPTION" subroutine and option-called routines are provided, and the user can add other computational routines as necessary. U201 should be the only program, besides the archival and purely "file management" programs, that will have/need the capability to input gridpoint data; once U201 and archival and packing programs are run, all following programs should need only vector data. Development done for "gridpoints" will be treated in the same way as development for "stations;" the gridpoints will be given "identifiers" that are basically a combination of the grid position of the point. U600, U660, U700, U850, and U900 will make continuous or discrete point binaries as a basic capability. An option subroutine, OPTX, is provided for use; OPTX is used for calling computational routines and should be usable for all such "vector" programs.

It is planned that all MOS-2000 programs can be run on either 32-bit or 64-bit computers. TDLPACK data are such that either can be used efficiently. The archival of gridpoint data and the operational programs--those making operational forecasts--will likely be run on NCEP mainframes, which may be 32-bit or 64-bit. Much of the development will be done on 32-bit workstations.

3. DEVELOPMENT GUIDELINES

A. Introduction

It is desirable that there be as much conformity across the MOS-2000 system as practicable. There are many aspects to this, including reuse of modules wherever possible (if an existing one needs to be made more general, we can consider doing that); names of variables, both numerical and plain language; use of status (or error) numbers; format of diagnostic messages; use of input/output file names; use of units of variables; formats of data; documentation; etc.

These "standards" were established as the software was being written and are documented in this chapter. It is expected that as additional software is being written that the "conventions" used will either be those contained herein or that they will be discussed and agreed to before they become "locked in."

The reader will have to be somewhat familiar with other portions of this office note, and may even have to refer to specific software documentation contained in TDL Office Note 00-2, to fully understand the material in this chapter.

B. System-wide Guidelines

Programming Languages and Guidelines

FORTRAN 77 or FORTRAN 90 shall be used for all "noninteractive" software. The guidelines are in the document "Techniques Development Laboratory Software Development and Documentation Guidelines" dated June 1, 1992. Documentation for MOS-2000 has started (i.e., for U201, U600, and U660 and all associated subroutines) and is being modified somewhat from what is in the "Guidelines;" examples are available. The C standards for the "interactive" portion of the system are contained in the document "Techniques Development Laboratory C Software Implementation Conventions" prepared for use in the AWIPS Design, Development, and Testing Teams. All programmers are expected to follow these guidelines and standards for writing what will be essentially all new code. While the existing code for the current MOS system can be used as a guide and certain algorithms will be the same, the code itself and how it fits into the overall system will be new. The routines already completed can be used as templates.

Program Names

Names of main programs shall consist of a 3-digit number preceded by the letter "U" (e.g., U201) indicating they are for a UNIX environment. (Actually, these "main" programs may be subroutines with a driver (e.g., DRU201).

Subroutine names will generally have six or less characters and be such that they will not be confused with "main" programs (subroutines with drivers).

File Number Use

KFILDI/ / "System" Default Input--Reads control file 'XXXX.CN'. The actual number varies from system to system.
KFILDO/ / "System" Default Output (print) file. The actual number varies from system to system.
KFILDS/ / "System" Default Screen output. The actual number varies from system to system.
KFIL10/99/ Access to the disk portion of the MOS-2000 Internal Storage System (SCRATCH). Unit No. 99 is reserved for this purpose in MOS-2000 programs.
KFILIO/20/ U201 Interpolated Output. (Suggested only, user input control.)
KFILEQ/20/ U600 operational EQuation output (Suggested only, user input control.)
KFILD(1)/28/ U201 station list if different from KFIL29 and KFILDI. (Suggested only, user input control.)
KFILD(2)/29/ U201 station directory. (Suggested only, user input control.)
KFILCP/ / U201 Predictor Constants directory. (User input control.)
KFILP/ / Used for reading Predictor ID's. (User input control.)
KFILRA/45-49/ Unit Nos. 45 to 49 are reserved for unit numbers associated with constant data in the MOS-2000 External Random Access File System.
KFILT/97/ Unit No. 97 is reserved for use in linearizing routines for reading thresholds and associated information. It is also used by U830 for an internal file unit number.
KFILDT/ / Unit number for reading date list. (User input control.)
KFILIN/ / Unit number for data input. (User input control.)
IPX/ / Unit numbers, where X takes a value composed of one or two digits, for special output. For instance, it may be desired to keep the KFILDO file small for easy reading, but retain certain information in other files (e.g., a date list, a moderately-sized input data set, or voluminous regression equations). All diagnostic output (errors and possible errors) is to be written to the file attached to Unit No. KFILDO but may be also written to an IPX file if appropriate. For instance, a possible error associated with an input date list might be written to the same file that the date list itself is written to. The idea here is to be as sure as possible that the user sees the diagnostic. These IPX values are part of the input to a main program and are stored in an array IP( ). The numbers read can be zero, indicating no output on this file. The IPX numbers can be used to easily control what information is saved and where.

These IPX values can have considerable generality across programs, but their use will not be as universal as, for instance, the PARAMETER values discussed later. When these values are carried forward to subroutines, the subroutines, if they are to be general for several programs, may have defined only a variable IP.

A number of IPX values should have general applicability. Whenever information may be written to Unit No. KFILDO and also to Unit No. IPX, it is not written twice if KFILDO = IPX. The information associated with these values is:

IP1 - All errors and other information not specifically identified with other IPX numbers. When IP1 is non-zero, KFILDO will be set to this value. When IP1 is zero, KFILDO is used as the default in a DATA statement in the program.

IP2 - The input dates in IDATE( ). When there are errors, print will be to Unit NO. KFILDO and to Unit No. IP2.

IP3 - The output dates, after extension between input date pairs. When there are errors, output will be to Unit No. KFILDO and to Unit No. IP3.

IP4 - The station list call letters. If there are input errors, the station list will be written to Unit No. KFILDO as well as to Unit No. IP4.

IP5 - The station directory information. If there are errors, the directory will be written to Unit No. KFILDO and to Unit No. IP5.

IP6 - The predictors as they are being read in. This is good for checkout; for routine operation, IP7, IP8, and IP9 may be more appropriate.

IP7 - The predictor list in summary form. If there are errors, the predictor list will be written to Unit No. KFILDO and to Unit No. IP7. This list includes the parsed ID's in IDPARS( ).

IP8 - The predictor list in summary form after reordering. This list includes the parsed ID's in IDPARS( ).

IP9 - The predictor list in summary form after reordering. This differs from IP8 in that IP9 does not include the parsed ID's in IDPARS( ), but rather includes the information taken from the predictor constant file on unit KFILCP.

IP10 - The predictor ID's for the first day (actually, the first "cycle") as read from the input tape.

IP11 - The predictor ID's of the input fields actually needed.

IP15 - When the predictor list indicates interpolated values (values at stations) are to be written for viewing, they will be written to Unit No. IP15.

IP23 - Information about file openings and closings.

IP24 - Where the variables are to be found--directly from input, computed or reused from a previous input, or through the option subroutine OPTX.

Reading IDs and Associated Information

Each of the MOS programs that require a variable ID (the 4-word ID described in Chapter 4) shall use the same format for those IDs and shall be as consistent as reasonable in other ancillary information provided in the same record. The formats of four of the development programs are indicated in the table below.

Format for variable IDs and associated information for MOS-2000 programs.

In the table, a dash means that the variable is not used in the program indicated in the column heading. "Same" means the use is the same as that in the previous column. The variables NTYPVR and ITAU, as well as other information, are explained in the writeups of the programs in the column headings.

By far, the best way to construct records such as these is to start from an existing template for the same, or similar, program.

Printing ID's

The 4-word Identification in ID( ) is usually printed with the FORMAT (1XI9.9,1XI9.9,1XI9.9,1XI10.3). That is, leading zeros are printed, except that the threshold in word 4 is not printed when it is zero. The FORMAT (1XI9.9,2I10,I11) may also be used, and occasionally (1XI9,2I10,2I11). (Sometimes in diagnostics when leading zeros are not printed, it is not obvious it is the 4-word ID that is printed. On the other hand, in lists of variable IDs, not having leading zeros makes viewing easier. For a general rule, a single ID should have leading zeros; lists of IDs should probably not.)

Diagnostic Errors

Any time an error is detected, a diagnostic is "printed." This output is written to the default unit number KFILDO, unless the variable IP(1) is nonzero, in which case the output is written to unit number IP(1). When IP( ) variables indicate certain output to different files, the diagnostic is also written to that file when it is relevant to the error. This is to assure the user sees the diagnostic no matter which file is "looked at." The diagnostic is always written with the FORMAT(/' ****xxx...') where xxx... is the diagnostic to be printed. Use the "/" rather than FORTRAN control characters because this seems to be more universally accepted by output devices as a blank line. This format provides for a blank line, a blank character on the line so that an output device won't interpret it incorrectly, and a line starting with "****" with no space after the asterisks.

All routines called directly from OPTION should check the predictor IDs for safety, because users may be modifying OPTION.

Grid Storage

Two-dimensional grids are stored according to NCEP convention, which is starting at the lower left with element (1,1), proceeding "west to east" by increasing the first index. Successive rows "to the north" will be with the second index increasing. The "GRIB-like" TDLPACK format is used. See Chapters 5 and 6 for more information.

Vector Data Storage

The first record in each vector data set consists of NSTA 8-character call letters, preceded by 64 bits containing the number of additional bytes in the record. The number of stations, NSTA, is this "record size" divided by 8. The ASCII call letters are written from an integer array in binary form. The TDLPACK format is not used for this record, but the record structure is the same to the extent that the first 64 bits tells the number of bytes that need be read following these 64 bits. See Chapters 5 and 6 for more information.

Units

Generally, the standard WMO units will be used, except for some English units where the U.S. usage make that more reasonable. Latitude is in degrees North, and negative for South Latitude. Longitude is in degrees West; East longitude is represented by West longitude > 180 (e.g., 170^{o} E becomes 190).

Error Returns

Most subroutines have an "error" or "status" return. A zero always means a good return. Other values are listed below, together with the name of the subroutine producing it.

-1	OPTION--Option sets this value when the ID cannot be identified for switching, and means for U201 to not write the record.
1	PKBG--Packing would overflow IPACK( ).
2	PKBG--IPOS not in range 1 to 32.
3	PKBG--NBIT not in range 0 to 32.
4	PKBG--NBIT = 0, but NVALUE not = 0.
5	PKGPOO, PKGPPO, PKGPPS--Value won't pack in 30 bits.
6	UNPKBG--LOC not in range 1 to ND5
7	UNPKBG--IPOS not in range 1 to 32.
8	UNPKBG--NBIT not in range 0 to 32.
9	
10	UNPACK--Beginning of GRIB message not found.
11	UNPACK--End of GRIB message marker not found.
12	UNPACK--GRIB edition number unexpected.
13	UNPACK--GRIB message indicates presence of bit map; not supported.
14	UNPACK--GRIB message has more groups than can be handled.
15	UNPACK--GRIB message indicates missing data in gridpoint data; not supported.
16	UNPACK--Probable error in GRIB message, Section 1. ND7 not large enough.
17	UNPACK--Dimension ND5 not large enough.
18	PACK--Gridpoint data are to be packed, but map is not polar stereographic or Lambert.
19	PACK--A section 3 is indicated, but PACK does not support it.
20	RDI, RDF, and RDC--Error or EOF when reading.
21	RDI, RDF, and RDC--List too long; terminator not encountered.
22	UNPACK--IS4(2) indicates simple packing, which is not supported.
23	UNPACK--More groups indicated than values packed.
24	
25	DATPRO error.
26	DATGEN--Ending date not after beginning date, or INCCYL LE 0.
27	DATGEN--Location to put date not in range 1 thru ND8.
28	DATGEN--Date array too small.
29	RDSNAM--Too many input data sets or an attempt to use Unit No. 10, 97, or an IP( ) value.
30	RDSNAM--Cannot find terminator.
31	RDSNAM, IPOPEN, and WRAA--Trouble opening file.
32	WRAA, RDAA--Trouble reading or writing file.
33	RDSTAD, RDSDIR--Error reading station directory.
34	RDSTAD--Too many stations in directory.
35	RDSTAD--Station not found in directory.
36	RDSTAD--One or more duplicate stations found.
37	RDSTAD--Both errors 35 and 36 have occurred.
38	RDSTRX, FCST71, RDVECT--ND5 not large enough.
39	VRBL61--NWORDS from GFETCH does not match NSTA.
40	RDPRED, RDVRBL, RDVR66--Error reading file.
41	RDPRED, RDVRBL, RDVR66--Too many variables.
42	RDVRBL, RDVR66--No predictors or predictands.

43 CKIDS--Predictor is precipitation amount but interpolation indicator is not 3.

44 CKIDS--Predictor is not precipitation amount but interpolation indicator is 3.

45 CKIDS--Binary indicator is not 0, 1, or 5 for NTYPVR = 1; binary indicator is not 0, 1, 2, 3, or 5 when NTYPVR ≠ 1; binary indicator = 5 and interpolation is not 0, 1, or 2; or smoothing indicator is not 0, 1, 2, 3, 4, or 5.

46 CKIDS, L12D2--Interpolation indicator is not 1, 2, or 3.

47 GFETCH--Data cannot be found.

48 GFETCH--Dimension ND5 not large enough to hold data.

49 GFETCH--ND2(3)*ND2(4) does not equal ND4(3).

50 GSTORE--No slots can be made available in LSTORE(,).

51 GCPAC--Dimension ND5 not large enough to hold logical record when compressing disk. Treated as not fatal in GCPAC and GSTORE.

52 DIFFV, TIMEPV, VERTX, SVRVEC, AVHRLY--NWORDS returned from GFETCH ≠ NSTA.

53 GRCOMB--Too many grid combinations.

54

55 RDSTR2, PRED2--No predictor model number matches model number input.

56 RDSTR2, RDSTRX--No fields found for Day 1.

57

58

59

60 DIRCMP, GWIND, EOWND, MAPLAT--Map projection number not expected. Fatal in DIRCMP. Currently 3 (Lambert) and 5 (polar stereographic) are provided for.

61 BINARY and BINFUL--Threshold value is indicated as missing.

62 BINARY and BINFUL--Binary indicator not a legitimate value.

63 BINFUL--Binary indicator = 0.

64 GRIDB--Threshold value is indicated as missing.

65 GRIDB--Binary indicator not a legitimate value.

66 GRIDB--Reserved.

67 GRIDB--Reserved.

68 TRANS--Transformation indicator not a legitimate value.

69 TRANS--Reserved.

70 LMSTR1--Could not compute maximum time offset.

71 LMSTR1--No fields found for day 1.

90 PRTGR--Overflow after scaling for gridprinting.

99 OPTION, OPTN2, OPTX, L1D, L2D--Predictor not defined.

100 VERTP, TIMEP, VORTW, WSPEED, EOWND, DIVW, TESTS--Grid characteristics not the same for the two fields.

101 VERTP, TIMEP--Array not large enough to hold data. Increase ND5.

102 VERTP, VERTX--IDPARS(5) not accommodated.
TIMEP, VERTP--IDPARS(10) not accommodated.
DIFFV--Processing indicator CxC in IDPARS(2) not accommodated.

103 FORIER, GWIND, VORTH, VORTW, WSPEED, EOWND, DIVW, MIXRAT, DEWPT, LCL, KINDEX--ID's not accommodated in routine.

104

105 LMIJLL--Error in input.

106 FCST71--No data found for day 1.

107

108
109
110
111 L1D1, L2D1--ID in file does not match cccfff.
112 L1D1, L2D1--Too many thresholds.
113 L1D1, L2D1--Too many regions.
114 L1D1, L2D1--Too many threshold combinations.
115 L1D1, L2D1--Unexpected error.
116 L2D3--Error in algorithm.
117
118 L1D1, L2D1--Not all stations needed are on the threshold file.
119
120 FINDST--Station can't be found on input file.
121 RDFORC--Error reading input.
122 RDFORC--Too many forced predictors.
123 RDFORC--Not expected number of forced predictors.
124 RDFORC--Forced predictor not in variable list.
125 XPROD--ND2 not large enough.
126 THSET--Binary threshold number not consistent with variable type.
127 PRED2, PRED22, VRBL62, FCST72--End of input data.
128 SUMRY--Problem reading data.
129
130 WRHEAD and WROPEQ--Error writing operational equations.
131 PLIST--ND13 not large enough.
132 PKMS99--MISSP = 0.
133 PKMS00--MISSP NE 0.
134 PACK1D--XMISSP = 0 and XMISSS ≠ 0, or XMISSP ≠ 0 and IS1(19) ≠ 0.
135 RDSTRX--ND5 not large enough.
136 FCST72--No data found for a particular date.
137 RDVECT, RDSDIR--End of file or data. Not necessarily an error.
138 RDSTRX, RDVECT, PRED22--Errors on input file.
139 RDVECT, GTVECT--Missing data.
140 SKIPWR, RDDIR--Error reading call letters record.
141 SKIPWR--CCALLP( ) not large enough.
142 SKIPWR--Station lists do not match.
143 SKIPWR--Error reading data when skipping records.
144 SKIPWR--Error writing call letters record.
145 RDDIR--Size of call letters record exceeds ND5.
146 RDVECT, RDDIR--End of file.
147 SKIPWR--Impossible situation, code error.
148 SKIPWR--ND5 too small.
149
150
151 FLOPNM--NRW not 1 or 2.
152 RDKEYM, WRKEYM, WRDATM, RDDATM--Record number of physical record size in error.
153 RDTM--No next record to read when reading sequentially.
154 RDTM, FLOPN--Array size NSIZE not large enough to hold data record. Also, in RDKEYM array size NWDS not large enough to hold key record.
155 RDTM--Can't find record to read.
156 WRTM--Replacing record and no record found to replace.
157 WRTM--Not replacing record, but checking IDs and a match found.
158

159	COMBIN--Second digit of CCC not identified.
160	CONST--Station not found in directory.
161	COMPID--No predictors returned.
162	COMPID--Error in COMPID.
163	RDSCR--No score number read.
164	CONST--Disagreement between index values from GFETCH and the data values from RDTDLM.
165	RDEQN--Internal and external file names don't match.
166	RDEQN--Number of predictands = 0 or > ND3.
167	RDEQN--Number of terms in equation = 0 or > ND2.
168	RDEQN--System reading error.
169	RDEQN--Number of equations > ND1 or > ND13.
170	CAT1--ICAT ≠ 1.
171	RDEQN--No equation for one or more of the stations in the station list.
172	TANDID--Error reading predictand/forecast table.
173	
174	
175	WRTDLR--Error reading station directory.
176	WRTDLR--Number of stations on random access file does not match directory, or station list does not match.
900--	IOSTAT--Returns from system.

PARAMETER Statements

Parameter statements are used only for variable dimensions or rarely for physical constants that should have the same value in all routines. The parameter variables for dimensions shall start with "ND" followed (usually) by one or two digits. This way, dimensions can always be easily recognized. Almost always will these be declared in the main program or driver if one exists.

Variable Dimensions

To the extent practicable, these parameter definitions should be used across programs. That is, many programs will use variables dimensioned with a maximum number of stations; ND1 should be used for this purpose.

ND1	U201, U600, and U700--Maximum number of stations.
ND2	U201--Maximum NX of grid.
ND2	U600--Size of storage array Q( ). (ND2 and ND3 are not needed for grid sizes in programs not dealing with grids.)
ND2	U700--Maximum number of predictands.
ND3	U201--Maximum NY of grid. (Grid size is actually determined by ND2*ND3)
ND3	U700--Maximum number of terms in one equation.
ND2X3	U201--ND2*ND3.
ND4	U201, U600, and U700--Maximum number of variables that can be dealt with in one run.
ND5	U201 and U600--Size of 4-byte arrays IPACK( ), IWORK( ), and WORK( ). Generally, the maximum number of values that can be dealt with for each station (including the number in the input directory records), so that ND5 ≥ ND1. Also, ND5

should be ≥ ND2X3 in U201, because the arrays are used for gridpoint data.

ND6	U201, U600, and U700--Maximum number of input data sets in TDLPACK format.
ND7	User Program--Size of IS0( ), IS1( ), IS2( ), and IS4( ). A value of 54 is sufficient.
ND8	U201, U600, and U700--Maximum size of date list IDATE( ). Used in DATPRO and DATGEN.
ND9	U201, U600, and U700--ID size in LSTORE(ND9,).
ND10	U201, U600, and U700--Size of a linear array holding data for random access.
ND11	U201--Number of combinations of grid characteristics that can be dealt with.
ND11	U700--Maximum number of sets of equations.
ND12	Maximum number of random access files (MOS-2000 External File System) that can be used. (Note that this is not the number that can be open at one time.)
ND13	U700--Maximum number of different equations per equation set.
NDG	PACK, UNPACK--Maximum number of groups for packing. Set here so dimensions won't have to be carried, and use for user is simplified. Size is protected against being too small.

Bits Per Word

In a very few cases in the MOS-2000 system, it is necessary to know the word length of the machine. Since the HP workstations are 32 bit and the CRAY is 64-bit, slightly different versions of some routines are necessary for the two machines. One PARAMETER statement is used to set the word length, and sometimes a couple of others are used that depend on that one.

L3264B	Set to 32 for the HP workstations and to 64 for the CRAY.
L3264W	Set to 64/L3264B, which is the number of words in 64 bits.
NBLOCK	6400/L3264B, where the 6400 is somewhat arbitrary. NBLOCK is the block size in words of the internal MOS-2000 storage system. This can vary from program to program. If 6400 is too large, disk space may be wasted; if too small, too many records will be required to hold a logical record.

Physical Constants

These values for these physical constants should be used, rather than some slightly different values (some routines were written before this list was established, and the values may vary slightly).

PI	Pi = 3.1416
RERTH	Radius of the earth (meters) = 6371200.
AGRAV	Acceleration of gravity (m/sec^2) = 9.8062
RADPDG	Radians per degree = PI/180.
OMEGA	Angular velocity of the earth (sec^{-1}) = 7.2921E-5
OMEGA2	Twice the earths rotation speed in (sec^{-1}) = 14.5842E-5
RD	Dry air gas constant (J kg^{-1} K^{-1}) = 287.04
CP	Specific heat of dry air at constant pressure (J kg^{-1} K^{-1}) = 1004.6
RKAPPA	RD/CP = .28573
CV	Specific heat of dry air at constant volume (J kg^{-1} K^{-1}) = 717.60

RV	Water vapor gas constant (J kg^{-1} K^{-1}) = 461.50
CVAP	Specific heat of water vapor at constant pressure (J kg^{-1} K^{-1}) = 1846.0
CLIQ	Specific heat capacity of liquid water (J kg^{-1} K^{-1}) = 4185.5
HVAP	Latent heat of vaporization of water at 0 C (J kg^{-1}) = 2.5000E+6
HFUS	Latent heat of fusion of water at 0 C (J kg^{-1}) = 3.3358E+5
PSAT	Saturation vapor pressure at triple point (pa) = 610.78
SBC	Stefan-Boltzmann constant (Jm^{-2} K^{-4} sec^{-1}) = 5.6730E-8
SOLR	Solar constant (Jm^{-2} sec^{-1}) = 1353.3
ABSZRO	(degrees) = -273.15

Units Conversions

K	= C + 273.15
1 cal	= 4.1855 J
1 m/sec	= 1.9425 kt
1 deg of lat	= 1.1120E+5 m
1 foot	= .30480 m
1 mb	= 100. pa
1 kwh	= 3.6E+6 J

Variable Names

Variable names are listed below that may have applicability across MOS-2000 programs.

RUNID	Information to identify output (CHARACTER*72).
NCYCLE	Cycle of model run, 2 digits.
NSKIP	Number of cycles to skip because of bad data before stopping.
IDATE( )	Date list, dimension of ND8.
NDATES	Number of values in IDATE( ).
IP(J)	Each value (J=1,25) controls a portion of print for a main program. A value of 0 means no print; a nonzero value designates an output unit number. Refer to IPX on a previous page in this chapter for uses of specific numbers.
IER	Status return. 0 = good return.
STATE	Variable set to indicate where in a program a stop occurred. For use in IERX. (CHARACTER*4)
NWORK(J)	Word array for DATPRO, dimension of ND8.
ITEMP(J)	Work array for RDF or RDI. Dimension $\geq$ the number of values per record.
INGRID(J)	Unit numbers for gridpoint data input to U201.
NAMGRD(J)	Names corresponding to INGRID(J).
NUMOD	Number of items in INGRID( ) AND NAMGRD( ).
IPINIT	4 characters to help to identify the input data sets for U201 (CHARACTER*4). Usually the user's initials plus a sequence number (e.g., HRG1).
DIRNAM(J)	20 characters for name of data set where station list resides (J=1) and where directory is (J=2). (CHARACTER*20)
NSTA	Number of stations being dealt with.
ISTOP	1 indicates to stop program after control input.
LBIT(M)	The number of bits required to pack the values in each group (M=1,LX).

JMAX(M)	The maximum of each group M of packed values after subtracting the group minimum value (M=1,LX).
JMIN(M)	The minimum of each group M of packed values after subtracting the group minimum value (M=1,LX).
NOV(M)	The number of values in group M (M=1,LX).
IS0(J)	The values for TDLPACK Section 0.
IS1(J)	The values for TDLPACK Section 1.
IS2(J)	The values for TDLPACK Section 2.
IS4(J)	The values for TDLPACK Section 4.
LSTORE(NID+6,J)	Array to use to store/retrieve data for U201 and possibly other programs.
DIR(K,J,M)	The XI and YJ (J=1,2) positions on the grid for the combination of grid characteristics M (M=1,MGRID) and station K (K=1,NSTA).
MGRID	The number of grid characteristic combinations that can be dealt with.
CCALL(K)	Station call letters (CHARACTER*8).
STALAT(K)	Station latitude.
STALON(K)	Station longitude.
NAME(K)	Station name (CHARACTER*20)
NX	Extent of grid in "horizontal" direction (1st dimension). Note that this is not the PARAMETER defining the size of the array. In some cases, the PARAMETER (normally ND2) will be the extent, but many times the data will occupy only a portion of the declared grid.
NY	Extent of grid in "vertical" direction (2nd dimension). (Also, see NX above.)
IX	Index for grid in "horizontal" direction (1st dimension). For real numbers, use XI.
JY	Index for grid in "vertical" direction (2nd dimension). For real numbers, use YJ.
XNP	North pole position in the IX direction in grid units from the lower left as (1,1). When integer is required, use NPX.
YNP	North pole position in the JY direction in grid units from the lower left as (1,1). When integer is required, use NPY.
FDX(,)	Up to 9 work arrays for grids, where X is a digit 1 through 9 (seven are used in U201).
FDA(,)	Used in PRED21 and PRED22 as temporary storage for a grid. It should not be used in OPTION.
FDM(,)	Map scale factor for polar stereographic map projection.
FDSIN(,)	Sine of the latitude.
XMISS	Missing value indicator, currently = 9999. (May need to be changed.)
ISCALE( )	Scaling exponent when packing data.
SMULT( )	Multiplicative factor when contouring or gridprinting data.
SADD( )	Additive factor when contouring or gridprinting data.
ORIGIN( )	The origin or the contours when gridprinting data. This is with respect to the original values, not after SMULT( ) and SADD( ) have been applied in PRTGR.
CINT( )	The contour interval when contouring data. This is with respect to the original values, not after SMULT( ) and SADD( ) have been applied in PRTGR.
PLAIN	Plain language description of the predictors. (CHARACTER*32)
IPLAIN( )	Plain language description of the predictors in an INTEGER array equivalenced to PLAIN.

UNITS( )	Plain language of the units of data. (CHARACTER*12)
NPRED	Number of predictors being used.
THRESH( )	Thresholds of predictors.
VORT(,)	Vorticity. Computed as /sec, carried as /sec E+5.
UWIND(,)	U-wind component in m/sec.
VWIND(,)	V-wind component in m/sec.
SWIND(,)	Wind speed in m/sec.
XLAT	Latitude at which the mesh length XMESHL applies.
XMESHL	Mesh (grid) length in meters at latitude XLAT.
ORIENT	Orientation of the grid; the longitude in degrees West to which the grid columns are parallel and vertical.
DIV(,)	Divergence. Computed as /sec, carried as /sec E+5.
ITAU( )	The number hours to "look ahead" in programs like U600 and U850 to find the data needed.

C. Main Program Template

Most of the MOS-2000 main programs will have a similar input/output structure. This section prescribes the order for such input/output and how the unit number and file names are to be handled. The input types are numbered; if one type is not needed in a particular program, it can be omitted. A needed type for a particular program that is not specified here can be added; the order to include it will probably be obvious.

Detailed explanations of variables are not included here for brevity, and can generally be found elsewhere. The input and order are:

(1) Read from KFILDI the variables IPINIT and (IP(J), J=1,25) according to the format (A4,25I3). Then immediately call IPOPEN to open the files and perform other necessary functions. Note that it will change the default output file unit number KFILDO from its original value to IP(1) when IP(1) ≠ KFILDO. Print the information read.

(2) Read from KFILDI the variable RUNID according to the format (A72). Print the information read.

(3) Read from KFILDI other control variables needed by the program. Print the information read in an easy to read list.

(4) Read from KFILDI with RDSNAM the unit number KFILDT and name of the file DATNAM where the dates are to be found. Print the information read.

(5) Read the dates with RDI. DATPRO (writeup No. 4.15) can be called for date extensions, etc. Print is controlled by IP(2) and IP(3).

(6) Read from KFILDI with RDSNAM the unit number(s) KFILIN( ) and name(s) of the file(s) MODNAM( ) where the primary input data are to be found. (For U201, this includes a model number). Print the information read.

(7) Read from KFILDI with RDSNAM the unit number(s) KFILRA( ) and name(s) of the file(s) RACESS( ) where the constant input data are to be found. Print the information read.

(8) Read from KFILDI with RDSNAM the unit number(s) KFILIO( ) and name(s) of the file(s) OUTNAM( ) where the primary output data are to be placed. Usually this would be only one file. It may be TDLPACK vector information or is the developed equations for U600. Print the information read.

(9) Read from KFILDI with RDSNAM the unit numbers KFILD(J) and names of the files DIRNAM(J) where the station list (J=1) and station directory information (J=2) are to be found. Print the information read.

(10) Read with RDSTAD the station list and directory information.

(11) Read from KFILDI with RDSNAM the unit number KFILP and name of the file PRENAM where the variable list is be found. Print the information read.

(12) Read from KFILDI with RDSNAM the unit number KFILCP and name of the file CONNAM where the constants for the variables are to be found. This is plain language and other information associated with the variables, not constant data addressed in (7) above. Print the information read.

(13) Read with RDPRED, RDVRBL RDVR66, or some similar subroutine from KFILP and KFILCP the variable list and associated information.

(14) Process data.

(15) Output primary results.

Note that usually the input from a particular file immediately follows the reading of the file unit number and name. Use U201, U600, or U660 as further guides in arranging the input/output as described above.

Certain subroutines are available, and are to be used, especially to facilitate the input and output. Most of these are relevant to more than one program. They are:

RDSNAM - This routine reads from the standard input unit KFILDI three things relevant to an input or output file:

(1) A unit number,
(2) A value that can be a model number or some other needed value. In many instances, this value is not needed and is just a dummy, and
(3) A file name of 60 characters.

More than one value of each can be read and returned, each set from a separate record with the format (2I3,1XA60). The call sequence specifies the maximum number that can be returned, and also, of course, the number read is returned. The routine also opens sequential file(s) specified according to the call sequence parameters and assures that the unit number does not equal any in the IP( ) list (the IP( ) variable is explained elsewhere); note that random access files with reserved unit numbers in the range 45 to 49 are not opened. Reading stops when the maximum number is about to be exceeded or a terminator "99" is reached in the unit number position.

RDI - Reads a list of integer values according to a format in the calling sequence.

RDF - Reads a list of real values according to a format in the calling sequence.

RDC - Reads a list of character values according to a format in the calling sequence.

RDSTAD - Reads a station list (call letters) and other directory information. This can be called multiple times to get multiple lists of stations. The list of stations is returned alphabetized. The user is warned about duplicate stations. Print is controlled by IP(4) and IP(5). (While these exact print control variables are not necessary, use them for uniformity.)

RDSTAL - Reads a station list and other directory information. RDSTAD performs the same function as RDSTAD except RDSTAL does not alphabetize the call letters; they are returned in the order read.

RDSTGN - Reads one or more station lists and directory information. The user is warned about duplicate stations only within a particular list.

RDSTGA - Reads and alphabetizes one or more station lists and directory information. Performs the same function as RDSTGN except it alphabetizes the list(s). The alphabetization is, or course, only within (not across) lists.

RDVRBL - Reads a list of variables in the MOS-2000 format and associated information for U201. This routine will not be exactly what is needed in other programs, but at a minimum should furnish guidance on how to write a similar routine. Also see RDPRED and RDVR66, which are used for U600 and U660, respectively. The reason that these routines cannot be generic is that each major program has a unique set of information associated with the variables.

RDPRED - Reads a list of variables in the MOS-2000 format and associated information for U600.

RDVR66 - Reads a list of variables in the MOS-2000 format and associated information for U660.

CKIDS - Checks internal consistency of a MOS-2000 ID.

CONSTX - Retrieves data from MOS-2000 External Random Access file for vector-oriented programs. (CONST is used for U201.)

COMPID - Computes IDs for data in MOS-2000 External Random Access file system.

DATGEN - Generates date/times between two date/times.

GSTORE - Stores data into Internal MOS-2000 Storage System.

GFETCH - Retrieves data from Internal MOS-2000 Storage System.

IERX - Writes diagnostic message for system errors.

IPOPEN - Opens IPX files used for output.

PACK1D - Packs vector data into TDLPACK format.

PACK2D - Packs gridpoint data into TDLPACK format.

PRSID1 - Parses MOS-2000 ID into component parts.

RDNAME - Reads plain language and scaling parameters from the variable constant file.

SKIPWR - Skips records up to and past a specified date/time and writes directory record if a new one is needed.

TIMPR - Time stamps a specified file or files.

UPDAT - Adds hours to a date/time.

Writeups are available; consult the writeup index.

D. Philosophy in Accessing Data

One of the major difficulties in dealing with large amounts of archived data, necessarily in sequential files, is that the data needed by particular programs is not necessarily needed in the order the data are located in the files. In U201, for example, it is not known until one day (cycle) of processing is completed exactly what variables are needed. This is partly because of the flexibility built into the OPTION computational routines; the data needed are not known until the computational subroutine is used the first time in the run. The tack that was taken in building the MOS-2000 system is that the user need not know, and specify to the program, all of these details, but the programs themselves would "figure out" what was needed and access the data appropriately. Also, it was not desirable to have to maintain the same order of variables on all inputs--inputs that might have been prepared by different persons and/or at different times.

Another major difficulty in the work performed by the MOS-2000 system is that the data needed for processing for a particular day can actually be data for a previous day (in the case of U201) or future day (in the case of most vector programs such as U600). This means that even though one were to know exactly what data are needed for a particular day, data must have been saved from a former day (in the case of U201) or the sequential file(s) have to be read ahead (the lookahead feature) to the day's data needed (in the case of vector programs), and then (possibly) arrangements made to save all needed intervening data for processing on the next day. Otherwise, the sequential files would have to be read repeatedly with sufficient backspacing to recover the needed data. Note that it has been determined that the lookback capability will be needed only by U201 (where it might be desirable to compute predictors from a previous run of a model, or to prepare as predictors observed data from a time previous to the date/time being processed) and that the lookahead feature will be needed for vector programs (where, for instance, U600 needs predictands at a date/time after the basic date being processed--the one for which the predictors are valid).

The MOS-2000 programs have to know from what date to (save and) use the data. That is, for U201, when does the lookback feature apply? For gridpoint data, this is adequately taken care of by RR in the ID (see Chapter 4). This positive value is the number of hours previous to NDATE, the date/time being processed, from which to use the data specified by (the relevant portions of) the 4-word ID. However, since it is desired to be able to read vector data prepared by any program into any program, there is a complication for vector data. That is, a variable prepared by U201 (or collated by U660) will usually have a non negative tau in the ID and may have a non negative RR, but this usually doesn't indicate that the RR is to be employed. That is, the processing has already been done. As a special feature in U201, when the input is vector data (MODNUM( ) = 0), RR is used only when DD in the variable ID = 0.

The lookahead is more complicated, because no direct provision for it is made in the ID. The ID does contain a tau, and in most cases this is the value in the ID with the data. For model predictors, it is the projection of the data from the model at the date/time NDATE (as modified by RR in U201). In U600, where a distinction is necessarily made between predictors and predictands, and the predictands are usually data (such as hourly) with a tau of 0, the tau for predictands only represents the number of hours ahead of NDATE to retrieve and use the data.

However, in U660 and U850, where no distinction is made between predictors and predictands, another variable, ITAU, is used with each ID that indicates explicitly how many hours ahead to acquire the data. So, there could be a non negative tau in the ID and a non negative ITAU. This leaves a slight problem for U600 in which an observation of a few, say 2, hours ahead of NDATE (e.g., at 0000 UTC) is needed as a predictor. A special feature (switch) is used in U600 to use the tau in the ID as lookahead for the case of observations only [CCC = 7xx (and DD ≠ 82 for AEV data)] as predictors (it is already used that way for predictands). It is not convenient to use the variable ITAU because U600 produces equations with the 4-word IDs and no provision is made to carry extra information (e.g., ITAU) with the equations. U600 needs to use these IDs in exactly the same way as U700 or U900 must use them.

To emphasize, the processing indicators in the ID's do not indicate whether the processing is to be done or has already been done. This is especially problematical when vector data prepared by U201 or U660 should be able to be fed back through U201 or U660. That is, any vector data should be able to be input to any program. This imposes certain restrictions on U201 for the use of vector data; however, it is thought these restrictions are not important. That is, most use of U201 will be in preparing predictor data, for which the input will be heavily gridpoint data, or in preparing predictand data, for which the input will be heavily hourly data on which any processing would normally be through the OPTION subroutine. The U201 writeup can be consulted for more details. It is not expected that much processing on vector data will be done outside of U201, except for the computation of binaries which each vector program has the capability to do; any processing that will be done will be through the option subroutine OPTX, and each computational program will need to know just how to handle the computations. ITAU is carried into OPTX.

Another difficulty is that "constant" or climatological data need to be provided for, data that may not exist for each day/cycle, but rather be for each 5th day, each month, each season, etc. Arrangements have to be made to access these data, store them internally as needed, and possibly to interpolate in time to the day/time being processed. Interpolation requires that two (or more) days of each constant be present, and be refreshed as necessary. These data are stored and accessible from the MOS-2000 External Random Storage System (see Chapter 7).

Yet another difficulty is the way the AEV data are identified and stored. Basically, the data are already collated, so to verify a forecast, one does not go forward in the file to the date/time of the verifying observation, but the verifying observation is identified at the same date/time as the forecast, and with an appropriate tau in the ID. (One reason for doing this is that observations are present, as sent from the forecast offices, for each projection. That is, the same "observation" may be there more than once.) The AEV data are archived with a DD of 80 for MOS forecasts, 81 for local forecasts, and 82 for observations. An attempt has been made in each vector program to deal with AEV data, as well as non AEV data, especially in the verification program U850. However, there will likely be situations where if AEV data are used, and especially mixed with non AEV data, complications will arise. For instance, AEV data may not be adequately handled through OPTX--but will they need to be?

Because of the complexity of accessing the various types of data and because of the required options in the various programs, in setting up a new "type of" run, one should always print some data for <u>at least two</u> cycles and verify that the correct data are indeed being accessed and used. This should be standard practice for even simple programs. As one gets used to the programs, simple changes in input can be assumed to work correctly; it is only when setting up for some new types of input need one be concerned.

In order to accommodate the flexibility needed and at the same time achieve acceptable efficiency, the following program structure was adopted (most reference will be to vector programs; a similar structure, but usually different subroutines, is used for U201):

- o Read all control information and initialize as necessary, as outlined above.

- o Read all data from all input sources (except from the MOS-2000 External Random Access files) for the first day (cycle) to be processed and store them in the random access MOS-2000 Internal Storage System. The data identifiers and necessary associated information is stored in the array LSTORE(,J), where J refers to each variable (J=1,LITEMS), and the data themselves are stored in CORE( ) or on disk as necessary (see Chapter 8 for a description of the MOS-2000 Internal Storage System). <u>This is done by subroutine RDSTRX</u> (RDSTR2 for U201).

- o Access the data for the first day and put into the array AA(K,J) (K=1,NSTA) (J=1,NVRBL), and keep track of if and how each variable is used. (Data from "constant" files in the MOS-2000 External Random Access file system are accessed through the option subroutine OPTX and other accessing routines. These constant data are stored and a record kept of what/where etc. in the MOS-2000 Internal Storage System. The explanation that follows does not address these random access data; their processing is handled internally by other processes.) This is done by subroutine VRBL61 for U600, by VRBL67 in U660, and by FILLAA, called from SCOPTA, in U850 (PRED21 for U201, but AA(,) is not used).

- o Process the data in AA(,) and output as necessary. This processing is unique to each program.

- o Prepare a list of all the variables that were used from sequential sources (and needed for following days) in the array MSTORE(,J) (J=1,MITEMS), and mark for elimination the entries (and associated data) in LSTORE(,). Note that this is not done for the random access constant data. This is done by subroutine LMSTR6 (LMSTR1 for U201).

- o Eliminate all data not needed for processing day 2. This is done by subroutine GCPAC.

- o Write certain information pertaining to Day 1 processing, including how many errors were diagnosed and, optionally, where the variables are to be found. This is pretty standard for each program, but is not handled by a specific subroutine.

- o Read the sequential input file(s) up to and just past the last date/time for which any data are needed for day 2; use any variables read in the process that can be used at that time; and store any variables which will be needed later. This processing is determined by the list of variables in MSTORE(,J) and the associated information. For instance, a variable that can be used when read and need not be saved is used and not stored. This is done by subroutine VRBL62 for U600, and VRBL67 in U660, which in turn use RDVECT (PRED22 for U201, which does not use RDVECT).

- o If there are data needed that have not been accessed by reading the sequential files for day 2, the data will already exist in the MOS-2000 Internal Storage System. These data are retrieved, also, by subroutine VRBL62 or VRBL67, which use GTVECT for the purpose (PRED22, which does not use GTVECT).

- o Mark for elimination in the array LSTORE(,) the data not needed for future days. This is done by subroutine LMSTR2 (LMSTR2 is for U201 and all vector programs).

- o Eliminate all data not needed for processing the next day. This is done by subroutine GCPAC, the same one used for Day 1.

All other days are handled by the previous four steps. At the end of all input (all dates processed), some additional or summary information may be output, or in the case of U600 and U850, the primary processing and output is done at the end of all input of data.

For programs that can function by putting data into an AA(,) array and then processing them, the above structure and subroutines should be used.

Routines VRBL62, VRBL67, RDVECT, and GTVECT, together with the other capabilities of MOS-2000, provide for two levels of data caching. One is the MOS-2000 Internal Storage System referred to above and explained in Chapter 8. The other is internal to the four subroutines VRBL62, VRBL67, RDVECT, and GTVECT (for vector programs), and consists of saving a variable being processed in array SDATA( ) when it is believed to also be the next variable to be processed. This is especially important in computing binary variables; the continuous variable need be found only once (possibly from disk or read directly from a sequential file), and then reused as many times as necessary to produce the binaries. This might, for instance, keep the variable from being entered into the MOS-2000 internal Storage System and the program from incurring the necessary overhead in storing it and in retrieving it and other variables with the longer LSTORE(,) list. The same kind of second level caching is performed in U201, with the data being saved in array FDA( ) in PRED22.

It is noted that a simple program that needs only one input variable from a sequential file need not have the above complicated structure. Rather, each day can be processed in the same way--read the sequential input file until the variable needed is found, process, and move on to the next day. Subroutine RDONEV can be used for the reading and return of the needed variable. While RDONEV could be used for more than one variable in multiple calls, the user would have to know in advance the order of the variables requested on the input file(s).

D. Documentation

Each main program and subroutine shall be documented if it is to have any lasting value in the MOS-2000 system. However, the documentation may be brief for routines that will not be called from new code to be written (e.g., the subroutine is specific to one higher level routine or is part of a subsystem and would not be used outside that subsystem, such as WRTM) or where the call sequence is so standard to MOS-2000 that the use of the routine can best be accomplished by looking at an example rather than by reading a writeup explaining all the variables. Examples of complete documentation are U201 and RDTDLM; examples of limited documentation are WRKEYM and L1D1.

Documentation is also addressed in Chapter 9.

E. Directory Structure and Use

Each developer will have a set of directories with established names for holding specific files. In addition, there is an "official" set of directories and contents. Names have been standardized in order for other than the owner to find subroutine source files and data. Agreed-upon names and conventions makes sharing of data and routines easier, and is important when someone leaves and others have to pick up the pieces. This is also very important in the seasonal redevelopment of equations; redevelopment by adding a season of data must become very quick, easy, and routine.

The "official" directory structure on blizzard, with home accounts under "home21" and contents are as follows:

```
  home21/tdllib/dru201  --  Examples of a U201 driver (dru201.f), a command
                            that creates an executable (hereafter, called a
                            load line) (u201.com), and a control file
                            (U201.CN).
    "      "   /u201lib --  All source (.f) and object (.o) files specific to
                            U201, a makefile (u201-makefile), an archive
                            library (libu201lib.a), and a README file.

    "      "   /dru660  --  Examples of a U660 driver (dru660.f), a load line
                            (u660.com), and a control file (U660.CN).
    "      "   /u660lib --  All source (.f) and object (.o) files specific to
                            U660, a makefile (u660-makefile), an archive
                            library (libu660lib.a), and a README file
```

etc. for all major, general use, main programs. In addition, there is a general library:

```
  home21/tdllib/moslib --   All source (.f) and object (.o) files not specific
                            to individual programs, a makefile (mos-makefile),
                            an archive library (libmoslib.a), and a README
                            file.
```

Each load line in the druxxx directories shall be of the form (the example is for U201, the same pattern is used for all directories/programs):

```
  /opt/fortran/bin/f77 +O2 +77 -o u201 dru201.o \
                               -W1,-L,/home21/tdllib/u201lib -lu201lib \
                               -W1,-L,/home21/tdllib/moslib -lmoslib
```

Note that the specific library is first and the general last. If the libraries contain the correct routines, it will not be necessary to load more than one specific library and the general one. It should be noted that for any routine in a library that calls another routine, the called routine must be in that same library or one later in the load line. In the example above, a routine in u201lib can call one in moslib, but not vice versa.

Each developer shall also have "dr" directories that mirror the druxxx official ones. For instance, Glahn would have directories with names:

```
  home21/glahn/dru201 -- A U201 driver (dru201.f and dru201.o), a load line
                         (u201.com), and a control file (U201.CN)

    "      "  /dru660 -- A U660 driver (dru660.f and dru660.o), a load line
                         (u660.com), and a control file (U660.CN)
```

The druxxx directory is used by the developer to run programs. It can be "self contained" except for the general purpose libraries and certain constant data sets and data. That is, the normal Fortran ftn12 output file will be in the same directory, as would be the print caused by the IP( ) variables in each program (see Section B of this chapter). Output can also be put here, and there is little reason to put it elsewhere. (After a run is deemed correct, the output file can be easily moved elsewhere if needed. An individual's output should not be put directly into the "official" area, see below.) However, a user could have a directory dedicated to files output from some program or programs and input to others.

For checkout of a new routine, put the routine source in the druxxx file, compile it with the "com" script (or "coml", "comd", or "comdl", see below), and insert it into the load line. For instance, suppose you have two new routines named new1 and new2. The load line in the Glahn directory home21/glahn/dru201 would then be:

```
/opt/fortran/bin/f77 +O2 +77 -o u201 dru201.o new1.o new2.o \
                             -W1,-L,/home21/tdllib/u201lib -lu201lib \
                             -W1,-L,/home21/tdllib/moslib -lmoslib
```

Note that by having the .o files instead of the .f files in the load line, you can control whether or not you compile with the /D option for each routine in the load line; having .f files in the load line causes recompilation on each execution, and either /D or not for all routines (see below).

In addition to the official libraries, there are certain data that are maintained in the directory:

home21/tdllib/table -- The contents include:

/station.tbl	--	the station directory (see Chapter 10).
/mos2000id.tbl	--	the MOS ID table that is used by most programs for plain language and packing factors (see Chapter 11).
/hrlydict.tbl	--	the hourly dictionary, not used by most programs.
/conxxxxx.tbl	--	the random access constant files (xxxxx to indicate which of, perhaps, several files) created by U350 and written by U351 or U352 containing data with CCC between 400 and 499 (see Chapter 14).

A staging area has been created into which routines are put before they are moved by a designated librarian into the corresponding "official" directory, for example:

home21/glahn/staging/u201lib	--	Source files (.f) specific to U201 and a README file
" " " /u660lib	--	Source files (.f) specific to U660 and a README file
" " " /moslib	--	Source files (.f) not specific to individual programs and a README file.

When a routine is modified or added to a library, running the makefile (e.g., u201-makefile), will compile the routine and (once the name is added to the makefile[1]) put it into the library ready for linking. If you need to work with a routine during checkout (e.g., compile with the /D option or put in print statements), copy it to your "dr" directory and follow the procedure above for new routines for checkout. That is, compile it and add it to your load line. Leave it there only as long as you need it for checkout. If you find an error, make it known to the librarian and your supervisor.

Individual compilations, with the appropriate optimization, are easily done with the scripts located on the HP workstations:

```
com   -- compiles with no .l file and without the /D option
coml  -- compiles without the /D option and produces a .l file
comd  -- compiles with the /D option and does not produce a .l file
comdl -- compiles with the /D option and produces a .l file
```

That is, compilation of the u201 driver with a .l file in the dru201 directory would be done by the command:

```
coml dru201
```

So a new run of u201 with only a change to one of the new subroutines, new1, can be made with two commands:

```
coml new1
u201.com
```

Note that the com, coml, comd, and comdl scripts do not use the .f in the file name of the source--less typing each time. The .l file contains very useful compilation information including a cross-reference list.

Once a good, complete run of U201 is made, the results will likely need to be moved by the MOS librarian to a common data area maintained by the MOS librarian, and a record kept of the file name, contents, etc. In addition, the .CN control file shall be saved in an area (directory) with a name that will associate it with the output file. In this way, a season of data, for instance, can be added to the output file by using the same control file with only the date list changed and the appropriate KSKIP variable used (or alternatively, a new file created, similar to the first, with the date list changed and with KSKIP = 0).

---

[1]This is a necessary step--easy, but easily forgotten--to be performed by the librarian. The routine name is added to the "LIB_SOURCES=" list. Another feature that can cause consternation is that removing a source from the LIB_SOURCES list does not automatically remove it from the library. To remove it from the library, trash the archive library and remake. Otherwise, if you were moving a routine from one library to another, it might then exist in both, and when both libraries were loaded the old routine might be the one loaded; this is a very difficult problem to diagnose, especially if the problem is not discovered soon after moving the routine.

One or two of the MOS-2000 programs are too large to compile with optimization without a special command. This command is imbedded in the macros:

comn -- compiles with no .l file and without the /D option
comln -- compiles without the /D option and produces a .l file
comdn -- compiles with the /D option and does not produce a .l file

That is, to compile U600 with optimization, no matter its size, type:

```
comn u600
```

There are probably various ways for you to use these scripts. One way is to copy them from home21/glahn/bin to your bin directory.

Not all procedures have been worked out yet. The point is--you or someone else must be able to rerun the program and get the same results. That is, there must be an association, probably by recognizable name, between all files used in a run. It is likely there will be a standard date list associated with a particular model. There will probably also be a standard station list, at least by variable. While these can be accessed on a file other than the .CN file, it may be easier to copy the "standard" files to the .CN file, then there will be fewer files to keep track of. Whatever method is accepted, it should be used by all.

The accounts on blizzard are used as examples above. The only difference between directories on blizzard and on chinook, the WWB HP machine, is the home account name. Procedures as similar as practical shall be maintained for the main-frame computer.

F. Opening and Writing Files

The names of files to be written and read and their associated Fortran unit numbers are read by the subroutine RDSNAM (see TDL Office Note 00-2 for information about RDSNAM). RDSNAM can return up to the number of file names and unit numbers specified in the call. The file numbers and names are always followed by a terminator, 99. It may be that no files are desired, in which case the number of files returned is zero, and the unit number is zero. The list of files read by RDSNAM can have the same unit number; when a sequence of the same unit number occurs, only the first in the series is opened. The default input file with name KFILDI is not opened, because it will have already been opened by the calling program. The calling program should use "NOT" in the call sequence when the unit numbers to be read are in the range 45 through 49 because those numbers are reserved for the MOS-2000 external random access file, and all file usage is handled by the software of that random access system (see Chapter 7).

When the unit number comes back zero (no files available for reading in the .CN control file), no file is opened. In this case, the calling program should make sure nothing is written to that unit number or an ftn00 file will be automatically opened and written.

When a file is to be opened as "OLD" and no file of that name exists (in the path indicated), then RDSNAM tries to open a file with that name as "NEW."

This is safe, because it is known at that point that no file of that name exists, and no overwriting will occur.

When a file is to be opened as "NEW" and a file with that name already exists, RDSNAM will stop with a diagnostic; this is to insure that data are not overwritten by accident.

Table 3-1 indicates the way the main programs and RDSNAM handles output sequential files.

Program	Unit Number	Output	Form of Output	Type of Opening	Action when Unit Number = zero	Action When Trouble Opening File
INT170	KFILIO	Vector	TDLPACK	NEW	Does not write	When file already exists, RDSNAM halts
INT170	KFILAS	ASCII	Formatted	NEW	Same as above	Same as above
U201	KFILIO	Interpolated Output	TDLPACK	NEW	Same as above	Same as above
U350	None					
U351	None					
U352	None					
U520	IFUNIT	QC'ed dates	Formatted	NEW	Writes ftn00	Same as above
U520	IPUNIT	Vector	TDLPACK	NEW	Writes ftn00	Same as above
U520	IEUNIT	Error listing	Formatted	NEW	Does not write	Same as above
INT600	KFILEQ	Equations	Formatted	NEW	Same as above	Same as above
INT602	KFILEQ	Equations	Formatted	NEW	Same as above	Same as above
INT660	KFILIO	Vector	TDLPACK	NEW	Same as above	Same as above
U700	KFILFC	Forecasts	TDLPACK	NEW	Same as above	Same as above
INT710	KFILIO	Vector	TDLPACK	NEW	Same as above	Same as above

Program	Unit Number	Output	Form of Output	Type of Opening	Action when Unit Number = zero	Action When Trouble Opening File
INT830	KFILAS	ASCII	Formatted	NEW	Same as above	Same as above
INT830	KFILIO	Thresholds	TDLPACK (Not implemented)	NEW	N/A	N/A
INT850	KFILIO	Verification Scores	Not implemented			
U900	KFILFC	Forecasts	TDLPACK	NEW	Does not write	When file already exists, RDSNAM halts
INT910	KFILIO	Vector	TDLPACK	NEW	Same as above	Same as above

4. VARIABLE IDENTIFICATION

A. Numerical Definitions

The 16 variable definitions and processing indicators in the identification (ID) are described in terms of the digits in the 4 integers below, which are designated as ID(1), ID(2), ID(3), and ID(4), respectively.

CCC FFF B DD V LLLL UUUU T RR O HH τττ **WXXXXYY I S G**

Each of these integers is 9 digits long, except ID(4) which can have a 0 or 1 as the leftmost (10th) digit, therefore fitting into 32 bits (2^{31} = 2,147,483,654). The parts of this ID are parsed into the variables IDPARS(J), J=1,15 and THRESH. The definitions of these parts are given below. For computed variables, the name of the subroutine that does the computing is indicated in parentheses. TDL management shall be notified of the use of a new ID so that duplicate definitions will not come into use.

IDPARS(1) IDPARS(2)

CCCFFF- Three-digit variable class (CCC) and subclass (FFF).

0xxxxx -	<u>Model and related variables</u>
001xxx -	**Mass**, for example
001000 -	Geopotential height in meters
002xxx -	**Temperature**, for example
002000 -	Temperature in deg K
003xxx -	**Moisture**, for example
003000 -	Relative humidity in percent
003210 -	6-h total accumulated precip in millimeters
004xxx -	**Horizontal winds**, for example
004000 -	U-wind with respect to the grid in m/sec
004002 -	Geostrophic U-wind with respect to the grid in m/sec
004010 -	U-wind with respect to north in m/sec (isobaric)
004011 -	U-wind with respect to north in m/sec (isohyetal)
004066 -	U-wind with respect to north in m/sec
004012 -	Geostrophic U-wind with respect to north in m/sec
004100 -	V-wind with respect to the grid in m/sec
004102 -	Geostrophic V-wind with respect to the grid in m/sec
004110 -	V-wind with respect to north in m/sec (isobaric)
004111 -	V-wind with respect to north in m/sec (isohyetal)
004166 -	U-wind with respect to north in m/sec
004112 -	Geostrophic V-wind with respect to north in m/sec
004200 -	Wind direction from isobaric U and V in deg
004201 -	Wind direction from isohetal U and V in deg
004202 -	Wind direction from geostrophic U and V in deg
004210 -	Wind speed computed from U and V in m/sec (isobaric)
004211 -	Wind speed computed from U and V in m/sec (isohyetal)
004212 -	Geostrophic wind speed in m/sec

005xxx -	**Vertical motion**, for example
005000 -	Vertical motion in Pa/sec
006xxx -	**Fluid flow functions**
006010 -	Vorticity from U- and V-winds. (VORTW)
006020 -	Geostrophic vorticity from heights. (VORTH)
006110 -	Divergence from U- and V-winds. (DIVW)
007xxx -	**Stability-related**, for example
007100-	CAPE in Joules/kg
008xxx -	**Observable sky and weather**
010xxx -	**Trigonometric functions**, for example
010201 -	Sine of day (in NDATE) of year. (FORIER)
010202 -	Cosine of day (in NDATE) of year. (FORIER)
010203 -	Sine of 2*day (in NDATE) of year. (FORIER)
010204 -	Cosine of 2*day (in NDATE) of year. (FORIER)
010205 -	Sine of day (in NDATE+tau) of year. (FORIER)
010206 -	Cosine of day (in NDATE+tau) of year. (FORIER)
010207 -	Sine of 2*day (in NDATE+tau) of year. (FORIER)
010208 -	Cosine of 2*day (in NDATE+tau) of year. (FORIER)
1xxxxx -	(Not assigned)
2xxxxx -	Statistical and local forecasts, for example
20xxxx -	MOS
21xxxx -	Calibrated MOS
22xxxx -	Modified perfect prog
23xxxx -	Calibrated perfect prog
24xxxx -	Klein-Lewis perfect prog
2x2000 -	Temperature
2x2001 -	Daytime max
2x2006 -	Calendar day max
2x2011 -	Nighttime min
2x2016 -	Calendar day min
2x3xxx -	Moisture variables
28xxxx -	Reserved for U850
29xxxx -	Reserved for U850
3xxxxx -	Combination of models[1]
4xxxxx -	Constants[2], for example

---

[1]The MOS-2000 ID system does not provide for two models contributing to the same variable. This can be accommodated if the need arises by specific subroutines, where the "switching" is done on the basis of CCC = 3xx. A "pattern" subroutine COMBIN exists.

[2]While a constant, as a predictor in an equation for instance, can be defined by the CCCFFF, constant data stored for access must have an auxiliary definition as to what month, day, etc. the data applies. The LLLL in the second word of the ID has been

400001 - Station call letters
400003 - Station WBAN numbers
400004 - Block/station numbers
400005 - Station Elevations
400006 - Station latitude
400007 - Station longitude
40xxxx - Constants not dealing with time
41xxxx - Daily values (LLLL = day of year)
412006 - Calendar day max
412016 - Calendar day min
42xxxx - 5-day values (max and min)
4x2xxx - Temperature
422006 - Calendar day max (every 5 days)
422016 - Calendar day min (every 5 days)
43xxxx - Monthly values (LLLL = month numbers 1-12 + 1000) to be interpolated to daily values (see 47xxxx below)
433624 - Monthly rel. freq. precip $\geq$ .01 in 00-12z
44xxxx - 2-seasonal (April-September; October-March) (LLLL = 1017 for summer or 1018 for winter)
45xxxx - 4-seasonal (Mar.-May; June-Aug.; Sept.-Nov.:Dec.-Feb.) (LLLL = 1013, 1014, 1015, 1016 for spring, summer, fall, and winter, respectively)
46xxxx - Yearly values (LLLL = 1019)
47xxxx - Monthly values (LLLL = month numbers 1-12 + 1000) not to be interpolated to daily values (see 43xxxx above)

5xxxxx - 1-d transformations (linearization), for example The ID scheme for 1-d linearization for CCCFFF is:
50xxxx - Indicates a specific combination of season number (e.g., summer, winter), projection number (e.g., 6-h increments, then repeat), and cycle formulation.
51xxxx - Indicates another specific combination of season number, projection number, and cycle formulation.
5xyxxx - Represents a particular model variable "y." Note that the CCC for all model variables is 00x so that a model variable can be linearized with a specific combination of season number, projection number, and cycle formulation by replacing CCC with CCC-500 in the routine L1D50CFFF, replacing CCC with CCC-510 in the routine L1d51FFF, etc. and using FFF as is. If other than a model variable were to be linearized, this scheme would break down, and a CCC of, perhaps, 59xxxx would be used, and then further switching done within L1D. For instance:
501000 - 1-dimensional linearization of 850-mg height (L1D501000)
506020 - 1-dimensional linearization of 850-mb geostrophic vorticity (L1D506020)

---

allocated for this purpose. That is, it is not foreseen that more than one "level" will have to be specified for a constant, and when only one level is needed, UUUU is used. The tau in the ID plus the date being processed determine the date for which the constants are needed.

6xxxxx - <u>2-d transformations</u> (linearization), for example
601000 - 2-dimensional linearization of 850- and 1000-mb heights (L2D601000)
600101 - 2-dimensional linearization of 850-mb geostrophic vorticity and 1000-mb height (L2D600101)

7xxxxx - <u>Observations</u>, for example
78xxxx - Reserved for U850
79xxxx - Reserved for U850

8xxxxx - (Not assigned)

9xxxxx - (Not assigned)

IDPARS(3)

B - Binary indicator.

0 - Continuous field, no (generic) binary operation indicated. (Binary operations could be imbedded in other computed variables with their own CCCFFF.)
1 - Point binary (cumulative from above). THRESH (see definition below) is used as the threshold or cutoff value. When the variable is $\geq$ THRESH, the binary takes the value of 1, and 0 otherwise. (Subroutine BINARY)
2 - Point binary (cumulative from below). THRESH is used as the threshold or cutoff value. When the variable is < THRESH, the binary takes the value of 1, and 0 otherwise. <u>Not</u> to be used as a <u>predictor</u>. (Subroutine BINFUL)
3 - Point binary (discrete). THRESH for this variable (TRESHU) and THRESH (TRESHL) for the "same" variable but with the next lower threshold are used together. When the variable is $\geq$ TRESHL and < TRESHU, the binary takes the value of 1, and 0 otherwise. (When there is no lower threshold, this discrete binary is the same as a cumulative from below.) <u>Not</u> to be used as a <u>predictor</u>. (Subroutine BINFUL)
4 - Not used.
5 - Grid binary. THRESH is used, and the binary orientation is the same as for the point binary. (Subroutine GRIDB)
6 - A "matching or" variable (used in U850). See Treatment of Thresholds below. (Subroutine BINARY)
7 - A "matching or" variable (used in U850). See Treatment of Thresholds below. (Subroutine BINARY)
8 - A "matching" variable (used in U850). See Treatment of Thresholds below. (Subroutine BINARY)
9 - A "matching" variable (used in U850). See Treatment of Thresholds below. (Subroutine BINARY)

IDPARS(4)

DD - Data source.

00 - Variables not depending on model (e.g., elevation).
06 - NGM model.
07 - Eta model.
08 - AVN model.
09 - MRF model.
80 - MOS AEV forecasts (see Chapter 16).
81 - Local (LCL) AEV forecasts (see Chapter 16).

82 - Observations matching AEV forecasts (OBS) (see Chapter 16).

For U201, most data will come from sequential files, and there is not necessarily any correspondence between DD and the unit number on which the data set is read. However, "constant" data (those with CCC in the range 400-499, e.g., monthly frequencies) are on random access files and the DD represents (for U201 only) the unit number associated with those data (see U201 writeup).

IDPARS(5)

V - Vertical application indicator. It indicates the action to take concerning the field defined by CCCFFF at the two levels defined by LLLL and UUUU. (Subroutine VERTP)

0 - No vertical processing.
1 - Difference, levels (UUUU-LLLL).
2 - Sum, levels (UUUU+LLLL).
3 - Mean, levels (UUUU+LLLL)/2.

IDPARS(6)

LLLL - The lower level to be used in the computation indicated by V. When V = 0, LLLL can = 0. When a computation involving two levels is done, the optional gridprinted map will be labeled with UUUU-LLLL.

As noted in a previous footnote, "constant" data (those with CCC in the range 400-499, e.g., monthly relative frequencies) need to be identified as to the data/time. For those variables, LLLL will be used for this additional information.

IDPARS(7)

UUUU - The upper level to be used with LLLL. When only one level is involved, UUUU will indicate that level, and LLLL will be zero. For instance for a 500-mb height, UUUU = 0500. (This value is also used to identify gridprinted maps and other output, and while other vertical coordinate systems could be used, the map may be labeled in meters rather than the correct units.) Values of LLLL and UUUU with V = 0 can indicate a basic field, for instance a mean over a layer. However, in doing a computation with the general facility (subroutine VERTP), only two variables can be included. If more levels or layers than two have to enter the computation, then the computation must be done in a special subroutine with a different CCCFFF.

IDPARS(8)

T - Nonlinear transformation of the interpolated variable. Note that this is done after the interpolation. (Subroutine TRANS)

1 - Square.
2 - Square root.

IDPARS(9)

RR - Run time offset in hours. The effect of RR is to use a previous date for the variable rather than the date (NDATE) currently being processed. For instance, when RR = 24 with a model field, the field will be from the model run 24 hours ago. This will be more fully explained below. RR, O, and HH are used together. (Subroutine TIMEP)

IDPARS(10)

O - Time operator. As explained below, two fields are involved in the time computation. Values 1 through 4 are used for different projections from the same run; as such a forecast change over time can be computed. Values 5 through 8 are used for different projections from different runs. The basic field indicated by CCCFFF--and even when V ≠ 0 by V, LLLL, and UUUU--is the same for both times. That is, the vertical computation is done before the time computation. This might be useful when computing the time change of two thickness values, the latter computed from geopotential heights.

For values 1-4, the run time for each variable is the NDATE-RR (the basic date offset by RR hours); the projection for variable 2 is tau (the projection τττ, see below) and the projection for variable 1 is tau+HH.

For values 5-8, the run time for variable 2 is NDATE and for variable 1 is NDATE-RR; the projection for variable 2 is tau and the projection for variable 1 is tau+HH.

1,5 - Mean.
2 - Difference, variable 1 minus variable 2. For a run of the same time, a positive difference means that the value has increased with increasing projection.
6 - Difference, variable 2 minus variable 1. For two fields valid at the same time (RR=HH), a positive difference means that the current run was the higher (value increased with run time).
3,7 - Maximum of the two variables.
4,8 - Minimum of the two variables.

IDPARS(11)

HH - The time period in hours between the projections of the two variables. Variable 2 has the projection tau and Variable 1 has the projection tau+HH.

Note that when RR = HH, both variables verify at the same time, but are from different runs and have different taus.

IDPARS(12)

τττ - The projection of the variables(s). When time computation is done (O ≠ 0), this is the tau used with HH (see above). When vertical computation is done, (V ≠ 0) both fields are of the same tau, τττ. Do not use values >899; 9xx is reserved for possible negative values (not yet implemented).

IDPARS(13)

I - Type of interpolation. Interpolation is done after smoothing, but before computations indicated by T ≠ 0.
- 0 - Interpolation is not needed, as in the case when the values of the variable being computed or retrieved from a data source (such as climatic values) are already at station locations.
- 1 - Biquadratic. Bilinear is used in the outside border, and linear extrapolation outside the grid. (Subroutine INTRPA)
- 2 - Bilinear. Linear extrapolation is done outside the grid. (Subroutine INTRPB)
- 3 - Special interpolation for precipitation amount, and possibly other variables where the field is mostly zero with some areas of positive values. (Subroutine INTRP)

IDPARS(14)

S - Smoothing indicator. Smoothing is done as the last step before interpolation.

- 0 - No smoothing.
- 1 - 5-point. (Subroutine SMTH5)
- 2 - 9-point. (Subroutine SMTH9)
- 3 - 25-point. (Subroutine SMTH25)
- 4 - 81-point (25-point applied twice). (Subroutine SMTH2X)
- 5 - 169-point (25-point applied trice). (Subroutine SMTH3X)

IDPARS(15)

G - Reserved. A possible use is with grid-length changes. For instance, an archive for a particular model will have a "basic" gridlength (which may vary by type of variable). However, that gridlength might change over the archive period. For some computations, such as vorticity, the computations can be carried out the same way no matter what gridlength is used (although the noise and amplitude characteristics might be somewhat different). However, for the smoothing, a 5-point smoothing on an 80-km gridlength field will be considerably different than a 5-point smoothing on a 40-km field. For some computations, an adjustment may have to be made for gridlength changes; this variable G is reserved to signal that process. (Note that the gridlength will be available with each field so that no assumption need be made in that regard.)

THRESH

WXXXXYY - While the digits in the threshold will pack into the 4th word as indicated, it is really used as the variable THRESH. It is read as a REAL (floating point) word represented as S_1.XXXXS_2YY, where S_1 is the sign of the value which becomes W = 0 for positive or 1 for negative, XXXX is the fraction to which the exponent is applied, S_2 is the sign of the exponent, and YY is the exponent. Before being stored in ID(4), 50 is added to YY when its sign is negative.

See the appendix to this chapter for a discussion of the considerations leading to the ID scheme.

B. Plain Language Definitions

The file 'MOS2000PRED' contains records with somewhat generic plain language identification (along with other information associated with scaling, units, etc.). Although the full 4-word ID is provided for in each record, to date only the first word is used. So when the cccfff in a record matches the cccfff of the predictor, the associated plain language in the record is used.

The plain language consists of 32 characters, the first 5 of which are left blank and are filled in by U201 with the level UUUU in ID word 2 when "V" in ID word 2 = 0. Suppose the file contained '^^^^^GEO REL VORT^^^^^^^^^^^^^^^' associated with the cccfff = 006020, and the ID word 2 = 000000850, then the plain language output would be:

```
' 850 GEO REL VORT               '
```

If "V" ≠ 0, then the information is moved to the right and both levels inserted separated by a "-", "+", or "A" indicating a difference, sum, or average, respectively. For the example above, with word 2 = 110000850, the plain language output would be:

```
' 850-1000 GEO REL VORT          '
```

When a time lag is indicated by the "O" in ID word 3 ≠ 0, "LX" is inserted into characters 24 and 25, where "X" is the value of "O". Suppose "O" = 1, the plain language for the same example would be:

```
' 850-1000 GEO REL VORT L1       '
```

If the predictor were to be made into a binary or grid binary, characters are inserted into positions 28 and 29, so that the output would be, respectively:

```
' 850-1000 GEO REL VORT L1   B   '
' 850-1000 GEO REL VORT L1  GB   '
```

If 5-, 9-, 25-, 81, or 169-point smoothing were to be performed on the grid binary, the output would be, respectively:

```
' 850-1000 GEO REL VORT L1  GBS5 '
' 850-1000 GEO REL VORT L1  GBS9 '
' 850-1000 GEO REL VORT L1  GBS25'
' 850-1000 GEO REL VORT L1  GBS2X'
' 850-1000 GEO REL VORT L1  GBS3X'
```

If a transformation were indicated by "T" ≠ 0 in ID word 3, a "T" would be inserted into the last example thusly:

```
' 850-1000 GEO REL VORT L1 TGBS25'
```

If a 1-dimensional or 2-dimensional linearization were to be done, the 9-point smoothing example above would, by inserting "D1" or "D2", become respectively:

```
' 850-1000 GEO REL VORT L1D1GBS9 '
' 850-1000 GEO REL VORT L1D2GBS9 '
```

In a like manner, for "constant" data (CCC in the range 400-499), a "C" is inserted in position 26:

```
      '     CAL DAY MAX TEMP    C      '
```

Note that the "T" and "D1" conflict as to location, but they would never be used together, so there is no problem.

The plain language is placed by subroutine SETPLN for each program. See that writeup for the current capabilities of plain language use by MOS-2000 programs; it should be kept up to date, while this chapter may not be in this regard.

C. Order of Processing

The order of processing imposed is as follows:

1. Grid offsets (not currently implemented).
2. Computations of variables such as divergence.
3. Vertical operations.
4. Time operations.
5. Grid binaries.
6. Spatial smoothing.
7. Interpolation.
8. Transformation of point variables.
9 Point binaries.

D. Treatment of Thresholds

One threshold is provided within a variable ID. This is parsed out into a variable called THRESH. For predictors, since the ID in the equation can contain only one threshold, only one type of binary is allowed--B = 1. This cumulative (from below) binary is set to 1 when the original value is ≥ THRESH, and zero otherwise. U201 is used to create data for input to other programs, each of which has a cumulative binary-making capability; therefore, the use of binaries in U201 should be fairly unusual except possibly for computations through OPTX, in which case special binary makers may be needed. U201 uses the subroutine BINARY which deals with B = 1, 6, 7, 8, and 9.

Programs other than U201 (e.g., U600, U660, U850, U700) have their own binary capability and all use the subroutine BINFUL. BINFUL can deal with B = 1, 2, and 3. For these programs, two variables are defined corresponding to each ID, TRESHL and TRESHU; these are the lower and upper thresholds, respectively.

The making of binaries is done by BINARY or BINFUL as indicated below:

B	TREHSL	TRESHU	COMMENTS
0	0	0	TRESHL and TRESHU would not be used.
1	THRESH	99999	Binary, cumulative from above. Set data value to 1 when the original value is ≥ THRESH (and < 99999) and 0 otherwise.

2	-99999	THRESH	Binary, cumulative from below. Set data value to 1 when the original value is < THRESH (and > -99999) and 0 otherwise.
3	(THRESH-1)	THRESH	Discrete binary. Set data value to 1 when the original value is ≥ TRESHL and < TRESHU and 0 otherwise. The value for TRESHL in this case is the threshold for the variable in the list just preceding the variable being dealt with when the two IDs are exactly the same except the threshold of the preceding one is < the one being dealt with. This means that when the subroutine THSET is called, which places the values in TRESHL and TRESHU (in programs other than U201), the variables must be "in order." This order is guaranteed (for predictors and predictands separately) in U600, but not in other programs, because the order may be important to the user. (This does pose some restrictions, but they are probably unimportant. These restrictions stem from the fact that only one threshold can be carried in the ID and the desire to, for convenience, limit other input concerning each variable.) In U850, for example, the input IDs must be in a particular order to achieve the desired result, and this is in agreement with the use of thresholds.
4	N/A	N/A	Not used.
5	THRESH	0	Grid binary, cumulative from above. Set data value to 1 when the original value is ≥ THRESH and 0 otherwise.
6	THRESH	0	"Matching or" binary, cumulative from above. Set data value to -9999 when the original value is > THRESH and 0 otherwise.
7	THRESH	0	"Matching or" binary, cumulative from below. Set data value to -9999 when the original value is < THRESH and 0 otherwise.
8	THRESH	0	"Matching" binary, cumulative from above. Set data value to 9999 when the original value is > THRESH and 0 otherwise.
9	THRESH	0	"Matching" binary, cumulative from below. Set data value to 9999 when the original value is < THRESH and 0 otherwise.

In programs in which the binaries have been defined by BINFUL, the printed output will correspond to the way the variables are defined (e.g., output package SCR21C in U850). In some cases, where categories are hardwired for convenience and subroutine BKCAT is used to determine a category number (e.g., output packages SCR31C and SCR41C in U850), the values may be determined as $\leq$ the threshold and printed as such.

APPENDIX TO CHAPTER 4

Considerations in Designing the ID Scheme

The identification scheme for the variables used in MOS-2000 is very important. Some of the considerations in designing the scheme were:

- o The variable would be easy to recognize by its identification (ID), especially the "basic" predictors (those from models), but also, at least as to a category (such as moisture), the computed predictors.

- o Processing information would be provided that would allow basic predictors, and even computed predictors, to be processed by generalized routines,

- o The ID would be composed of four, 32-bit integers which fit into (can be EQUIVALENCED to) a REAL*16 word, the latter to facilitate sorting and searching,

- o In the use of the ID, an order in which operations (e.g., smoothing, truncation, interpolation) are performed would be imposed. This restriction would not dictate the position of certain indicators within the ID, but would contribute to which indicators were needed (i.e., no indicators were needed to indicate order of processing).

- o If necessary, some capability would be sacrificed in order to meet the four, 32-bit word limit. Tradeoffs would be made between defining a new variable by a unique CCCFFF (and thereby having to have at least a somewhat unique subroutine to handle it) and using a more generic CCCFFF and putting more processing information into the ID itself (which might enable the use of a more generic subroutine).

5. DATA RECORD STRUCTURE--TDLPACK

A. Introduction

Nearly all data (except for control files and other small ASCII files) accessed by MOS-2000 programs are in TDLPACK format. This is true whether the data are gridpoint or vector, predictor or predictand, or are in sequential or direct access files. TDLPACK is based on the World Meteorological Organization (WMO) GRIdded Binary (GRIB) code, but has been tailored to Techniques Development Laboratory (TDL) needs for data.[1]

Since the TDLPACK format was adapted from GRIB, it is especially relevant to gridpoint data. Station data have the same general record structure for data, but because the location of each datum in the record is not specified within the record, a "directory" record must be present. For the exact file structure of these vector data, see Chapter 6 "Sequential Data Archive Files."

There are six possible "sections" in each record. Each section plays the same role as the equivalent section in GRIB, although the exact contents of the sections are different. Five of these are described below; the bit map section is not currently implemented. All packed values are (scaled) integers; no floating point numbers are used. In all cases, when the value can be negative, the minus sign is indicated by a bit in the leftmost position of the "word."

When packing or unpacking records, values in four arrays are either furnished to the routines or are returned (the routine writeups are more explicit). The contents of ID arrays IS0( ), IS1( ), IS2( ), and IS4( ) are indicated below. The information indicated is always furnished to the user by the unpacking software. Most information must be furnished to the packing software; however, an asterisk following an element of the array in the definitions of the sections below means that the user need not furnish this value to the packer in that array element (e.g., IS1(1)). When the data are vector, rather than gridpoint, as indicated by a zero in bit 8, byte 2 of the Product Definition Section 1 in IS1(2), IS2( ) is not used.

The primary purpose here is to document the record structure, although some comments are also made which pertain to the MOS-2000 software that will use these records.

B. General Features

Some of the features embodied in, or supported by, this record structure are:

- o The record can be easily identified as a TDL-packed record [see IS0(1)].

---

[1]The reader is referred to WMO (1988) for a description of the GRIB code, to Stackpole (1993) for NMC's implementation of it, and to Glahn (1992, 1993, 1994, 1995, 1997, 1998) for studies pertaining to its efficiency and lack thereof.

- The date/time in YYYYMMDDHH form, as used in most MOS-2000 programs, and the TDL 4-word ID can be accessed after reading the record without unpacking the record; they are contained in words IS1(8) through IS1(12), which (after the record length word) are (4-byte) words 5 through 9 of the "packed" record. This makes searching more efficient than if the record had to be unpacked to determine whether or not it was wanted. Adjustment for 64-bit integers is necessary on the CRAY.

- Although the bit map (Section 3) is not presently implemented, it can be added later if necessary. This is unlikely, however, because missing values are already dealt with in another manner.

- This single format for gridpoint and point data [see word IS4(2)] makes for easy use in MOS-2000, although the point data option will not accommodate all the flexibility of the WMO Binary Universal Form for the Representation of Meteorological Data (BUFR) code.

- A missing datum is handled by the insertion of a specific value [see words IS4(2), IS4(4), and IS4(5)] in the place of the missing datum.

- A "primary" missing value and a "secondary" missing value are provided for in both vector (1-dimensional) and gridpoint (2-dimensional) data. The primary would normally correspond to 9999. The secondary is to account for the 9997 value produced by some regression equations and exists as such on the MOSARC archive. In addition, the secondary value can be a legitimate value that is, or may be, very frequent in the data. For instance, the 888 that indicates unlimited height in cloud layer height fields can be used as a secondary "missing" value. The software is such that a secondary is never indicated unless a primary is indicated, although a primary can be indicated without a secondary. Although these primary and secondary missing values are furnished to the packer, and returned from the unpacker, they must, of course, have values that will not occur as legitimate data values. Also, the primary and secondary missing values are hardwired in certain calling programs as 9999 and 9997 (or, say, 888), respectively, so use of these values is essentially mandatory; the packer and unpacker, themselves, are oblivious to the values used.

- Specifying latitude and longitude of the gird in degrees * 10,000 (tenths of millidegrees) gives an accuracy of about 12 m = .012 km.

- Specifying grid length in millimeters (km * 10^6) guarantees about the same location accuracy for a 5000-point grid (e.g., 1 km grid covering the U.S.) as specifying latitude/longitude in degrees * 10,000 (maximum error/grid unit = $.5 \times 10^{-6}$ km $\times 5 \times 10^3 = 2.5 \times 10^{-3}$ km = 2.5 m). Conversion of these numbers to REAL and standard units (e.g., m or km for gridlength and degrees for latitude/longitude should provide suitable accuracy. A gridlength of 4.123456 km is packed as INTEGER 4123456. REAL will retain that accuracy in a 32-bit word. For a larger number such as 380.123456, only about 7.2 decimal places of accuracy may be retained, but the number of grid intervals is much less and the accuracy should be sufficient. That is, more places of accuracy are necessary for small values than for large values.

- o The gridpoint data are completely defined (except that the identification of the field, together with its scaling factor, implies the units of the variable).

- o Only complex packing (removal of local minima as well as overall minimum) is supported [see word IS4(2)]. With the other aspects of the packing implemented here (e.g., a primary missing value indicated by all bits being 1 in a value), complex packing will almost always give better results than simple packing.

- o Packed values can be either the original values or 2nd order differences [see word IS4(2)], the latter being more efficient in the use of space in most instances of gridpoint fields.

- o Data values are packed in the order provided. MOS-2000 gridpoint data are packed boustrophedonically; this makes maximum use of the spatial redundancy of meteorological fields.

- o Both decimal and binary scaling is provided, the latter primarily to be able to essentially duplicate the fields packed by NCEP. Note, however, that the exact usage of these factors does not match the GRIB definitions (e.g., the binary scale factor has the sign reversed).

- o Since a value to be packed (a <u>scaled</u> integer value reduced by an overall minimum and a local group minimum) could be the same as the missing value indicator, the missing value is actually put into the TDLPACK record (as an integer) multiplied by 10000 to minimize the risk. The user need not worry about this; the software essentially eliminates the risk by reducing the actual packed values by 1 if a match occurs. This possible, but very rare, occurrence would compromise the data by 1 scaled unit.

C. Specifics of the Packed Data

The following six sections (Sections 0 through 5) correspond to the GRIB sections, although the contents have been tailored to TDL needs. Note that these sections (which comprise the "record" in GRIB parlance) are <u>preceded</u> by 8 bytes containing the "record length" in bytes. This record length is always evenly divisible by 8 to accommodate full words on a 64-bit machine. This way, reading can be by number of words with a normal FORTRAN read. This number of bytes, then, may be greater than contained in byte Nos. 5-7 in Section 0 below. [Also, this number of bytes does not include the 8 bytes in which the record length number is contained in sequential files (see Chapter 6).] In these sections, the variables IS0( ), IS1( ), etc. refer to "words" on whichever machine is being used--32-bit or 64-bit.

Indicator Section (Section 0)--This section is always 8 bytes long and serves only to identify this as TDL packed data and specify the total length of Sections 0 through 5.

Byte Nos.	ID Word IS0( )	Description
1-4	(1)*	The characters "TDLP" which designate this as TDL Packed data.
5-7	(2)*	Record length in bytes. This includes the 4 bytes in Section 5 but not any padding necessary to make the record length evenly divisible by 8 nor the leading "record length" of 8 bytes (see above).
8	(3)*	TDLP edition number. The first edition is 0.

Product Definition Section (Section 1)--This section identifies the data, the model (source) date/time, projection, variable, etc.

Byte Nos.	ID Word IS1( )	Description
1	(1)*	Section length in bytes. The * means this value is filled in by the software and does not have to be furnished by the user when packing. All other values must be user furnished when packing. All values are furnished to the user when unpacking.
2	(2)	Flags relative to presence of the Grid Definition and Bit Map Sections. (Bit numbers are counted from the left, 1-8.) Bit 7--0 = Bit Map Section not included. 1 = Bit Map Section included. While not currently supported, this allows for inclusion at a later time if necessary. Bit 8--0 = Grid Definition Section not included. This also implies the data are not gridpoint, but vector instead. 1 = Grid Definition Section included.
3-4	(3)	Year (e.g., 1994).
5	(4)	Month number (e.g., 1 for January).
6	(5)	Day number in the month.
7	(6)	Hour (e.g., 12 for 1200 UTC). All times will be UTC.
8	(7)	Minutes within hour (e.g., 30 for 1230 UTC).
9-12	(8)	Date/time in format YYYYMMDDHH. Matches IS1(3-6).
13-16	(9)	First Word of ID (CCCFFFBDD).
17-20	(10)	Second Word of ID (VLLLLUUUU).
21-24	(11)	Third Word of ID (TRROHH$\tau\tau\tau$).
25-28	(12)	Fourth Word of ID (binary cutoff plus ISG).
29-30	(13)	Projection (tau) in hours. Must not be negative. This is redundant with the $\tau\tau\tau$ in the third word of the ID.
31	(14)	Projection (tau) in minutes, used in conjunction with 29-30 (i.e., add minutes to hours to get total projection).

32	(15)	Model or Process number. Redundant to DD in the first ID word.
33	(16)	Sequence number within Model number. This can be used for multiple runs of the same model (e.g., ensemble forecasts).
34	(17)	Decimal scale factor, D (e.g., a 1 means that the original values have been multiplied by 10^1). See binary scale factor below.
35	(18)	Binary scale factor, E, (e.g., a 1 means that the original values have been multiplied by 2^1). The scaled value = original value*10^D*2^E. This combination of binary and decimal scale factor applies to all data values, the overall minimum, the group minima, the 2nd order differences, etc. Note that while the scale factors have similar definitions as the WMO GRIB, they are used slightly differently in MOS-2000, and especially the sign of E is reversed.
36-38	(19-21)	Reserved. Set to zero.
39	(22)	The number of bytes, LPL, of plain language (ASCII) variable description. Generally limited in MOS programs to 32.
40 to LPL+39	(23-xx)	LPL bytes of plain language description, one byte per word. Maximum length of IS1( ) = 22 + 32 = 54. Maximum length of section = 39 + 32 = 71 bytes.

Grid Definition Section (Section 2)--This section identifies the grid when the data packed are gridpoint. It's presence is indicated by a bit in the last (rightmost) position of byte 2 in the Product Definition Section. It is 28 bytes long.

Byte Nos.	ID Word IS2( )	Description
1	(1)*	Section length in bytes. The * means this value is filled in by the software and does not have to be furnished by the user. All other values must be user furnished when packing. All values are furnished to the user when unpacking.
2	(2)	Map projection to which the grid is relative (e.g., 3 for Northern Hemispheric Lambert, 5 for north polar stereographic; a different number would be used for Southern Hemispheric projections). This can also indicate the type of grid (e.g., latitude/longitude). If spectral coefficients were to be in the record, this could be indicated here. Because the section length is always present, the exact definition of the following bytes can be changed depending on the value in this byte.
3-4	(3)	NX, the number of gridpoints in the IX direction (left to right, usually west to east).
5-6	(4)	NY, the number of gridpoints in the JY direction (bottom to top, usually south to north).

7-9	(5)	Latitude of the lower left corner gridpoint in degrees*10000 (tenths of millidegrees). Range is 0-90 degrees in the northern hemisphere, and 0 to -90 degrees in the southern hemisphere. Note that negative values are supported.
10-12	(6)	Longitude of the lower left corner gridpoint in degrees*10000 (tenths of millidegrees). Normally, the range is 0 to 359.9999 degrees, where west longitude is 0 through 180 and east longitude is 180 through 359.9999. However, negative values are supported.
13-15	(7)	Grid orientation (i.e., the vertical longitude) in degrees*10000 (tenths of millidegrees). Normally, the range is 0 to 359.9999 degrees, where west longitude is 0 through 180 and east longitude is 180 through 359.9999. However, negative values are supported.
16-19	(8)	Grid length in millimeters (i.e., km*1000000). This accommodates a gridlength of over 2000 km.
20-22	(9)	Latitude at which the grid length applies in degrees*10000 (tenths of millidegrees). Range is 0-90 degrees in the northern hemisphere, and 0 to -90 degrees in the southern hemisphere. This is also the latitude of tangency in case of the Lambert map projection. Note that negative values are supported.
23-28	(10-12)	Reserved. Set to zero.

<u>Bit Map Section (Section 3)</u>--This section is not currently supported. Missing data are handled in a different manner, making a bit map unnecessary. It can be included later, if necessary [see IS1(2)].

<u>Data Section (Section 4)</u>--This section contains the data. The data will be complex packed (i.e., includes 2nd order minimum removal). However, the possibility of simple packed data is provided (see IS4(2), bit 5); the software and extended definitions can be implemented, if needed. The definitions in this section pertain to complex packed. The number of groups can be small, or even degenerate to 1, when station data are being packed and the order of the data do not provide the redundancy present in gridpoint data. In the case of gridpoint data, the NX*NY points are consecutive starting at the lower left of the grid, proceeding left to right, then hopping up to the next row and proceeding right to left; the scan direction is reversed on alternate rows.

Byte Nos.	ID Word IS4( )	Description
1-3	(1)*	Section length in bytes. The * means this value is filled in by the software and does not have to be furnished by the user. All other values are user furnished. (IS4(2) comes into subroutine PACK from PACK1D or PACK2D; all other values are filled by PACK)
4	(2)*	Packing flags. Bit numbers are counted left to right. Others can be added if needed. Bit 4--0 = Gridpoint data. 1 = Not gridpoint data. Bit 5--0 = Simple packing. While not currently supported, this allows for inclusion at a later time if necessary. 1 = Complex packing. Bit 6--0 = Original scaled data packed. 1 = 2nd order (spatial) differences of original scaled data packed. Bit 7--0 = No primary missing values. 1 = Primary missing values may be present. Bit 8--0 = No secondary missing values. 1 = Secondary missing values may be pres- ent. This should not be 1 unless Bit 7 is also 1.
5-8	(3)*	Number of values packed. When these are gridpoint data, this value will be NX*NY. It will be this value even though 2nd order differences are packed. This value is furnished to the packer as a separate argument.
9-12	(4)*	Primary missing value indicator; this is present <u>only</u> when bit 7 in IS4(2) is a 1. This value is furnished to the packer as a separate argument. Normally 9999.
13-16	(5)*	Secondary missing value indicator; this is present <u>only</u> when bit 8 in IS4(2) is a 1. This value is furnished to the packer as a separate argument. Normally 9997.

In the rest of the record, the bits indicated are sequential; they are <u>not</u> byte-oriented. When packing 2nd order differences (IS4(2) has a 1 in the third bit from the right), 2 values are needed that <u>are not used</u> when packing original scaled values. These values are the first value in the original field and the first first-order difference. <u>They are indicated in the following section marked by a row of *'s.</u>

****

Bits Required	ID Word	Description
32		The first value in the original field. A negative is indicated by a 1 in the first (leftmost) position.
5		MBIT, the number of bits required for the absolute value of the first first-order difference. That is, MBIT is contained in 5 bits, and as such, $2^5 - 1 = 31$ is the largest number of bits that can be used by the absolute value of the first first-order difference.
1		The sign of the first first-order difference (1 = negative).
MBIT		The absolute value of the first first-order difference.

****

Bits Required	ID Word	Description
5		NBIT, the number of bits required for the absolute value of the overall minimum value. That is, NBIT is contained in 5 bits, and as such, $2^5 - 1 = 31$ is the largest number of bits that can be used by the absolute value of the overall minimum.
1		The sign of the overall minimum (1 = negative).
NBIT	(6)*	The overall minimum value. As with all other values in the remainder of the record, this can pertain to the original values or to the 2nd order differences.
16	(7)*	The number of groups, LX.
5		IBIT, the number of bits required for the group minimum values. That is, IBIT is contained in 5 bits, and as such, $2^5 - 1 = 31$ is the largest number of bits that can be used by the group minima.
5		JBIT, the number of bits required to hold the number of bits required for each group's values. That is, JBIT is contained in 5 bits, and as such, $2^5 - 1 = 31$ is the largest number of bits that can be used to indicate the number of bits required for each group.
5		KBIT, the number of bits required for the number of values in each group. That is, KBIT is contained in 5 bits, and as such, $2^5 - 1 = 31$ is the largest number of bits that can be used to hold the number of values in each group.
IBIT*LX		JMIN(L), the group minimum value for group L of the LX groups (L=1,LX).
JBIT*LX		LBIT(L), the number of bits required for each value in group L (L=1,LX).
KBIT*LX		NOV(L), the number of values in the group for each group L (L=1,LX).
LBIT(L)*NOV(L)		The data values after the group minimum removal for each group. Note that a total of NX*NY values must be represented. Empty groups (LBIT( )=0) are possible. When, and only when, a primary missing

value can be present, as indicated by bit 7 = 1 in IS4(2), the maximum value (within the width provided by LBIT( ) for the group values) is reserved to indicate that primary missing value.[2] Also, when a secondary missing value can be present, as indicated by bit 8 = 1 in IS4(2), the maximum value minus 1 is reserved to indicate that secondary missing value. In the case when all values in a group have the primary missing value, that primary value becomes the minimum value, and no actual "data" values are needed. That is, all groups (which can be as few as one) are empty.

This section is padded to a full byte with zeros.

End Section (Section 5)--This section contains a terminator.

Byte Nos.	Description
1-4	Character representation "7777".

The following diagram indicates the overall structure.

No. Bytes	Contents
8	Section 0
39+	Section 1
	Minimum of 39, maximum of 71 bytes
28	Section 2
0	Section 3 (omitted)
8	Section 4
4	Primary missing indicator (only) if present
4	Secondary missing indicator (only) if present
Variable	Present only for 2nd order (spatial) differences
Variable	Data
Variable	Pad to an even byte, if necessary
4	Section 5

[2]This method will nearly always be more efficient in bits than use of a bit map and will never be worse. Only when the maximum value in a group is an exact power of 2 minus 1 will an extra bit be required to accommodate a missing value. (Note that the minimum is always 0.) When the maximum value is an exact power of 2 minus 1, an extra bit will be required for each value in the group to accommodate a possible missing value. For a bit map, one bit is required for every value (for all groups) to accommodate a possible missing value.

D. Software Supporting the Record Structure

Packing

PACK-- The basic packing subroutine is PACK, which calls PKMS00, PKMS99, PKC4LX, PKS4LS, and PKBG.
PKMS00-- Called when there are no missing values indicated. It calls PACKGP.
PKMS99-- Called when missing values can be present. It calls PKMS77, when secondary missing values may be present, and PACKGP.
PACKGP-- Determines the groups for complex packing.
PKBG-- Stuffs bits into 4-byte or 8-byte words in a specified place.

Since the information furnished to PACK is somewhat different depending on whether the data are gridpoint, two preparatory routines are used.

PACK1D-- Called when data are not gridpoint. Basically, it just scales the data and initializes IS4(2) and calls PACK.
PACK2D-- Called when data are gridpoint. It determines, by calling PACKXX or PACKYY (depending, respectively, on whether there can be missing values), whether or not to use 2nd order spatial differences, puts data into boustrophedonic order, and initializes IS4(2).
PACKXX-- Determines whether or not to pack 2nd order spatial differences, without undue computation.

Unpacking

The unpacking is done by calling UNPACK.

UNPACK-- Does all the unpacking. It calls UNPKBG, UNPKLX, UNPKOO, UNPKPO, and UNPKPS.
UNPKLX-- Incorporates UNPKBG capability into the loop for efficiency.
UNPKOO-- Incorporates UNPKBG capability into the loop for efficiency. Used when there are no missing values.
UNPKPO-- Incorporates UNPKBG capability into the loop for efficiency. Used when there can be primary missing values.
UNPKPS-- Incorporates UNPKBG capability into the loop for efficiency. Used when there can be primary and secondary missing values.
UNPKBG-- Unstuffs a specified number of bits from a specific place in 4-byte words.

6. SEQUENTIAL DATA ARCHIVE FILES

A. Introduction

The majority of archived data will be stored in sequential files in TDLPACK format (see Chapter 5). The "record length" in 8 bytes is the first "word" in the record, and this value can be used with FORTRAN reading routines to read the entire record. (On a 32-bit machine, this is two "words.") Note that this value does not include the 8-byte value itself; it is the number of bytes to read following the "record length." Internal buffering to keep the records to constant size or to reduce the number of physical records is not used.

The MOS-2000 archive, and other data files as well, are structured so that they can be read and used on either a 32-bit or a 64-bit machine. Modern workstations and their FORTRAN compilers usually employ 32-bit words; however, the CRAY super computers have a 64-bit word length. The CRAY has an option to use 64-bit integers; this archive has been designed to take advantage of that feature. Frequent mention in this document is made of "words." This usually means a 4-byte (32-bit) word. The reader may have to interpret this in terms of an 8-byte word for the CRAYs.

The data in any particular archive (e.g., forecasts from the aviation model, or hourly data) are arranged by date/time. That is, all data for a particular date/time are together as a group of records and are on the file before any subsequent date/time. An order of variables within the date/time block is not specified, nor necessarily maintained as constant over the period of archive.

B. Gridpoint Data

Format

The format is TDLPACK as described in Chapter 5. The grid is fully defined within the data record, except for the data units; archiving software ensures that the archive is constant over time in that respect. Also, certain other processing may be necessary or desirable to make the data useful without extreme complexity, leading to possible errors that would be hard to detect. For instance, if the accumulated precipitation amount output from a particular model is at 2-hourly intervals, and it is desired to be able to use 6-hourly amounts as predictors, it may be better to make the required sums while doing the archiving. This is especially true when the time increments are not constant over projection, model, and date.

The Product Definition Section of TDLPACK includes the MOS-2000 ID's as described in Chapter 4.

Preparation and Use

Gridpoint data may be produced by a number of programs, including those that reformat extensive existing archives, those that archive daily the current output from NCEP models, and those that prepare "operational" data from which to produce real-time guidance forecasts. Whatever the driving program, the subroutine PACK2D and its called subroutines PACKXX, PACK, PKMS00, PKMS99, PACKGP, and PKBG are used for the packing. Once packed, the data are written by the subroutine WRITEP.

Gridpoint data are unpacked by subroutine UNPACK and its called subroutine UNPKBG. The data definition sections can then be used to determine the characteristics of the data. Because a particular program may need only a small portion of the data in the file, rather than unpack everything and use the resulting definition to determine whether or not to keep the data, three "prescreening" methods can be employed. First, the date/time can be rather easily checked without unpacking even all the identification sections, as is done in RDSTR1 and PRED2 for U201. For a 32-bit machine, after the data have been read into IPACK( ), the date/time, IDATE, can be found by:

```
IDATE=IPACK(5)
```

For a 64-bit machine, IDATE is found by calling UNPKBG thusly:

```
LOC=3
IPOS=1
CALL UNPKBG(KFILDO,IPACK,ND5,LOC,IPOS,IDATE,32,L3264B,IER,*900)
```

where KFILDO is the unit number for printed output, ND5 is the dimension of IPACK( ), L3264B = 64, and 900 is the statement to return to when there is an error IER ≠ 0.

Secondly, the 4-word ID can be determined for a 32-bit machine by:

```
ID(1)=IPACK(6)
ID(2)=IPACK(7)
ID(3)=IPACK(8)
ID(4)=IPACK(9)
```

and for a 64-bit machine by:

```
LOC=3
IPOS=33
CALL UNPKBG(KFILDO,IPACK,ND5,LOC,IPOS,ID(1),32,L3264B,IER,*900)
CALL UNPKBG(KFILDO,IPACK,ND5,LOC,IPOS,ID(2),32,L3264B,IER,*900)
CALL UNPKBG(KFILDO,IPACK,ND5,LOC,IPOS,ID(3),32,L3264B,IER,*900)
CALL UNPKBG(KFILDO,IPACK,ND5,LOC,IPOS,ID(4),32,L3264B,IER,*900)
```

Thirdly, subroutine UNPACK has the option of unpacking and returning only the identification sections without unpacking the gridpoint data themselves. (If many of the data records are needed, this latter facility may not be advantageous, because when it is determined that the data are needed, the whole record, including the identification, must be unpacked. That is, in this scenario, the identification sections are unpacked twice. This is a safety feature designed to never have identification sections that don't match the unpacked data. However, the processing associated with unpacking the identification sections is small compared to unpacking a grid, unless the grid is small.)

Existing Archived Gridpoint Data

These data are described in Chapter 12.

C. Vector Data

Format

The format is TDLPACK as described in Chapter 5. Since the data are vector, and the geographic location of each datum in the record is not defined within the packed record as gridpoint values are, there must be a "directory" record or records. Therefore, the first record on each vector file consists of station call letters of 8 characters each. If the data do not apply to stations, as particular gridpoint values would not, call letters must be "manufactured." For gridpoint values, these are composed of 4 characters defining the IX location and four defining the JY location. Note that these values are in reference to a particular grid. Since the values are exact (at gridpoints), 4 characters will accommodate a 9999 by 9999 grid.

It is necessary for some purposes to include in a file multiple directory records, each one applying to the data following it and before the next directory. This is especially true for archived hourly data, where the station list is not constant from hour to hour. The file structure for vector data is as follows:

- o A directory record consisting of, as for gridpoint data, 8 bytes which specify the bytes in the record following these 8 bytes. Following these 8 bytes are the station call letters of 8 bytes each. The number of stations is this "record length" divided by 8. This record is not packed, as packing would serve little purpose and would just complicate the programming. Most files will have only one, or at most a few, directory records, except hourly data archives.

- o One or more records in TDLPACK format (with the usual 8 leading bytes for record length), each value associated with the station (or location) in the corresponding position in the directory record.

- o A "trailer" record. A trailer record consists of the usual 8 bytes for record length, plus 24 (or more) bytes. These 24 bytes are sufficient to contain the date/time in a normal packed record. (24 bytes are used rather than 20, which are sufficient to contain the date/time, to make even 64-bit words.) The other bytes in the trailer can be zeros. When the date/time = 9999, this is a trailer record indicating that the next record is another directory record or an end of file (EOF). Note that a trailer record will normally exist after all data, even before the EOF, and the switching of files by certain software depends on this.

- o A repeat of the above three segments, until an EOF is reached.

A directory record can be written by subroutine SKIPWR, which has the capability to:

- o When desired, skip past a user-specified date/time.

- o When desired, check an existing directory record in the file with one provided to make sure they match before continuing.

- o When desired, writing in the correct format the directory record.

The Product Definition Section of TDLPACK includes the MOS-2000 ID's as described in Chapter 4.

Preparation and Use

Vector data can be produced by a variety of programs. Generally, predictor and predictand data are prepared by U201, and (only) for efficiency in following programs which employ a lookahead feature, predictors and predictands should be on separate files. Whatever the driving program, the subroutine PACK1D and its called subroutines PACK, PKMS00, PKMS99, and PKBG are used for the packing. Once packed, the data are written by the subroutine WRITEP.

Vector data are unpacked by subroutine UNPACK and its called subroutine UNPKBG. The data definition sections can then be used to determine the characteristics of the data. Because a particular program may need only a small portion of the data in the file, rather than unpack everything and use the resulting definition to determine whether or not to keep the data, three "prescreening" methods can be employed. See the section on preparation and use of gridpoint data above for how these methods can be employed; examples are in RDSTRX and RDVECT.

Existing Archived Vector Data

These data are described in Chapter 13.

7. EXTERNAL RANDOM ACCESS FILE SYSTEM

A. Introduction

This file system is used in MOS-2000 to store certain data that may be frequently needed by a variety of developmental and operational programs. While the predominant source of data for MOS-2000 programs is sequential files with data packed in TDLPACK format (either gridpoint and/or vector, depending on the program), "constant" data (e.g., daily max temperatures or precipitation relative frequencies) pertaining to specific locations (normally station locations) may be needed more than once in a particular program, and the total volume of this kind of data is large enough to preclude all data being resident in memory. In addition, there may be gridpoint data of a "constant" nature (e.g., terrain height) that are needed several times in a program.

This file system is also used to store forecasts from the operational U900 and U910 programs, because various follow-on programs (including certain NCEP programs) need to access these forecasts in a random manner. Forecasts from different cycles of the same model can't be on the same file because the key record does not contain a date/time. (It is actually possible to put them there, but the IDs from the different cycles would be the same and finding what is wanted would not be very practical.) The data are identified by date/time <u>within</u> each record.

The data in this system are stored in TDLPACK format, except for the "directory" record, which contains the definition of the locations of the data in each vector record. The word "external" is used in the definition to preclude it being confused with the "internal" random access storage system used in many MOS-2000 programs (see Chapter 8). These can be "vector" type random access files or gridpoint random access files; vector (station oriented) and gridpoint data will not reside on the same file. The only difference in the vector and gridpoint files is that the former has a directory record, while the latter does not.

This file system can be used on either a 32- or 64-bit computer, but a particular file should not be copied from one word-length computer to another.

This file system was adapted from the TDL File System that has been in use by LAMP for several years on three different platforms--AFOS, VAX, and HP workstations. Because of specific needs of LAMP and because of the need to be especially efficient in memory on AFOS, three different types of files were accommodated--gridpoint oriented, vector oriented, and station oriented, the latter pertaining to synoptic data of different types in one record. The system was simplified in various ways, so that now only vector and gridpoint types are accommodated.

B. File Structure

There are three types of records in each file--a master key record, one or more data key records (usually called just key records), and data records. These are described here. In the descriptions, a "record" is a logical record and is usually so designated. Each logical key or data record may be composed of more than one physical record.

Master Key Record

The first record in each file is a master key record. It always contains six words. This information must always be present and is read whenever the file is opened. It is written by the file system program U350 which creates the file and is modified by the file software as data are added to the file. The six words are defined as:

Word No.	Program Name	Description
1	(Reserved)	Not used at the present. Can be used for a file format version number if it becomes necessary.
2	NIDS	Number of words in each ID. This is constant at 4, conforming to the 4-word MOS ID system. The software would have to be modified slightly to handle any other value.
3	NWORDS	Number of words in a physical record. This will be either 32- or 64-bit words, depending on the machine being used (how L3264B is set).
4	NKYREC	Number of key records in this file. As explained below, there is always one--the second record in the file--and there may be more if the first one becomes full.
5	MAXENT	Maximum number of key entries in a key record of this file.
6	LASTKY	Location (record number) of the last key record in the file. Initially, this is 2 when there is only one key record.

When this master key record is read and stored for each open file, there is a 7th word appended in memory. It designates whether (1) or not (0) this key record need to be written when closing the file.

Data Key Records

The second logical record in each physical file is a data key record. It contains in order:

(1) The actual number of keys in this key record.

(2) The number of physical records it takes to contain this logical key record (the logical key record can be split over several physical records).

(3) The location (physical record number in the file) of the next key record in the file (9999 is used in the last key record).

Following these three words are up to MAXENT keys, MAXENT being set when the file is created. The actual number of keys in this key record is indicated by word (1) above. The logical record size is constant whether or not all MAXENT keys are present. Each key contains in order six words:

(1-4) The four MOS ID's.

(5) The number of data values in the record corresponding to this key (the physical records are of constant size, but the logical record can be split over several physical records).

(6) The beginning physical record number of the logical data record corresponding to this key*1000 + the number of physical records it takes to make up this logical data record.

Note that a date/time is not in the key.

Following the logical key record, the first one being record No. 2 in the file, are the data records corresponding to the keys. Once MAXENT logical data records are written and the corresponding key record (logical record No. 2) is filled, another logical key record is written, etc. There is no practical limit to the number of records in the file. Note, again, that both key and data records can be split over several physical records.

Data Records

Each data record, except the directory record, is packed in the TDLPACK format, so further description of the data can be present in the identification sections of the packed data. For instance, a date can be there if relevant. For vector oriented files, the first data record is the "directory," or identification of the location for each datum in each data record. Normally, this consists of station call letters, each being 8 characters. If the data are inherently at gridpoints, a location can be composed of the IX and JY locations in the grid. This directory record is not packed in TDLPACK.

C. File System Software Overview

A group of subroutines (see Section D) control access to the data in the file system, only three of which the user need to use directly (besides creating the file, which will be explained later). One routine, WRTDLM, is for writing; another, RDTDLM, is for reading; and CLFILM is for closing a file. Note that the user need not be concerned about opening the file.

Considerable information about the open files and the data therein must be maintained in memory. In order to not burden the user with passing around all the variables involved, a COMMON block, ARGC, is used to communicate among the subroutines. Several of the arrays are 2- or 3-dimensional; it is possible to reduce the dimensionality by one in most cases when using lower level routines. The dimensions of the variables in ARGC which it might be necessary to change are set by PARAMETER statements in each of the five routines (WRTDLM, RDTDLM, FLOPNM, CLFILM, and TDLPRM) in which this block exists. The variables set by PARAMETER are:

MAXOPN The maximum number of files that can be open at any one time. Since one key record of each open file is resident in memory, the maximum number that can be open is important in controlling the amount of storage allocated for this purpose. It is likely that for most purposes 2 is sufficient. Note that this does not limit the number of files that can be used in one run; if a file needs to be closed to make room for another to be opened, the software will do that automatically. If a random access file were to be used for writing MOS forecasts from each of multiple model runs (ensembles), then it might be desirable (but not mandatory) to increase MAXOPN to the number of model runs being used plus the number of inputs. This would necessitate, then, the five routines in which the PARAMETER statements exist to be loaded specifically into the using program. That is, the "library" version of those subroutines that would be used for almost all other purposes, would not be used.

NW The maximum number of entries in any key record being used. Note that this is a maximum; the actual maximum number of keys in each key record is established for the file when the file is created and the master key record written. It is likely that each file will be created using the value of MAXENT input to U350 that is decided on for general MOS-2000 use. This value also has considerable effect on the amount of storage allocated to the random access file bookkeeping. The variable KEYREC is dimensioned (6,NW,MAXOPN). It is likely a value of 300 is about right.[1] Too small a value may cause key records to be read and written excessively; too large a value will cause some wasted space, but not a lot provided MAXOPN is kept to a small number. Since the actual number of entries is kept as part of the key record, searching only occurs over the filled portion.

MAXFIL This is the dimension of only two variables and has little effect on storage or efficiency. It is the maximum number of random access files for which physical record size information is kept and used in case a file has to be closed and reopened. (More about this later.) A value of 20 is good enough and should not have to be changed.

Assuming a file has been established and some data put into it, a process which will be explained later, the user will want to read or write to the file. All reading is done with the routine RDTDLM. RDTDLM arranges for the file to be opened as necessary, the master key record read, and the first data key record read and searched for the data wanted. If the data wanted, determined by the 4-word MOS ID, is found, the appropriate data record is read, and the data returned in the array provided. The user must unpack the data and using the "directory" record (for vector data), which is read by a call to RDTDLM also, pick out the specific values needed.

When a file is first opened, it is not known what the physical record size is. It is opened to the number of bytes necessary to contain the 6-word master key record (24 for the 32-bit HP Workstations). The master key record is read, the file is closed, and then reopened to the correct size obtained from word No. 3 (the value in words being converted to bytes). This value is stored, and if the file is closed and reopened, the correct record size will be known (unless the PARAMETER MAXFIL has been exceeded).

[1]Note that NW is also used in U350 when a file is created. This value need not be the same as NW in the routines using the file and can be a much larger number to accommodate whatever size the user of U350 needs.

When a file is opened from RDTDLM, it is assumed it will be used for reading only and is opened that way. Therefore, file usage can be shared among programs. The first key record is read first for searching. When a file is opened from WRTDLM, it is known that writing will be required and is opened for both reading and writing. The key record read will be the last one in the file, its location being known from word No. 6 in the master key record, in anticipation of writing and adding a record to that key record. Any other accesses from RDTDLM will search the key record then existing in memory, then if the key cannot be found, the search will start with the first key record and proceed through the key records in sequence, except that the one already searched will not be searched again. Any other access from WRTDLM will cause the last key record to be read immediately if it does not exist in memory, unless replacement or key checking is being done, in which case it may be necessary to proceed through all key records. In many cases, there may be only one key record necessary, depending on the value of MAXENT used.

Since the file system does not write a key record unless necessary (another key record from the same file is read, or the file is closed), after the last write access to a file, the user must close the file by calling CLFILM.

By providing for multiple key records, there is no practical limit to the number of records the file can contain (the absolute limit is $(2^{31}-1)/1000 =$ 2,147,483). There is also no practical restriction on the physical record size, except the number of physical records per logical record is $\leq$ 999. However, each logical record starts on a new physical record; the physical record corresponding to the FORTRAN "record." This means that there can be some wasted space, but the volume of data required to be in these random access files is relatively small, so this is of no real concern. However, one would not want to assign a physical record size several times the expected size of a logical record. The software imposes no limit on the number of files that can be used on one run.

Records can be written with or without replacement. When replacement will occupy more physical records than the existing one occupies, the existing key is "zeroed out" and a new record added. This will fragment the file. However, because the intended use is for data that are relatively stable, an occasional fragmentation should not be a problem. Note that the new record does not have to have $\leq$ the number of bytes of the old record for actual replacement to occur, but only to require $\leq$ the same number of physical records.

When writing without replacement, a check can be made for a duplicate. If one is found, an error is indicated and the record is not written. Duplicate checking does, of course, take some time, but is probably wise.

D. Creating a Random Access File

Each random access file must be created with program U350 (see Section E). Input to U350 consists of the maximum number of entries in the data key record (MAXENT), the physical record size (NBYTES), and the file name. If NBYTES is so large that more than MAXENT entries could be accommodated in one physical record, MAXENT will be increased to match. See the writeup for U350 for more information.

E. Accessing Routines

A system of subroutines and programs manipulate this file system for the user. They are described briefly below. A detailed writeup is provided in the MOS-2000 documentation for the three--WRTDLM, RDTDLM, and CLFILM--that are called directly by the user; more limited information is furnished about the others. Also described are routines for creating a file and putting data into it.

MOS-2000 No.	Name	Purpose
3.0	U350	Creates a file, writes the master key record, and one (empty) data key record.
3.1	U351	Writes the call letters record, and optionally the latitude, longitude, WBAN number, and elevation of each station, these latter values being taken from the Station Directory file. It is a general template for writing data into a MOS-2000 External Random Access file.
3.3	U352	Performs the same functions as U351 and also reads sequential packed data, associates the values with the directory for the random access file, and writes those data.
3.4	U353	Copies records from one random access file to another, omitting records as desired. It can be used to inventory a file and to defrag it.
3.5	U354	Inventories a random access file.
3.6	U361	Writes a gridpoint record to a random access file. Similar to U351 for vector records.
3.2	PAWRA	Called by U351 and U352 to pack (with PACK1D) and write a vector record (with WRTDLM) to the file system.
3.7	PAWRAG	Called by U361 to pack (with PACK2D) and write a gridpoint record (with WRTDLM) to the file system.
4.55	WRTDLM	Writes a record <u>for the user</u> to the file system. It provides for the file to be open by calling FLOPNM and calls WRTM, reducing the dimensionality of most variables in the COMMON block by one before inserting them into the call sequence.
4.56	WRTM	Called by WRTDLM. Is the workhorse of the writing process.
4.57	WRKEYM	Writes data key records. Called by RDTM and CLFM.
4.58	WRDATM	Writes a data record. Called by WRTM.
4.59	RDTDLM	Reads a record <u>for the user</u> from the file system. It provides for the file to be open by calling FLOPNM and calls RDTM, reducing the dimensionality of most variables in the COMMON block by one before inserting them into the call sequence.
4.60	RDTM	Called by RDTDLM. Is the workhorse of the reading process.
4.61	RDKEYM	Reads data key records. Called by RDTM and FLOPNM.
4.62	RDDATM	Reads a data record. Called by RDTM.
4.63	FLOPNM	Arranges for an appropriate file to be open. All accesses to the file system go through FLOPNM, except for closing a file by the user. Called by WRTDLM and RDTDLM.
4.64	CLFILM	Closes a file <u>for the user</u> by calling CLFM, after writing the appropriate data key and master key records. CLFILM

		is also called by FLOPNM if it is necessary to close a file in order to open another one.
4.65	CLFM	Closes a file. Called by CLFILM and FLOPNM.
4.66	ARINIT	Initializes COMMON block ARGC for FLOPNM.
4.67	TDLPRM	A diagnostic routine that can be called from other routines, particularly FLOPNM, when compiled with the "D" option. It lets a user monitor the use of the file system. Considerable print may be generated.

In most MOS-2000 programs, the subroutine CONST performs the function of matching IDs with unit numbers, fetching the data from the appropriate random access file, unpacking the data, and providing to the calling program the values wanted in the correct order corresponding to the "station list" provided. The correspondence between the directory record on the file and the order in which the data are needed is determined once for the file and the resulting record stored in the MOS-2000 Internal Storage System. This correspondence is then reused, even if the file is closed and reopened, so that the directory record need be read only once. CONST is accessed from OPTION in U201 and from OPTX from other programs; it is not considered part of the MOS-2000 External File System, but since the file system was created specifically for furnishing constant data, it should be very useful in accessing the file system. For gridpoint data, the subroutine CONSTG can be accessed directly by the user or for U201 is accessed through CONST when the Unit No = 44.

The unit numbers for the random access files must be in the range 44 through 49, and those unit numbers are used for the following purposes related to values of CCC in ID(1):

Unit No.	CCC Range	Use (Read only except No. 49)
44	400-499	Gridpoint constants (e.g., terrain height).
45	400-499	"True" constants (rel. freq., means, etc.)
46	500-599	1-d and 2-d constants, probably for U201
47	800-899	Thresholds for best category forecasts, etc.
48	200-299	Forecasts read only
49	200-299	Forecasts read/write

F. Use for Operational Forecasts

Data in a particular random access file will pertain to some specific date/time or range of date/times. For instance, a daily max temperature would pertain to a particular date and the time would be an agreed-on value, a relative frequency might pertain to a full month, and station elevations would pertain to all date/times. The MOS-2000 4-word ID and the information stored in the TDLPACK record carries the necessary information. However, because the file system was designed to hold "constants" that remained constant from year to year, no date/time is carried in the data key records. This means that forecasts produced by U900 from one model run time should not be in the same file with forecasts from another run time of the same model; a different file has to be used else duplicates would occur in the data key records. This restriction is probably not major, since the forecasts are archived to sequential files and not stored permanently in random access form. The file names can differentiate the different runs.

8. INTERNAL STORAGE SYSTEM

Many of the MOS-2000 programs require that data be stored internally in a direct (or random) access manner. For instance, U201 may need a particular gridpoint field multiple times. Also, the amount of data required at one time may be greater than can reasonably be accommodated in memory. On the other hand, one would not like to make a disk access every time a field were needed unless necessary. In addition, some data may be required for a number of "iterations" within the program, and other data may be needed for only a short time. This leads to a combination of memory/disk storage system that can be selectively purged.

Usually, it is not known when a program starts how much data will need to be stored internally for either short or long periods. Since the primary input data are in sequential archive files, it is necessary to store all the data that may be needed for the first day (Day 1), to keep track of what is needed on Day 1, and on future days be able to just use those data without storing them internally if possible, or to be able to store those needed and to purge them as soon as possible. (Actually, Day 1 refers to the first date/time, and could be the first "cycle" for a day on which multiple cycles are needed.)

The MOS-2000 Internal Storage System has been designed for the MOS-2000 programs, and access routines GSTORE, GFETCH, LMSTR1, LMSTR2, LMSTR6, GCPAC, WRDISK, and RDDISK provided for its use. In addition, UNPACK is used. No other routines should be needed.

The storage and purging is mainly controlled by data in the array LSTORE(L,J), L=1,12; J=1,LITEMS, and later the array MSTORE(,). For each item J, the values stored by GSTORE or modified in GFETCH in LSTORE(,) for L are:

L = 1,4 - A 4-word ID for the data item J. Normally, this would be the MOS 4-word ID, but could be something different depending on the need. For instance, the linearization routines for U201 may store thresholds here with an appropriate definition. The index record(s) for the MOS-2000 External Random Access Files are stored here by subroutine CONST (see Chapter 7, Section E).

L = 5 - Location of the data item J. When the data are in memory, this is the location in the array CORE( ) where the data start. When the data are on disk, this is minus the record number where the data start. That is, when positive (negative), the data are in memory (on disk).

L = 6 - The number of 4-byte "words" of data stored. This can be either 64-bit or 32-bit data, depending on the machine being used.

L = 7 - 2 when the data are packed in TDLPACK format; 1 when not packed. When accessed by GFETCH, the data will be unpacked as necessary.

L = 8 - The date/time of the data in format YYYYMMDDHH. For data like climatological that have no specific time, this can be 0.

L = 9 - A count of the number of times the data have been retrieved by GFETCH. This is used in certain programs to determine whether these data are needed for another date/time.

L = 10 - For gridpoint data, this is NSLAB, the number of the slab in DIR(, ,L) and in NGRIDC(,L) defining the characteristics of this grid. The values in DIR(, ,) and NGRID(,) are used to promote efficiency. Basically, the station locations with reference to the grid need not be computed every time a grid is used unless it is a grid not used before. A "new" grid could be encountered if the size, location, etc. of a grid changed during the archive. This value is returned by subroutine GRCOMB and incorporated in the GSTORE call sequence. It is returned by GFETCH. If the data are not gridpoint, this value can be something else; in U600 it is the number in the list of unit numbers from which the data came.

L = 11 - The number of the first variable in the (sorted) list in ID(,N) (N=1,NVRBL) for which this variable is needed, when it does not need to be stored after Day 1. When the variable must be stored (to be accessed through OPTION or a similar routine) for all days, LSTORE(11,) is 7777 + the number of the first predictor in the sorted list for which this variable is needed.

L = 12 - NRR, a value that indicates when the data can be discarded. Before that is known, GSTORE (called, for instance, from RDSTR1) will store 0 here. Later, the last date for which these data are needed is stored here.

At the end of Day 1, LMSTR1 sets the values in LSTORE(12,) to indicate the last date for which the data will be needed. (Note that this is not known prior to processing the data for Day 1.) It also sets LSTORE(1,) to zero if the data are no longer needed. LMSTR1 also makes an entry in MSTORE(,J) (J=1,MITEMS) for all items in LSTORE(,) that has LSTORE(1,) > 0.

Then GCPAC discards the data no longer needed by reference to LSTORE(1,). The entries in LSTORE(,) are moved up as necessary so that all remaining values in LSTORE(1,) are not zero and LITEMS set accordingly. Also, the data in the array CORE( ) are moved up so that only the first part is occupied, and the rest is available for future storage. The data stored in direct access on disk, if any, are also moved up, so that the first records are occupied and the remainder are unused. In this way, no fragmentation occurs, and the table for searching is kept small. It is anticipated that most runs of a program using this storage system will not need to carry data from one cycle to another, except for, perhaps, climatological or constant data, which will rise to the top of either CORE( ) or disk and just sit there. Note that any data remaining on disk are not moved to memory, even if there is space available.

At this point, <u>for U201</u>, the values in array MSTORE(L,J) (L=1,7) (J=1,MITEMS) initialized for L are:

- L = 1,4 - The 4-word variable ID taken from LSTORE(1-4,).

- L = 5 - The value taken from LSTORE(11,) which indicates whether or not to store the variable and the first predictor to use it for. This is the same value for each cycle entry in MSTORE(6,).

- L = 6 - The cycle time for which this variable is needed for the date being processed. A variable needed for more than one cycle time will have an entry for each cycle time needed, the cycle times being governed by the time (hours) between cycles, INCCYL.

- L = 7 - The maximum time offset RR (see IDPARS(9)) corresponding to MSTORE(6,).

MSTORE(,) in U201 is not changed after exit from LMSTR1 (entered after Day 1).

<u>For vector programs</u> (e.g., U600), the values in array MSTORE(L,J) (L=1,8) (J=1,MITEMS) initialized for L are:

- L = 1,4 - The 4-word variable ID taken from LSTORE(1-4,).

- L = 5 - The value taken from LSTORE(11,) which indicates whether or not to store the variable and the first predictor to use it for.

- L = 6 - The number of hours ahead the variable needs to be saved. This is based on the input file and how far ahead it needs to be read for the lookahead feature. For instance, if a predictor is on the same file as a predictand that has a tau of 60 hours, then there would be an entry in MSTORE(,) for each projection ahead it needs to be saved as the file is read, the projections and number of entries being governed by the time (hours) between cycles, INCCYL. However, if the predictand is on a separate file, there would be only one entry for the predictor, and MSTORE(6,) would be 0.

- L = 7 - The value taken from LSTORE(10,), which is the number "IN" in the list of unit numbers from which the data came.

- L = 8 - Calculated from the maximum tau [MAXTAU(IN), where "IN" is the input file number in MSTORE(7,)] and the value in MSTORE(6,) for each new cycle. This keeps calls to UPDAT to a minimum.

MSTORE(,) for vector programs is not changed after exit from LMSTR6 (entered after Day 1), except for MSTORE(8,).

So, LSTORE(,) exists basically to manage the actual storage of data, while MSTORE(,) exists to determine which data to use and/or store after Day 1.

Since the data stored in this system are very dynamic (continually moving in and out), the keys in LSTORE(,) are not ordered, as the insertion would probably be more time consuming than the faster searches would offset.

On subsequent days, only the data needed to be stored are stored. For U201, fields needed only as basic fields, possibly smoothed, made into point binaries, or transformed, need not be stored in the internal system. Fields needed to be accessed through OPTION or similar routines must be stored. By using the date of the data needed and MSTORE(,), it can be easily told, as the data are read sequentially, whether they are of the correct date and whether they are needed or not. Additionally, it can be told whether the data are needed to be stored or can be used as they are read and not stored.

Then, after processing any day after Day 1, LMSTR2 for both U201 and vector programs will prepare for purging the data no longer needed by setting LSTORE(1,) = 0 on the basis of the date in LSTORE(12,) and the next date/time to be processed.

Finally, GCPAC discards the data no longer needed.

It is hoped that the size of CORE( ) can be made after a trial run to about the size needed for the data <u>after Day 1</u>, so that disk accesses can be kept low. Subroutine LMSTR2 compiled with /D will print the fields stored. The values in the 5th column of the table are either the starting location in CORE( ) of the variables or, when negative, signifies minus the record number of the variable on disk. Therefore, no negative numbers means that all data are in CORE( ). Positive numbers allow the user to see about how much CORE( ) space is needed. Some data, such as thresholds saved by U201 for linearization routines or directory correspondence records for climatological data, may have been stored on disk for Day 1 that will be needed later. These are never moved to CORE( ). Note that since CORE( ) is a linear array, even if it were much too big, threshing (extreme paging) should not occur, because only one contiguous portion of it would be used.

9. SOFTWARE LIBRARIES

A. Introduction

Software libraries have been built and are being augmented as necessary as new software is written. Generally, there is one library in the directory 'moslib' which contains most or all subroutines that have use across several programs in MOS-2000. It contains the source ".f" files, object ".o" files, a "mos-makefile," a ".a" library file, and a "README" file. The makefile is an executable; when it is executed, all new object files are put into the library, and any source files that have been modified since compilation will be compiled and put into the library. It is executed with the "make" command

```
mf mos-makefile
```

The user should set up macros to allow easy compilation. The default should be level 2 optimization and without the /D option. If a user wants to use the /D option for a particular routine (say XYZ), the routine can be compiled with the command

```
comd XYZ
```

provided the macro is available. This produces the object file, which will be inserted into the library with the make command. See file '.cshre' in directory 'glahn' for useful aliases. for instance, 'mf' = 'make -f'.

In a similar manner, a full ".l" listing can be produced with

```
comdl XYZ or coml XYZ
```

depending, respectively, on whether or not the /D option is desired. Note that the make command will not complete successfully if there is a .l file present; delete the .l file before executing the make command.

When a new source is entered into the directory, the make file must be modified to contain that routine name; it must be entered in only one place (in the "LIB SOURCES =" section), so addition of a routine is very easy--one need only remember to make the entry.

(<u>A caution here</u>. If one were to want to remove a member from a library, it is <u>not</u> sufficient to remove the reference to it in the makefile, because it will still exist in the library, even though the .f and .o files might both be removed. To exorcise it, the .a file must be deleted, and then recreated with the make command. No big deal. However, if one were to put a member in one library, then move it to another library and subsequently modify it, the fact that it might exist in both places, one modified and one not, can give very perplexing results. The order of specifying the libraries on the "load line" will determine whether or not the newer version will be executed!)

The README file is just an ASCII file in which notes are kept about changes to the directory. This is in addition to the comments in the source files themselves. If maintained, this file allows a user to see when routines have been modified and some idea of how and why.

Besides the general 'moslib' directory, there is likely an individual library directory for each major program. For instance, there is a 'u201lib' directory. Its structure is the same as for 'moslib' and contains routines specific to U201. Note that the main program (DRU201) is not located here.

Then, for each main program, each user will likely have a separate directory for running a particular program. Most major programs will have a "driver," which will be tailored to the particular run being made. As an example here, consider program U201. The directory name for U201 would be 'dru201', and would be contained as a directory under the users named directory. This directory contains the driver source 'dru201.f' and object 'dru201.o', and the "load line." The load line, after checkout, will likely contain only three things to be loaded: the 'dru201.o' file, 'moslib', and 'u201lib'. In addition, the directory will contain a file named 'U201.CN'. This is the name of the control file that U201 will expect. Other data files can, of course, be used; they can be in either this same directory, or in another one. All input and output file names, and usually FORTRAN unit numbers, will be input in the control file 'U201.CN'. When the file to be used is not in the same dru201 directory, read the path with the file name in 'U201.CN'. The output(s) for printing, or reading at the console, will be produced in the same dru201 directory, the file names being specified in 'U201.CN'. The load line for U201 is in the executable 'u201.com':

```
      opt/fortran/bin/f77 +O2 +U77 -o u201 dru201.o \
                    -W1,-L/home21/moslib/u201lib -lu201lib \
                    -W1,-L/home21/moslib/moslib -lmoslib \
```

where 'home21/moslib/ is replaced with the appropriate path to 'u201lib' and 'moslib'.

By using this "pattern," each user can segregate the input and output specific to that program, both from other programs to the extend desired and from other users.

There is a common "data" directory in which data that are commonly used by several persons, named 'mos2000data' (e.g., the station directory). This is at the same level in the tree as individual users. In addition, an individual user could have such a directory under his/her own directory for data shared among his/her programs, but not necessarily shared among users. If this is done, a different name should probably be used to avoid confusion, even though the same name is accommodated because the path is different.

B. Location

Accounts for individuals have been set up on the HP755 Blizzard in Room 10300 under 'home21' (e.g., 'home21/glahn'). This is where the libraries reside. Similar accommodation will be made on the CRAY and on other workstations.

C. Documentation

The program and subroutine write-ups are provided in chapters in another document TDL Office Note 00-2, with chapter numbers and an associated numbering scheme for main programs as follows:

Chapter	Program No.	Description
0	001-099	Archiving.
1	100-199	Inventory and archive manipulation.
2	200-299	Predictor preparation.
3	300-399	Merging data sets.
4	400-499	General library routines.
5	500-599	Predictand preparation.
6	600-699	Equation generation.
7	700-799	Making forecasts and manipulation of equations.
8	800-899	Verification.
9	900-999	Operational.

A write-up for a subroutine used exclusively for a particular main program (or is very likely to be used only there) will be placed in the chapter with its respective main program. Main programs and subroutines are given a number (in the upper right) consisting of the "chapter number" followed by a period and a sequence number. In many cases, an external document will be of little use. Many of the parameters in the sometimes lengthy call statement are just for passing to other basic routines and the user would likely not be concerned with them. It is also likely that any use of the routine, other than for using it in running the main program, would be for the purpose of diagnosing an error, making a modification to it, or using it as a pattern for another similar routine. In these cases, the user would go to the source code, where all call sequence parameters are explained. Write-ups for these "exclusive" routines will usually consist of only a purpose and a reference back to the main (or calling) program.

It is envisioned that in MOS-2000 most grid-oriented computations will be done as part of U201, and the documentation is associated with U201. On the other hand, computational routines which operate on a vector (e.g., station values) may have use in a variety of programs, and are generally put into the "general library" chapter. The contents of each chapter parallel the contents of the associated directory. That is, a U201-specific routine such as PRED21 is in the chapter with a number in the 200-299 series and in library 'u201lib', while a more general routine that is in 'moslib' has a number in the 400-499 series. If routines are placed properly, U201 should not have to use the library specific to U600, and vice versa.

While it is not always obvious when written whether a routine will have general use or be specific to a program, one rule must be followed: A routine in one library must not call a routine in another library except for 'moslib'. Specifically, a routine in moslib must not call a routine in another library (e.g., 'u201lib').

10. STATION DIRECTORY FILE--station.tbl

The 'station.tbl' constant file is located on blizzard in the directory 'home21/tdllib/table' and contains information about station locations that is essential to various programs. Similar files will be set up on other computers as necessary. For instance, this file is read by RDSTAD or RDSTAL in U201 and U600. This information includes station call letters, name, elevation, latitude, and longitude that many programs will need. It also includes the WBAN numbers, a link back from the current ICAO station identifiers (call letters) to the previous call letters identifying the station, up to four additional "substitute" call letters, and other information dealing with station history. Although the line (record) length might become too long for easy reading or printout, comments can be added at the end of each line without impacting existing programs as necessary.

The fields are explained below, and the 17 currently identified fields that can reasonably be processed by software can be read with the format:

```
(A8,1XA8,1XA17,4XA2,1XI6,1XI5,1XA1,F7.4,1XA1,F8.4,1XI3,1XI1,
    1XA1,4(1XA8),1XI10,1XI5)
```

The column numbers where the data are located are in parentheses at the end of the explanation. Note that each field is separated by a colon which is skipped when reading with the above format.

CCALL - 8-character ICAO identifier. (1-8)

CCALLS - 8-character call letters formerly used to identify the site. (10-17)

NAME(1:17) - 17 characters of a station name or location. Although 20 characters are available for this field for visual inspection, MOS-2000 software uses only 17 characters. (19-38)

NAME(19:20) - 2 characters of the state (or country) where the station is located. Note that the total station location is made up of 17 characters from the "name," a blank, and 2 characters from the "state," making a total of 20 characters. (40-41)

NBLOCK - 6 characters of which the first 5 make up the WMO block/station number, and the last is used by the U.S. Air Force and NCEP. (43-48)

NELEV - Station elevation in feet, which should be the elevation of the barometer (H_P). (50-54)

SLAT - Latitude of the station in ten-thousandths of degrees. Note that this is read as two fields according to the format above, the first being either "N" or "S" representing, respectively, north or south latitude. (56-63)

SLON - Longitude of the station in ten-thousandths of degrees. Note that this is read as two fields according to the format above, the first being either "E" or "W" representing, respectively, east or west longitude. (65-73)

ITIMEZ - A time zone indicator, being the number of hours the station is different from UTC. Local standard time = UTC + ITIMEZ. (75-77)

ITYPE - The "type" of station, where:
- 0 or blank = manual, precip reported,
- 1 = semiautomated, precip reported,
- 2 = manual, precip not reported,
- 3 = automated, precip not reported, and
- 4 = automated, precip reported.

(79)

OPEN - Indicates whether or not another station is present in the dictionary with a different location, but the same call letters; a blank indicates the call letters are current and an "O" indicates the station has closed and the call letters have been reused for another station. (81)

CCALLK1 - 8-characters containing call letters or ICAO identifiers used to link the station with past reporting stations. (83-90)

CCALLK2 - 8-characters containing call letters or ICAO identifiers used to link the station with past reporting stations. (92-99)

CCALLK3 - 8-characters containing call letters or ICAO identifiers used to link the station with past reporting stations. (101-108)

CCALLK4 - 8-characters containing call letters or ICAO identifiers used to link the station with past reporting stations. (110-117)

IDATE - 10-digit date which indicates the date that data were first received from the station identified as CCALL, when known. If errors are detected in SLAT or SLON, IDATE indicates the date the dictionary entry was corrected. Many are blank, indicating the station has been in the directory and sending data for a long time. (119-128)

IWBAN - The 5-digit WBAN number. (130-134)

IBLANK - 10 positions left vacant for possible future use. (136-145)

COMENT - As many columns as necessary can be used for comments. Usually this will be for visual inspection and not used by software. This is in distinction to the "IBLANK" field above which is of fixed size and is intended for software processing if necessary. Note that very long record lengths may present problems when printing, so comments are best kept short whenever possible. This field is currently restricted to 60 columns. (147-206)

It is expected that when the ICAO station identifiers are not found in an input directory record for vector data, the old call letters (CCALLS) will be looked for, and if not found, CCALLK1, CCALLK2, CCALLK3, and CCALLK4 will be searched for in that order until one is found. That is, there can be up to a total of six sets of call letters associated with one location.

The fact that the first alternate station's call letters is in columns 10-14 and the other four alternates are later in the record is historical. At the beginning of MOS-2000 development, only the current ICAO identifiers (CCALL) and the old SAO call letters (CCALLS) were accommodated. Later, the other four were added. For practical purposes CCALLS, CCALLK1, CCALLK2, CCALLK3, and CCALLK4 are treated in the same way and just represent the order of search for an alternate.

The listing below shows representative entries for columns up through CCALLK4. The directory is alphabetical according to ICAO identifier, and some software depends on this. For purposes of printing, the comments section has been truncated.

```
KHRL     :HRL     :HARLINGEN                :TX:999999:   36:N26.2333:W097.6667: -6:4: :        :        :
KHRO     :HRO     :HARRISON                 :AR:999999: 1385:N36.2667:W093.1500: -6:0: :        :        :
KHRT     :HRT     :HURLBURT FLD             :FL:747770:   89:N30.5167:W086.3000: -6:0: :        :        :
KHSE     :HAT     :HATTERAS       ASOS      :NC:723139:   11:N35.2322:W075.6225: -5:4: :KHAT    :HSE     :
KHSI     :HSI     :HASTINGS       ASOS      :NE:999999: 1955:N40.6000:W098.4333: -6:4: :        :        :
KHSP     :HSP     :HOT SPRINGS              :VA:999999: 3768:N37.9500:W079.8167: -5:2: :        :        :
KICT     :ICT     :WICHITA        ASOS      :KS:724500: 1340:N37.6500:W097.4167: -6:4: :        :        :
KIDA     :IDA     :IDAHO FALLS              :ID:999999: 4744:N43.5167:W112.0667: -7:0: :        :        :
KIDI     :        :STEWART FIELD            :PA:999999: 1407:N40.6300:W079.1000: -5:4: :        :        :
KIEN     :        :PINE RIDGE               :SD:999999: 3274:N43.0300:W102.5200: -7:4: :        :        :
KIFP     :IFP     :BULLHEAD CITY            :AZ:999999:  538:N35.1667:W114.5000: -7:0: :P06     :        :
KIGM     :IGM     :KINGMAN        ASOS      :AZ:723700: 3389:N35.2667:W113.9500: -7:4: :        :        :
```

11. VARIABLE CONSTANTS FILE--mos2000id.tbl

The 'mos2000id.tbl' constant file is located on blizzard in the directory 'home21/tdllib/tbl' and contains information about MOS-2000 variables that is useful to various programs. Similar files will be set up on other computers as necessary. It is read by SETPLN (as called by RDPRED in U201, RDVRBL in U600, and RDVR66 in U660. This information is for plain language identification of variables, their units, the degree of precision for which the variable is to be packed in the TDLPACK format, and constants for gridprinting. Gridprinting is a capability in U201 that can be used for checkout. Although the line (record) length might become too long for easy reading or printout, other information could be added at the end of each line without impacting existing programs if it becomes desirable to do so.

The fields are explained below and read with the format:

(3(I9,1X),I3,2XA32,I3,F11.4,F10.4,F9.2,F8.2,2XA12)

The column numbers where the data are located are in parentheses at the end of the explanation.

IDPRED(J), J=1,4 - The 4-word variable ID, sans any possible threshold. This can be very specific so that it applies to only one variable, or can be more generic so that it applies to several possible variables. For instance, it might apply to 500-mb height, no matter whether it is a grid, smoothed or unsmoothed, or a vector obtained by either linear or quadratic interpolation. Of course, the using program must be "smart" enough to deal with the specifics. (1-33)

PLAIN - 32 characters of plain language identifying the variable. As with ID( ), this can be specific or generic. Certain rules must be followed for the using routines to operate properly. These are:

- o - The first 5 characters should not be used. The level followed by a blank is inserted by the software to conform to the UUUU in the ID( ).
- o - Columns 23-32 should be left blank to accommodate inserting further description of the variable, including processing information. See the subroutine SETPLN writeup for more specific information on what is inserted.

(36-67)

ISCALE - A digit representing the number of decimal places to include in the packed output. Said in another way, each value before packing is multiplied by 10^{ISCALE} and rounded to an integer. ISCALE can be positive or negative. (68-70)

SMULT - Multiply the gridpoints by SMULT before gridprinting. (71-81)

SADD - After applying SMULT, add SADD to the result before gridprinting. (82-91)

CONT - The contour interval in the <u>original</u> units of the data, not after applying SMULT. (92-100)

ORIGIN - The origin for contouring in terms of the <u>original</u> data. (101-108)

UNITS - The units for which the variable is (usually) available, before applying SMULT. (111-122)

So far, SETPLN distinguishes variables mainly on the basis of CCCFFF (the first 6 digits of IDPRED(1). Undoubtedly, SETPLN will need to be augmented to deal with some specific variables. For instance, if the scaling ISCALE for height were to vary by height, then at least one more entry would have to be made, and the software be smart enough to use the specific IDPRED( ), or even to use a specific IDPRED( ) in certain circumstances and a generic one in others. The writeup for SETPLN (which should be kept up to date) should be consulted for its capability for distinguishing between specific and generic entries. As of this writing, their capabilities include:

- o The full CCCFFF^DD is checked prior to checking CCCFFF. If a model-specific entry is needed, place it <u>before</u> the more generic entry.

- o In the absence of a full CCCFFFB^^, a point binary is given the ISCALE value of 0. That is, if CCCFFF1^^ is not in the directory, the variable CCCFFF would be given ISCALWE = 0; if CCCFFF1^^ is in the directory ISCALE will be retained as the value in the table.

- o In the absence of a full CCCFFFB, a grid binary is given the ISCALE( ) value of 2.

- o AEV archive data have a DD = 80 for MOS forecasts, 81 for local forecasts, and 82 for matching observations. SETPLN accommodates these values of DD.

The listing below shows representative entries. Other entries should be made so that IDPRED( ) is generally ordered just for convenience of use, not that it's required by the software; the order is not absolute, though (e.g., CCCFFF^08 would precede CCCFFF^00). The user should consult the file for the current information.

001000107	000000000	000000000	000	HEIGHT	3	1.	0	60.	0.	M
001000006	000000000	000000000	000	HEIGHT	3	1.	0.	60.	0.	M
001050007	000000000	000000000	000	HEIGHT	2	1.	0.	60.	0.	M
001000000	000000000	000000000	000	HEIGHT	1	1.	0.	60.	0.	M
002000000	000000000	000000000	000	TEMP	1	1.	-273.15	5.	273.15	DEG C
003000000	000000000	000000000	000	REL HUM	1	1.	0.	5.	0.	PERCENT
003200000	000000000	000000000	000	6-H PRECIP	1	1.	0.	5.	0.	KG/SQM
003210000	000000000	000000000	000	6-H CONV PRECIP	1	1.	0.	5.	0.	KG/SQM
003220000	000000000	000000000	000	6-H NC PRECIP	1	1.	0.	5.	0.	KG/SQM
003250000	000000000	000000000	000	PRECIP WATER	1	1.	0.	5.	0.	KG/SQM
003252000	000000000	000000000	000	PRECIP WATER	0	1.	0.	5.	0.	KG/SQM
004000000	000000000	000000000	000	GRID U-WIND	1	1.	0.	5.	0.	M/SEC
004002000	000000000	000000000	000	GRID GEO U-WIND	1	1.	0.	5.	0.	M/SEC
004010000	000000000	000000000	000	EARTH U-WIND	1	1.	0.	5.	0.	M/SEC
004012000	000000000	000000000	000	EARTH GEO U-WIND	1	1.	0.	5.	0.	M/SEC
004100000	000000000	000000000	000	GRID V-WIND	1	1.	0.	5.	0.	M/SEC
004102000	000000000	000000000	000	GRID GEO V-WIND	1	1.	0.	5.	0.	M/SEC
004110000	000000000	000000000	000	EARTH V-WIND	1	1.	0.	5.	0.	M/SEC
004112000	000000000	000000000	000	EARTH GEO V-WIND	1	1.	0.	5.	0.	M/SEC
004210000	000000000	000000000	000	WIND SPEED	1	1.	0.	5.	0.	M/SEC
004212000	000000000	000000000	000	GEO WIND SPEED	1	1.	0.	5.	0.	M/SEC

12. GRIDPOINT DATA ARCHIVES

A. Format

The format of the gridpoint data is explained in Section B of Chapter 6 "Sequential Data Archive Files."

B. Nested Grid Model Archive

Description of Grid and Variables Archived

Initial and forecast fields from NCEP's Nested Grid Model (NGM) are saved daily from the 0000 and 1200 UTC forecast cycles on the grid shown in Fig. 12-1. This grid corresponds to a polar stereographic projection oriented 105° W with a grid resolution of 190.5 km at 60° N. The archive consists of data at 1558 gridpoints, with 41 points in the x-direction (from left to right in the diagram, or approximately west to east) and 38 points in the y-direction (from bottom to top in the diagram, or approximately south to north). The data are packed in TDLPACK format as described in Chapter 5 by subroutine PACK2D beginning at the lower left corner gridpoint (defined as x=1, y=1). The data are returned by the subroutine UNPACK starting at the lower left, proceeding left to right (in order of increasing x), then bottom to top (in order of increasing y).

The data archived are extracted from NCEP files (see below). Note that these NCEP files contain the data in GRIB format on a grid which is a superset of the one shown in Fig. 12-1. Note, too, that the NGM computational grid has a finer resolution (approximately 90.75 km) than the one on which the data are archived (see below for further details).

Table 12-1 lists the MOS-2000 identifier for each of the fields archived, the corresponding GRIB identifier of the field in the NCEP files, the precision of the data, and the units and projections for which the variables are saved. Note that standard GRIB units were not used in archiving the pressure and precipitation amount fields because of a need to retain consistency with previous archive datasets (see below).

Archive Process

At the end of each forecast cycle (0000 and 1200 UTC), a job is run on the Cray to access the NCEP NGM GRIB files, read the fields of interest, extract the data for the area shown in Fig. 12-1, and put the data into TDLPACK format.[1] The TDLPACK data are then stored for each day and cycle in a set of daily rotating files. Once a week, the daily files are concatenated and stored on both disk and tape. After a month's data are collected, the disk and tape files are complete. The disk files are not retained for an extended period of time, but are removed as disk space is required for more current data. The tape files are structured so that a volume-set contains a year of data divided into 12 files corresponding to the 12 months of the year. Note that the entire archive process and the storage of the data are currently done on the Cray. Note, also, that all of these files are written in binary format

[1]NCEP is in the process of converting from the CRAYs to an IBM mainframe. This writeup reflects the current situation.

with standard Unix blocking (that is, the data are not blocked according to the Cray Operating System, but are, in fact, non-COS blocked) so that the data can be read on either the Cray main-frame or the HP workstation platform.

File Names

The following files are used in the archive process described above:

NCEP GRIB files: /com/rgl/prod/rgl.**yymmdd**/ngm.T**hh**Z.pgrb.f**xx**, where **yy** equals the two digit year, **mm** equals the two digit month, **dd** equals the two digit day of the month, **hh** equals the cycle (00 or 12), and **xx** equals the forecast projection (00, 06, 12, 18, 24, 30, 36, 42, and 48).

Daily rotating file: /tdl4/we21ar/rotate/ngm/**yyyymmddhh**, where **yyyy** equals the four digit year, **mm** equals the two digit month, **dd** equals the two digit day of the month, and **hh** equals the cycle (00 or 12).

Monthly file: /tdl4/we21ar/archive/ngm/ngm**yyyymm**, where **yyyy** equals the four digit year and **mm** equals the two digit month. This file is stored on disk; a copy of the file is also stored on tape as described in the yearly file below.

Yearly file: The volume set name of the tape is designated as: we21ar/tdlngm**yyyys**, where **yyyy** corresponds to the four digit year and **s** corresponds to the tape status, that is, a **p** for the primary tape set and a **b** for the backup set. Each monthly file within the yearly volume set is designated as: NGM**yyyymm**, where **yyyy** equals the four digit year and **mm** equals the two digit month.

Archive Tapes

As indicated above, the archived NGM data are eventually stored on cartridge tapes and are listed in the Cray Reel Librarian (CRL) system. That database system can be queried by the Cray user to determine the names and availability of the various volume sets. Generally, the tapes are stored in a separate room from the Cray, and a request must be made to the tape librarian to bring the tapes back to the Cray silos in order to be used. This retrieval of the database usually takes 24 hours. Once the tapes are placed back into the silo, the user can copy the data from tape to disk and then proceed with a specific task. A file on the Cray named: /jdsk41/we/tdllib/log/tapes/we21ar.list gives the names of the NGM files. However, this list is not necessarily current so that the interested user should query the CRL system. Use the "man" facility for the command "rlr" for further details on querying CRL.

Unlike some of the other archives, at the time this section was written, no effort had been made to store the tape data on more accessible disk files.

Period of Record

The NGM archive begins with 0000 UTC data on October 2, 1986, and continues to the present. The archive was converted to run on the Cray in March 1997; data prior to that time were originally collected on the Hitachi Data System (HDS) main-frame machines. The format of the original archive datasets is described fully in TDL Office Note 74-14, pp. IV-16 through IV-18. This same section also provides some information about changes made to the NGM during the archive period. The older HDS archive datasets were converted to TDLPACK format and to Unix blocking by use of a program identified as U030 and designed specifically for that purpose. Note, also, that the original HDS archive of NGM data began in September 1984. Data prior to October 1986 were coincident with a period of numerous changes in the formulation of the NGM. Consequently, these earlier data were never used in NGM MOS development and were discarded when the archives were converted to the TDLPACK format. Finally, data for the period of October 1986 through September 1987 were not collected in real-time, but were gathered by retrospectively running the NGM on that year's set of initial conditions.

For information on the NGM, the reader is referred to an article entitled "The Regional Analysis and Forecast System of the National Meteorological Center" by Hoke et al. and published in the September 1989 edition of Weather and Forecasting.

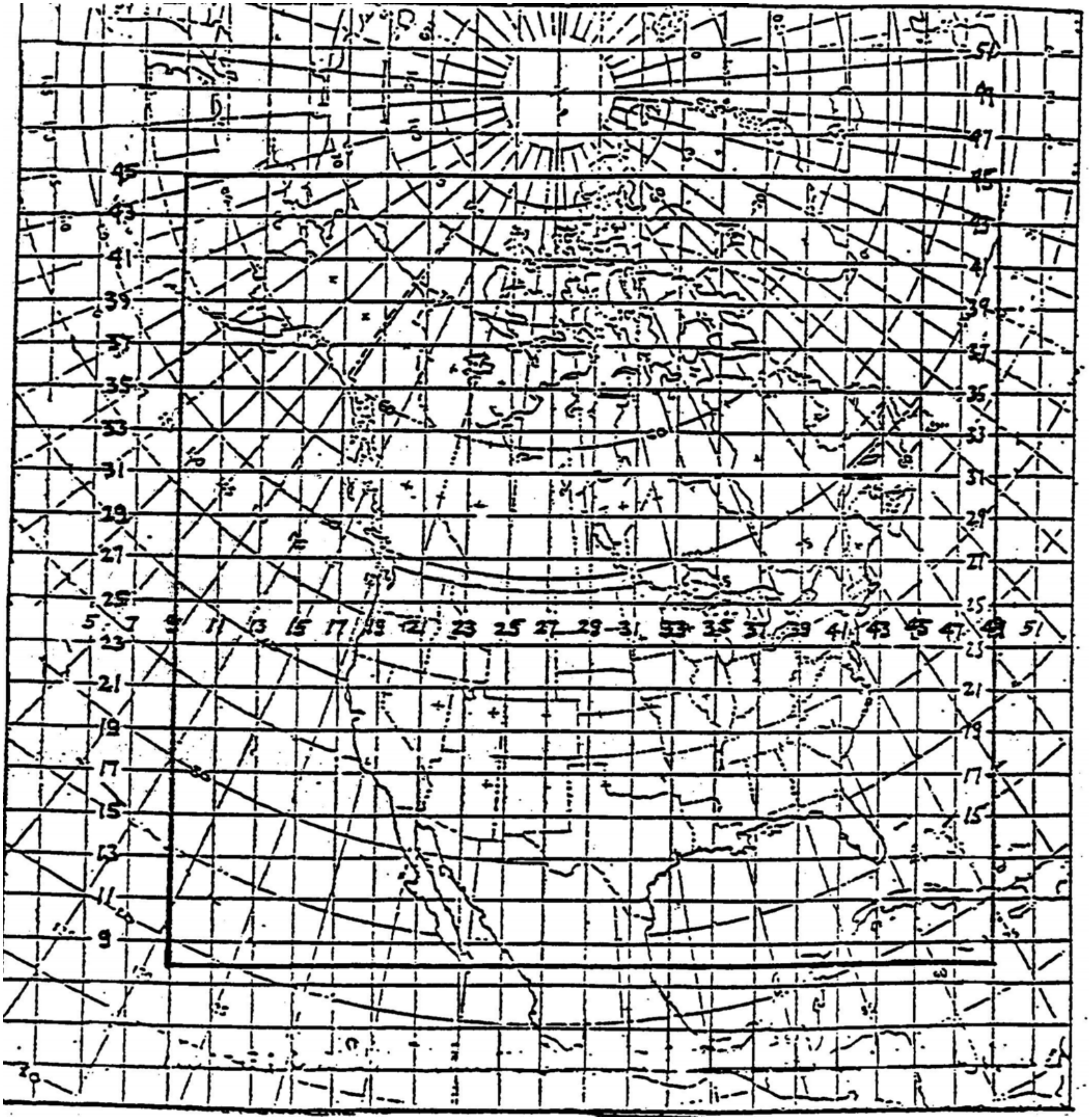

Figure 12-1. 41 X 38 grid (dark box) on which NGM data are archived. Only alternate grid lines are shown. The numbers refer to a reference grid. The upper right point is (41,38) with reference to the lower left (1,1).

Table 12-1. Fields archived from the Nested Grid Model. The columns labelled CCCFFFBDD and VLLLLUUUU indicate the first two words of the MOS-2000 identifier, respectively. The third word of the identifier contains the forecast projection; the fourth word is always set to 0. The column labeled Plain Language gives the plain language stored in the 32 bytes allowed for variable description in the TDLPACK format. The Units and the Prec. (for precision) columns describe the units and subsequent precision of the field after the data are unpacked. The GRIB ID columns lists the bytes by number which are contained in the GRIB Product Definition Section (PDS) and which precisely specify the variable. These values are included in hexadecimal format and are essential to the archive of the data. Note that the value of tt corresponds to the ending forecast projection, xx equals tt-6, and yy equals tt-12. While not indicated in the table below, all variables in the NGM archive currently created on the Cray are identified by a value of 39 (decimal) in byte 6 (indicating the NGM) and by a value of 6 in byte 7 (identifies the data as being stored on a polar stereographic grid with dimensions of 53 by 45). Forecast projections are in hours from 0000 and 1200 UTC.

CCCFFFBDD	VLLLLUUUU	Plain Language	Units	Prec.	GRIB ID							Forecast Projection								
					9	10	11/12	18	19	20	21	0	6	12	18	24	30	36	42	48
001000006	000000300	300 MB HGT NGM	m	1.0	07	64	012C	01	tt	00	00	x	x	x	x	x	x	x	x	x
001000006	000000500	500 MB HGT NGM	m	1.0	07	64	01F4	01	tt	00	00	x	x	x	x	x	x	x	x	x
001000006	000000700	700 MB HGT NGM	m	1.0	07	64	02BC	01	tt	00	00	x	x	x	x	x	x	x	x	x
001000006	000000850	850 MB HGT NGM	m	1.0	07	64	0352	01	tt	00	00	x	x	x	x	x	x	x	x	x
001000006	000000950	950 MB HGT NGM	m	1.0	07	64	03B6	01	tt	00	00	x	x	x	x	x	x	x	x	x
001000006	000001000	1000 MB HGT NGM	m	1.0	07	64	03E8	01	tt	00	00	x	x	x	x	x	x	x	x	x
001201006	000000000	MSL(STND) PRES NGM	mb	0.1	02	66	0000	01	tt	00	00	x	x	x	x	x	x	x	x	x
002000006	000000300	300 MB TEMP NGM	K	0.1	0B	64	012C	01	tt	00	00	x	x	x	x	x	x	x	x	x
002000006	000000500	500 MB TEMP NGM	K	0.1	0B	64	01F4	01	tt	00	00	x	x	x	x	x	x	x	x	x
002000006	000000700	700 MB TEMP NGM	K	0.1	0B	64	02BC	01	tt	00	00	x	x	x	x	x	x	x	x	x
002000006	000000750	750 MB TEMP NGM	K	0.1	0B	64	02EE	01	tt	00	00	x	x	x	x	x	x	x	x	x
002000006	000000800	800 MB TEMP NGM	K	0.1	0B	64	0320	01	tt	00	00	x	x	x	x	x	x	x	x	x
002000006	000000850	850 MB TEMP NGM	K	0.1	0B	64	0352	01	tt	00	00	x	x	x	x	x	x	x	x	x
002000006	000000900	900 MB TEMP NGM	K	0.1	0B	64	0384	01	tt	00	00	x	x	x	x	x	x	x	x	x
002000006	000000950	950 MB TEMP NGM	K	0.1	0B	64	03B6	01	tt	00	00	x	x	x	x	x	x	x	x	x
002000006	000001000	1000 MB TEMP NGM	K	0.1	0B	64	03E8	01	tt	00	00	x	x	x	x	x	x	x	x	x
003000006	000000300	300 MB RH NGM	%	1.0	34	64	012C	01	tt	00	00	x	x	x	x	x	x	x	x	x
003000006	000000500	500 MB RH NGM	%	1.0	34	64	01F4	01	tt	00	00	x	x	x	x	x	x	x	x	x
003000006	000000700	700 MB RH NGM	%	1.0	34	64	02BC	01	tt	00	00	x	x	x	x	x	x	x	x	x
003000006	000000750	750 MB RH NGM	%	1.0	34	64	02EE	01	tt	00	00	x	x	x	x	x	x	x	x	x
003000006	000000800	800 MB RH NGM	%	1.0	34	64	0320	01	tt	00	00	x	x	x	x	x	x	x	x	x
003000006	000000850	850 MB RH NGM	%	1.0	34	64	0352	01	tt	00	00	x	x	x	x	x	x	x	x	x
003000006	000000900	900 MB RH NGM	%	1.0	34	64	0384	01	tt	00	00	x	x	x	x	x	x	x	x	x

003000006	000000950	950 MB RH NGM	%	1.0	34 64 03B6	01 tt 00 00	x	x	x	x	x	x	x	x	x
003000006	000001000	1000 MB RH NGM	%	1.0	34 64 03E8	01 tt 00 00	x	x	x	x	x	x	x	x	x
003002006	000000470	MEAN RH(1.-.47SGY) NGM	%	1.0	34 6C 2F64	01 tt 00 00	x	x	x	x	x	x	x	x	x
003211006	000000000	6-H TOTAL PRECIP NGM	m	0.0001	3D 01 0000	01 xx tt 04		x		x		x		x	
003221006	000000000	12-H TOTAL PRECIP NGM	m	0.0001	3D 01 0000	01 yy tt 04			x		x		x		x
003350006	000000000	TOTAL P. WATER NGM	kg/m^2	1.0	3C 6C 0064	01 tt 00 00	x	x	x	x	x	x	x	x	x
004000006	000000300	300 MB U GRD NGM	m/s	0.1	21 64 012C	01 tt 00 00	x	x	x	x	x	x	x	x	x
004000006	000000500	500 MB U GRD NGM	m/s	0.1	21 64 01F4	01 tt 00 00	x	x	x	x	x	x	x	x	x
004000006	000000700	700 MB U GRD NGM	m/s	0.1	21 64 02BC	01 tt 00 00	x	x	x	x	x	x	x	x	x
004000006	000000850	850 MB U GRD NGM	m/s	0.1	21 64 0352	01 tt 00 00	x	x	x	x	x	x	x	x	x
004000006	000000950	950 MB U GRD NGM	m/s	0.1	21 64 03B6	01 tt 00 00	x	x	x	x	x	x	x	x	x
004020006	000000010	10 M U GRD NGM	m/s	0.1	21 69 000A	01 tt 00 00	x	x	x	x	x	x	x	x	x
004100006	000000300	300 MB V GRD NGM	m/s	0.1	22 64 012C	01 tt 00 00	x	x	x	x	x	x	x	x	x
004100006	000000500	500 MB V GRD NGM	m/s	0.1	22 64 01F4	01 tt 00 00	x	x	x	x	x	x	x	x	x
004100006	000000700	700 MB V GRD NGM	m/s	0.1	22 64 02BC	01 tt 00 00	x	x	x	x	x	x	x	x	x
004100006	000000850	850 MB V GRD NGM	m/s	0.1	22 64 0352	01 tt 00 00	x	x	x	x	x	x	x	x	x
004100006	000000950	950 MB V GRD NGM	m/s	0.1	22 64 03B6	01 tt 00 00	x	x	x	x	x	x	x	x	x
004120006	000000010	10 M V GRD NGM	m/s	0.1	22 69 000A	01 tt 00 00	x	x	x	x	x	x	x	x	x
005003006	000000300	300 MB VV NGM	mb/s	0.01	27 64 012C	01 tt 00 00		x	x	x	x	x	x	x	x
005003006	000000500	500 MB VV NGM	mb/s	0.01	27 64 01F4	01 tt 00 00		x	x	x	x	x	x	x	x
005003006	000000700	700 MB VV NGM	mb/s	0.01	27 64 02BC	01 tt 00 00		x	x	x	x	x	x	x	x
005003006	000000850	850 MB VV NGM	mb/s	0.01	27 64 0352	01 tt 00 00		x	x	x	x	x	x	x	x
005003006	000000950	950 MB VV NGM	mb/s	0.01	27 64 03B6	01 tt 00 00		x	x	x	x	x	x	x	x

C. Eta Model Archive

Description of Grid and Variables Archived

Initial and forecast fields from NCEP's Eta model are saved daily from the 0000 and 1200 UTC forecast cycles on the grid shown in Fig. 12-2. This grid corresponds to a polar stereographic projection oriented 105° W with a grid resolution of 90.75 km at 60° N. The archive consists of data at 2806 gridpoints, with 61 points in the x-direction (from left to right in the diagram, or approximately west to east) and 46 points in the y-direction (from bottom to top in the diagram, or approximately south to north). The data are packed in TDLPACK format as described in Chapter 5 by subroutine PACK2D beginning at the lower left corner gridpoint (defined as x=1, y=1). The data are returned by the subroutine UNPACK starting at the lower left, proceeding left to right (in order of increasing x), then bottom to top (in order of increasing y).

The data archived are extracted from NCEP files (see below). Note that these NCEP files contain the data in GRIB format on a grid which is also shown in Fig. 12-2. Note, too, that the Eta computational grid has varied during the time of the archive (see below). The Eta model is currently run at a much finer horizontal resolution (32 km as of November 1998) than the archive grid used. The Eta model post-processor does the interpolation to the appropriate grid (see below for further details).

Table 12-2 lists the MOS-2000 identifier for each of the fields archived, the corresponding GRIB identifier of the field in the NCEP files, the precision of the data, and the units and projections for which the variables are saved. Note that standard GRIB units were used in archiving all the variables. As a consequence, variables archived initially on the Hitachi Data System main-frame (from April 1994 until February 1997) were converted to the appropriate units when the archives were moved to the Cray.

Archive Process

At the end of the 0000 and 1200 UTC Eta forecast cycle, a job is run on the Cray to access the NCEP Eta GRIB files, read the fields of interest, extract the data for the area shown in Fig. 12-2, and put the data into TDLPACK format.[2] The TDLPACK data are then stored for each day and cycle in a set of daily rotating files. Once a week, the daily files are concatenated and stored on both disk and tape. After a month's data are collected, the disk and tape files are complete. The disk files are not retained for an extended period of time, but are removed as disk space is required for more current data. The tape files are structured so that a volume-set contains a year of data divided into 12 files corresponding to the 12 months of the year. Note that the entire archive process and the storage of the data are currently done on the Cray. Note, also, that all of these files are written in binary format with standard Unix blocking (that is, the data are not blocked according to the Cray Operating System, but are, in fact, non-COS blocked) so that the data can be read on either the Cray main-frame or the HP workstation platform.

[2]NCEP is in the process of converting from the CRAYs to an IBM mainframe. This writeup reflects the current situation.

<u>File Names</u>

The following files are used in the archive process described above:

NCEP GRIB files: /com/eta/prod/erl.**yymmdd**/eta.T**hh**Z.GRBGRD**xx**.tm00, where **yy** equals the two digit year, **mm** equals the two digit month, **dd** equals the two digit day of the month, **hh** equals the cycle (00 or 12), and **xx** equals the forecast projection (00, 06, 12, 18, 24, 30, 36, 42, and 48).

Daily rotating file: /tdl4/we21ar/rotate/eta/**yyyymmddhh**, where **yyyy** equals the four digit year, **mm** equals the two digit month, **dd** equals the two digit day of the month, and **hh** equals the cycle (00 or 12).

Monthly file: /tdl4/we21ar/archive/eta/eta**yyyymm**, where **yyyy** equals the four digit year and **mm** equals the two digit month. This file is stored on disk; a copy of the file is also stored on tape as described in the yearly file below.

Yearly file: The volume set name of the tape is designated as: we21ar/tdleta**yyyys**, where **yyyy** corresponds to the four digit year and **s** corresponds to the tape status, that is, a **p** for the primary tape set and a **b** for the backup set. Each monthly file within the yearly volume set is designated as: ETA**yyyymm**, where **yyyy** equals the four digit year and **mm** equals the two digit month.

<u>Archive Tapes</u>

As indicated above, the archived Eta data are eventually stored on cartridge tapes and are listed in the Cray Reel Librarian (CRL) system. That database system can be queried by the Cray user to determine the names and availability of the various volume sets. Generally, the tapes are stored in a separate room from the Cray, and a request must be made to the tape librarian to bring the tapes back to the Cray silos in order to be used. This retrieval of the database usually takes 24 hours. Once the tapes are placed back into the silo, the user can copy the data from tape to disk and then proceed with a specific task. A file on the Cray named: /jdsk41/we/tdllib/log/tapes/we21ar.list gives the names of the Eta model files. However, this list is not necessarily current so that the interested user should query the CRL system. Use the "man" facility for the command "rlr" for further details on querying CRL.

As of November 1998, all of the archived Eta model data have been moved back from tape to the Cray4 /dm (data migration) disk system for purposes of development. The Eta data are stored in the following file system: /dm/we21ar/modeldata/eta/eta**yyyymm**, where **yyyy** equals the four digit year and mm equals the two digit month. Data are available from April 1994 to the present in the /dm system on Cray4. However, the user is reminded to check the status of these files by issuing the "ls -l" command. If the files have been migrated to the silo tape system, the user can retrieve the files within a minute or so by issuing a "dmget" command. Alternatively, the submission of a batch job to Cray 4 which requires the /dm files will require the system to retrieve the files back to disk.

Period of Record

The Eta archive begins with 0000 UTC data on March 1, 1994, and continues to the present. The archive was converted to run on the Cray in March 1997; data prior to that time were originally collected on the Hitachi Data System (HDS) main-frame machines. The format of the original archive datasets was analogous to the format of the NGM archive which is described fully in TDL Office Note 74-14, pp. IV-16 through IV-18. The older HDS Eta archive datasets were converted to TDLPACK format and to Unix blocking by use of a program identified as U030 and designed specifically for that purpose. At the time of conversion, the units of all the fields archived on the HDS were converted to the standard GRIB units listed in Table 12-2.

A number of articles have been published since 1994 in Weather and Forecasting which describe the Eta model or changes made to the model. Extensive changes have been made to the Eta model since TDL began an archive in 1994. The changes are documented on the NCEP web page. This resource also includes a model description along with a list of related references. The interested user may wish to look at the following:

http://nic.fb4.noaa.gov:8000/research/gcip.html - model description and links to the reference list and the log of eta model changes

http://nic.fb4.noaa.gov:8000/research/gcp/etarefs.html - reference list

http://nic.fb4.noaa.gov:8000/research/eta.log.html - log of operational eta model changes, June 1995 - June 1998

http://nic.fb4.noaa.gov:8000/pub/gcp/eta/eta.log.code - log of operational eta model changes, Jan. 1992 - June 1998

http://www.ncep.noaa.gov/NCO/PMB/announcements/meso_eta32.html - 5/18/98 announcement regarding Meso-Eta 32 implementation

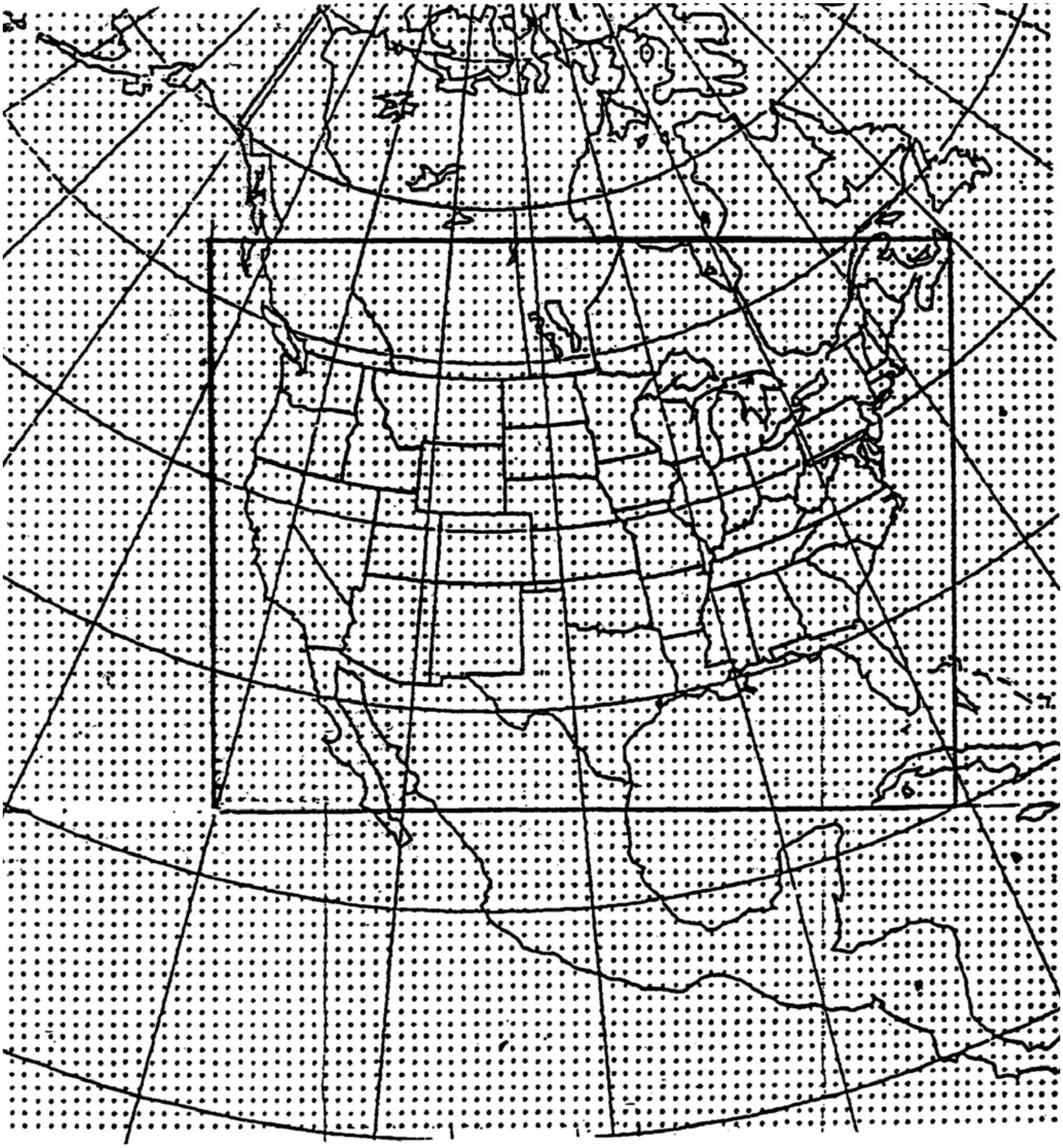

Figure 12-2. 61 X 46 grid on which ETA data are archived. The upper right point is (61,46) with reference to the lower left (1,1).

Table 12-2. Fields archived from the Eta Model. The columns labelled CCCFFFBDD and VLLLLUUUU indicate the first two words of the MOS-2000 identifier, respectively. Though not shown in the table, the third word of the identifier contains the forecast projection; the fourth word is always set to 0. The column labeled Plain Language gives the plain language stored in the 32 bytes allowed for variable description in the TDLPACK format. The Units and the Prec. (for precision) columns describe the units and subsequent precision of the field after the data are unpacked. The GRIB ID columns lists the bytes by number which are contained in the GRIB Product Definition Section (PDS) and which precisely specify the variable. These values are included in hexadecimal format and are essential to the archive of the data. Note that the value of tt corresponds to the ending forecast projection. In the identifier for precipitation amount, xx equals tt-6, and yy equals tt-12. While not indicated in the table below, all variables in the Eta archive currently created on the Cray are identified by a value of 89 (decimal) in byte 6 (indicating the Eta model) and by a value of 104 in byte 7 (identifies the data as being stored on a polar stereographic grid with dimensions of 147 by 110). Forecast projections are in hours from 0000 and 1200 UTC.

CCCFFFBDD	VLLLLUUUU	Plain Language	Units	Prec.	GRIB ID							Forecast Projection								
					9	10	11/12	18	19	20	21	0	6	12	18	24	30	36	42	48
001000007	000000150	150 MB HGT ETA	m	1.0	07	64	0096	01	tt	00	00	x	x	x	x	x	x	x	x	x
001000007	000000200	200 MB HGT ETA	m	1.0	07	64	00C8	01	tt	00	00	x	x	x	x	x	x	x	x	x
001000007	000000250	250 MB HGT ETA	m	1.0	07	64	00FA	01	tt	00	00	x	x	x	x	x	x	x	x	x
001000007	000000300	300 MB HGT ETA	m	1.0	07	64	012C	01	tt	00	00	x	x	x	x	x	x	x	x	x
001000007	000000400	400 MB HGT ETA	m	1.0	07	64	0190	01	tt	00	00	x	x	x	x	x	x	x	x	x
001000007	000000500	500 MB HGT ETA	m	1.0	07	64	01F4	01	tt	00	00	x	x	x	x	x	x	x	x	x
001000007	000000600	600 MB HGT ETA	m	1.0	07	64	0258	01	tt	00	00	x	x	x	x	x	x	x	x	x
001000007	000000700	700 MB HGT ETA	m	1.0	07	64	02BC	01	tt	00	00	x	x	x	x	x	x	x	x	x
001000007	000000750	750 MB HGT ETA	m	1.0	07	64	02EE	01	tt	00	00	x	x	x	x	x	x	x	x	x
001000007	000000800	800 MB HGT ETA	m	1.0	07	64	0320	01	tt	00	00	x	x	x	x	x	x	x	x	x
001000007	000000850	850 MB HGT ETA	m	1.0	07	64	0352	01	tt	00	00	x	x	x	x	x	x	x	x	x
001000007	000000900	900 MB HGT ETA	m	1.0	07	64	0384	01	tt	00	00	x	x	x	x	x	x	x	x	x
001000007	000000950	950 MB HGT ETA	m	1.0	07	64	03B6	01	tt	00	00	x	x	x	x	x	x	x	x	x
001000007	000001000	1000 MB HGT ETA	m	1.0	07	64	03E8	01	tt	00	00	x	x	x	x	x	x	x	x	x
001004007	000000273	273 K HGT ETA	m	0.1	07	04	0000	01	tt	00	00	x	x	x	x	x	x	x	x	x
001105007	000000000	TROP PRES ETA	Pa	10.0	01	07	0000	01	tt	00	00	x	x	x	x	x	x	x	x	x
001107007	000000970	BDY LYR1 PRES ETA	Pa	10.0	01	74	1D00	01	tt	00	00	x	x	x	x	x	x	x	x	x
001210007	000000000	MSL(MSN) PRES ETA	Pa	10.0	07	66	0000	01	tt	00	00	x	x	x	x	x	x	x	x	x
002000007	000000300	300 MB TEMP ETA	K	0.1	0B	64	012C	01	tt	00	00	x	x	x	x	x	x	x	x	x
002000007	000000400	400 MB TEMP ETA	K	0.1	0B	64	0190	01	tt	00	00	x	x	x	x	x	x	x	x	x
002000007	000000500	500 MB TEMP ETA	K	0.1	0B	64	01F4	01	tt	00	00	x	x	x	x	x	x	x	x	x
002000007	000000600	600 MB TEMP ETA	K	0.1	0B	64	0258	01	tt	00	00	x	x	x	x	x	x	x	x	x

002000007	000000700	700 MB TEMP ETA	K	0.1	0B 64 02BC	01 tt 00 00	x	x	x	x	x	x	x	x	x
002000007	000000750	750 MB TEMP ETA	K	0.1	0B 64 02EE	01 tt 00 00	x	x	x	x	x	x	x	x	x
002000007	000000800	800 MB TEMP ETA	K	0.1	0B 64 0320	01 tt 00 00	x	x	x	x	x	x	x	x	x
002000007	000000850	850 MB TEMP ETA	K	0.1	0B 64 0352	01 tt 00 00	x	x	x	x	x	x	x	x	x
002000007	000000900	900 MB TEMP ETA	K	0.1	0B 64 0384	01 tt 00 00	x	x	x	x	x	x	x	x	x
002000007	000000950	950 MB TEMP ETA	K	0.1	0B 64 03B6	01 tt 00 00	x	x	x	x	x	x	x	x	x
002000007	000001000	1000 MB TEMP ETA	K	0.1	0B 64 03E8	01 tt 00 00	x	x	x	x	x	x	x	x	x
002001007	000000002	2 M TEMP ETA	K	0.1	0B 69 0002	01 tt 00 00	x	x	x	x	x	x	x	x	x
002002007	010000970	BDY LYR1 TEMP ETA	K	0.1	0B 74 1D00	01 tt 00 00	x	x	x	x	x	x	x	x	x
002005007	000000000	TROP TEMP ETA	K	0.1	0B 07 0000	01 tt 00 00	x	x	x	x	x	x	x	x	x
003000007	000000300	300 MB RH ETA	%	1.0	34 64 012C	01 tt 00 00	x	x	x	x	x	x	x	x	x
003000007	000000400	400 MB RH ETA	%	1.0	34 64 0190	01 tt 00 00	x	x	x	x	x	x	x	x	x
003000007	000000500	500 MB RH ETA	%	1.0	34 64 01F4	01 tt 00 00	x	x	x	x	x	x	x	x	x
003000007	000000600	600 MB RH ETA	%	1.0	34 64 0258	01 tt 00 00	x	x	x	x	x	x	x	x	x
003000007	000000700	700 MB RH ETA	%	1.0	34 64 02BC	01 tt 00 00	x	x	x	x	x	x	x	x	x
003000007	000000750	750 MB RH ETA	%	1.0	34 64 02EE	01 tt 00 00	x	x	x	x	x	x	x	x	x
003000007	000000800	800 MB RH ETA	%	1.0	34 64 0320	01 tt 00 00	x	x	x	x	x	x	x	x	x
003000007	000000850	850 MB RH ETA	%	1.0	34 64 0352	01 tt 00 00	x	x	x	x	x	x	x	x	x
003000007	000000900	900 MB RH ETA	%	1.0	34 64 0384	01 tt 00 00	x	x	x	x	x	x	x	x	x
003000007	000000950	950 MB RH ETA	%	1.0	34 64 03B6	01 tt 00 00	x	x	x	x	x	x	x	x	x
003000007	000001000	1000 MB RH ETA	%	1.0	34 64 03E8	01 tt 00 00	x	x	x	x	x	x	x	x	x
003007007	000000970	BDY LYR1 RH ETA	%	1.0	34 74 1D00	01 tt 00 00	x	x	x	x	x	x	x	x	x
003004007	000000273	273 K RH ETA	%	1.0	34 04 0000	01 tt 00 00	x	x	x	x	x	x	x	x	x
003210007	000000000	6-H TOTAL PRECIP ETA	mm	0.1	3D 01 0000	01 xx tt 04		x		x		x		x	
003220007	000000000	12-H TOTAL PRECIP ETA	mm	0.1	3D 01 0000	01 yy tt 04			x		x		x		x
003240007	000000000	6-H CONV PRECIP ETA	mm	0.1	3F 01 0000	01 xx tt 04		x		x		x		x	
003250007	000000000	12-H CONV PRECIP ETA	mm	0.1	3F 01 0000	01 yy tt 04			x		x		x		x
003350007	000000000	TOTAL P. WATER ETA	kg/m^2	0.1	3C 6C 0064	01 tt 00 00	x	x	x	x	x	x	x	x	x
004000007	000000150	150 mb U GRD ETA	m/s	0.1	21 64 0096	01 tt 00 00	x	x	x	x	x	x	x	x	x
004000007	000000200	200 mb U GRD ETA	m/s	0.1	21 64 00C8	01 tt 00 00	x	x	x	x	x	x	x	x	x
004000007	000000250	250 mb U GRD ETA	m/s	0.1	21 64 00FA	01 tt 00 00	x	x	x	x	x	x	x	x	x
004000007	000000300	300 MB U GRD ETA	m/s	0.1	21 64 012C	01 tt 00 00	x	x	x	x	x	x	x	x	x
004000007	000000400	400 MB U GRD ETA	m/s	0.1	21 64 0190	01 tt 00 00	x	x	x	x	x	x	x	x	x
004000007	000000500	500 MB U GRD ETA	m/s	0.1	21 64 01F4	01 tt 00 00	x	x	x	x	x	x	x	x	x
004000007	000000700	700 MB U GRD ETA	m/s	0.1	21 64 02BC	01 tt 00 00	x	x	x	x	x	x	x	x	x
004000007	000000750	750 MB U GRD ETA	m/s	0.1	21 64 02EE	01 tt 00 00	x	x	x	x	x	x	x	x	x
004000007	000000800	800 MB U GRD ETA	m/s	0.1	21 64 0320	01 tt 00 00	x	x	x	x	x	x	x	x	x
004000007	000000850	850 MB U GRD ETA	m/s	0.1	21 64 0352	01 tt 00 00	x	x	x	x	x	x	x	x	x
004000007	000000900	900 MB U GRD ETA	m/s	0.1	21 64 0384	01 tt 00 00	x	x	x	x	x	x	x	x	x
004000007	000000950	950 MB U GRD ETA	m/s	0.1	21 64 03B6	01 tt 00 00	x	x	x	x	x	x	x	x	x
004000007	000001000	1000 MB U GRD ETA	m/s	0.1	21 64 03E8	01 tt 00 00	x	x	x	x	x	x	x	x	x

004020007	000000010	10 M U GRD ETA	m/s	0.1	21	69	000A	01	tt	00	00	x	x	x	x	x	x	x	x	x
004040007	000000000	TROP U GRD ETA	m/s	0.1	21	07	0000	01	tt	00	00	x	x	x	x	x	x	x	x	x
004070007	000000970	B.L.(30 MB) U GRD ETA	m/s	0.1	21	74	1D00	01	tt	00	00	x	x	x	x	x	x	x	x	x
004100007	000000150	150 mb V GRD ETA	m/s	0.1	22	64	0096	01	tt	00	00	x	x	x	x	x	x	x	x	x
004100007	000000200	200 mb V GRD ETA	m/s	0.1	22	64	00C8	01	tt	00	00	x	x	x	x	x	x	x	x	x
004100007	000000250	250 mb V GRD ETA	m/s	0.1	22	64	00FA	01	tt	00	00	x	x	x	x	x	x	x	x	x
004100007	000000300	300 MB V GRD ETA	m/s	0.1	22	64	012C	01	tt	00	00	x	x	x	x	x	x	x	x	x
004100007	000000400	400 MB V GRD ETA	m/s	0.1	22	64	0190	01	tt	00	00	x	x	x	x	x	x	x	x	x
004100007	000000500	500 MB V GRD ETA	m/s	0.1	22	64	01F4	01	tt	00	00	x	x	x	x	x	x	x	x	x
004100007	000000700	700 MB V GRD ETA	m/s	0.1	22	64	02BC	01	tt	00	00	x	x	x	x	x	x	x	x	x
004100007	000000750	750 MB V GRD ETA	m/s	0.1	22	64	02EE	01	tt	00	00	x	x	x	x	x	x	x	x	x
004100007	000000800	800 MB V GRD ETA	m/s	0.1	22	64	0320	01	tt	00	00	x	x	x	x	x	x	x	x	x
004100007	000000850	850 MB V GRD ETA	m/s	0.1	22	64	0352	01	tt	00	00	x	x	x	x	x	x	x	x	x
004100007	000000900	900 MB V GRD ETA	m/s	0.1	22	64	0384	01	tt	00	00	x	x	x	x	x	x	x	x	x
004100007	000000950	950 MB V GRD ETA	m/s	0.1	22	64	03B6	01	tt	00	00	x	x	x	x	x	x	x	x	x
004100007	000001000	1000 MB V GRD ETA	m/s	0.1	22	64	03E8	01	tt	00	00	x	x	x	x	x	x	x	x	x
004120007	000000010	10 M V GRD ETA	m/s	0.1	22	69	000A	01	tt	00	00	x	x	x	x	x	x	x	x	x
004140007	000000000	TROP V GRD ETA	m/s	0.1	22	07	0000	01	tt	00	00	x	x	x	x	x	x	x	x	x
004170007	000000970	B.L.(30 MB) V GRD ETA	m/s	0.1	22	74	1D00	01	tt	00	00	x	x	x	x	x	x	x	x	x
004905007	000000000	TROP VW SH ETA	1/s	0.0001	88	07	0000	01	tt	00	00	x	x	x	x	x	x	x	x	x
005000007	000000150	150 mb VV ETA	Pa/s	0.001	27	64	0096	01	tt	00	00	x	x	x	x	x	x	x	x	x
005000007	000000200	200 mb VV ETA	Pa/s	0.001	27	64	00C8	01	tt	00	00	x	x	x	x	x	x	x	x	x
005000007	000000250	250 mb VV ETA	Pa/s	0.001	27	64	00FA	01	tt	00	00	x	x	x	x	x	x	x	x	x
005000007	000000300	300 MB VV ETA	Pa/s	0.001	27	64	012C	01	tt	00	00	x	x	x	x	x	x	x	x	x
005000007	000000400	400 MB VV ETA	Pa/s	0.001	27	64	0190	01	tt	00	00	x	x	x	x	x	x	x	x	x
005000007	000000500	500 MB VV ETA	Pa/s	0.001	27	64	01F4	01	tt	00	00	x	x	x	x	x	x	x	x	x
005000007	000000700	700 MB VV ETA	Pa/s	0.001	27	64	02BC	01	tt	00	00	x	x	x	x	x	x	x	x	x
005000007	000000750	750 MB VV ETA	Pa/s	0.001	27	64	02EE	01	tt	00	00	x	x	x	x	x	x	x	x	x
005000007	000000800	800 MB VV ETA	Pa/s	0.001	27	64	0320	01	tt	00	00	x	x	x	x	x	x	x	x	x
005000007	000000850	850 MB VV ETA	Pa/s	0.001	27	64	0352	01	tt	00	00	x	x	x	x	x	x	x	x	x
005000007	000000900	900 MB VV ETA	Pa/s	0.001	27	64	0384	01	tt	00	00	x	x	x	x	x	x	x	x	x
005000007	000000950	950 MB VV ETA	Pa/s	0.001	27	64	03B6	01	tt	00	00	x	x	x	x	x	x	x	x	x
005000007	000001000	1000 MB VV ETA	Pa/s	0.001	27	64	03E8	01	tt	00	00	x	x	x	x	x	x	x	x	x
005007007	000000970	BDY LYR1 V VEL ETA	Pa/s	0.001	27	74	1D00	01	tt	00	00	x	x	x	x	x	x	x	x	x
007007007	000000000	BEST LI(6BDY) ETA	K	0.1	84	74	8400	01	tt	00	00	x	x	x	x	x	x	x	x	x
007100007	000000000	SFC CAPE ETA	J/Kg	1.0	9D	01	00 00	01	tt	00	00	x	x	x	x	x	x	x	x	x

D. Aviation Model Archive

Description of Grid and Variables Archived

Initial and forecast fields from NCEP's Aviation (AVN) run of the Global Spectral Model (GSM) are saved daily from the 0000, 0600, 1200, and 1800 UTC forecast cycles on the grid shown in Fig. 3. This grid corresponds to a polar stereographic projection oriented 105° W with a grid resolution of 95.25 km at 60° N. The archive consists of data at 11,097 grid points, with 137 points in the x-direction (from left to right in the diagram, or approximately west to east over the contiguous U.S.) and 81 points in the y-direction (from bottom to top in the diagram, or approximately south to north over the contiguous U.S.). The data are packed in TDLPACK format as described in Chapter 5 by subroutine PACK2D beginning at the lower left corner gridpoint (defined as x=1, y=1). The data are returned by the subroutine UNPACK starting at the lower left, proceeding left to right (in order of increasing x), then bottom to top (in order of increasing y).

The data archived are extracted from NCEP files (see below) which contain the data on a 1 degree latitude/longitude grid. Note that these NCEP files contain the data in GRIB format. NCEP software is used to convert the data from the latitude/longitude grid to the rectangular grid shown in Fig. 3. While the horizontal resolution of this rectangular grid approximately corresponds to the resolution of the operational AVN model, the AVN computational grid has varied during the history of the archive (see below for further details).

Table 12-3 lists the MOS-2000 identifier for each of the fields archived, the corresponding GRIB identifier of the field in the NCEP files, the precision of the data, and the units in which the variables were saved. Note that standard GRIB units are used in archiving all the variables.

Archive Process

At the end of the 0000, 0600, 1200, and 1800 UTC AVN forecast cycles, a job is run on the Cray to access the NCEP AVN GRIB files, interpolate the data to the rectangular grid shown in Fig. 3, read the fields of interest, extract the data for the area shown in Fig. 3, and put the data into TDLPACK format.[3] The TDLPACK data are then stored for each day and cycle in a set of daily rotating files. Once a week, the daily files are concatenated and stored on both disk and tape. After a month's data are collected, the disk and tape files are complete. The disk files are not retained for an extended period of time, but are removed as disk space is required for more current data. The tape files are structured so that a volume-set contains for one cycle a year of data divided into 12 files corresponding to the 12 months of the year. Thus, unlike some of the other archives, a separate volume set is available for each of the four forecast cycles. Note that the entire archive process and the storage of the data are done on the Cray. Note, also, that all of these files are written in binary format with standard Unix blocking (that is, the data are not blocked according to the Cray Operating System, but are, in fact, non-

[3]NCEP is in the process of converting from the CRAYs to an IBM mainframe. This writeup reflects the current situation.

COS blocked) so that the data can be read on either the Cray main-frame or the HP workstation platform.

File Names

The following files are used in the archive process described above:

NCEP GRIB files: /com/avn/prod/avn.**yyyymmdd**/gblav.T**hh**Z.PGrbF**xx**, where **yyyy** equals the two digit year, **mm** equals the two digit month, **dd** equals the two digit day of the month, **hh** equals the cycle (00, 06, 12, or 18), and **xx** equals the forecast projection (00, 03, 06, 09, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57, 60, 63, 66, 69, 72, 75, and 78).

Daily rotating file: /tdl4/we21ar/rotate/a**hh**/**yyyymmddhh**, where **yyyy** equals the four digit year, **mm** equals the two digit month, **dd** equals the two digit day of the month, and **hh** equals the cycle (00, 06, 12, or 18).

Monthly file: /tdl4/we21ar/archive/a**hh**/a**hhyyyymm**, where **hh** equals the cycle (00, 06, 12, or 18), **yyyy** equals the four digit year, and **mm** equals the two digit month. This file is stored on disk; a copy of the file is also stored on tape as described in the yearly file below.

Yearly file: The volume set name of the tape is designated as: tdla**hhyyyys**, where **hh** corresponds to the cycle (00, 06, 12, or 18), **yyyy** corresponds to the four digit year and **s** corresponds to the tape status, that is, a **p** for the primary tape set and a **b** for the backup set. Each monthly file within the yearly volume set is designated as: A**hhyyyymm**, where **hh** equals the cycle (00, 06, 12, or 18), **yyyy** equals the four digit year, and **mm** equals the two digit month.

Archive Tapes

As indicated above, the archived Aviation model data are eventually stored on cartridge tapes and are listed in the Cray Reel Librarian (CRL) system. That database system can be queried by the Cray user to determine the names and availability of the various volume sets. Generally, the tapes are stored in a separate room from the Cray, and a request must be made to the tape librarian to bring the tapes back to the Cray silos in order to be used. This retrieval of the tapes nominally takes 24 hours; in reality, the time required can take more than a week. Once the tapes are placed back into the silo, the user can copy the data from tape to disk and then proceed with a specific task. A file on the Cray named: /jdsk41/we/tdllib/log/tapes/we21ar.list gives the names of the Aviation files. However, this list is not necessarily current so that the interested user should query the CRL system. Use the "man" facility for the command "rlr" for further details on querying CRL.

As of November 1998, all of the archived Aviation model data have been moved back from tape to the Cray4 /dm (data migration) disk system for purposes of development. These data are stored in the following file system: /dm/we21ar/modeldata/a**hh**/a**hhyyyymm**, where **hh** denotes the cycle (00, 06, 12, or 18), **yyyy** equals the four digit year, and **mm** equals the two digit month. Data are available from April 1997 to the present in the /dm system on Cray4. However, the user is reminded to check the status of these files by issuing the "ls -l" command. If the files have been migrated to the silo tape system, the user can retrieve the files within a minute or so by issuing a "dmget" command. Alternatively, the submission of a batch job to Cray4 which requires the /dm files will force the system to retrieve the files back to disk.

Period of Record

The AVN archive begins with 0000 UTC data on April 1, 1997, and continues to the present. The archive was initiated on the Cray in mid-July 1997; data prior to that time were obtained from NCEP's run-history archive. Tables 12-3 and 12-4 describe the variables and projections available in the archive.

Information Resources

Various versions of the global spectral model have been operational since 1980. The interested user may wish to look at the following:

http://nic.fb4.noaa.gov:8000/research/mrf.html for information about the model status as well as appropriate references. Unfortunately, at the time this chapter was written, this page was not being kept current and so only included model changes through October 1995. The reader might also want to look at NWS Technical Procedures Bulletins Nos. 442 and 443 for more current information concerning the GSM.

Two additional sources can be checked:

http://nic.fb4.noaa.gov:8000/research/global2.html - contains information about NCEP's Global Modeling Branch as well as links to GSM information;

http://sgi62.wwb.noaa.gov:8080/web2/homepage.html - contains documentation on the GSM that is currently being revised.

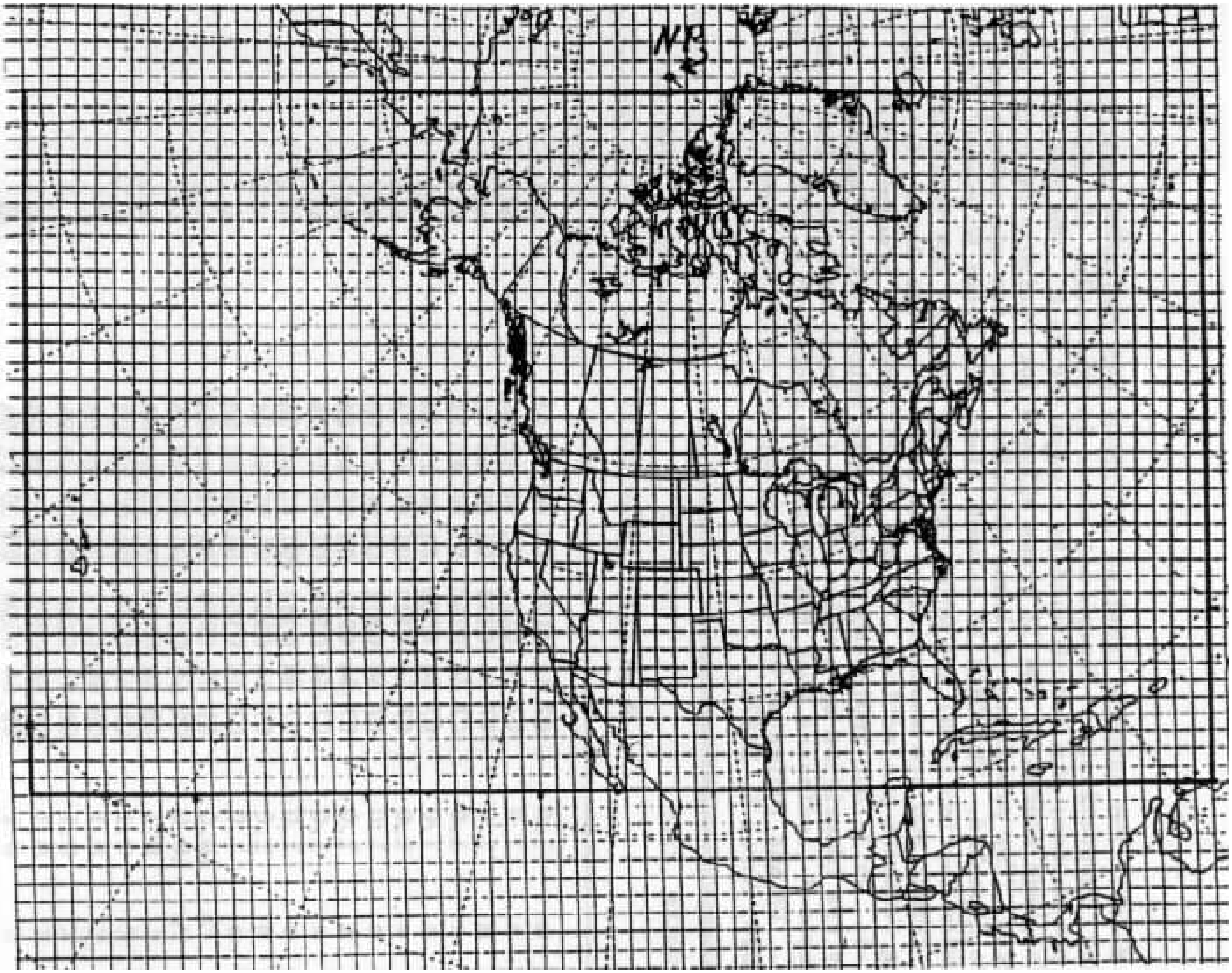

Figure 12-3. 137 X 81 grid on which AVN data are archived. The upper right point is (137,81) with reference to the lower left (1,1).

Table 12-3. Fields archived from the AVN Model. The columns labelled CCCFFFBDD and VLLLLUUUU indicate the first two words of the MOS-2000 identifier, respectively. Though not shown in the table, the third word of the identifier contains the forecast projection; the fourth word is always set to 0. The column labeled Plain Language gives the plain language stored in the 32 bytes allowed for variable description in the TDLPACK format. The Units and the Prec. (for precision) columns describe the units and subsequent precision of the field after the data are unpacked. The GRIB ID columns lists the bytes by number which are contained in the GRIB Product Definition Section (PDS) and which precisely specify the variable. These values are included in hexadecimal format and are essential to the archive of the data. Note that the value of tt corresponds to the ending forecast projection. In the identifiers for precipitation amount, xx equals tt-3, and yy equals tt-6. While not indicated in the table below, all variables in the Eta archive currently created on the Cray are identified by a value of 129 (decimal) in byte 6 (indicating the GSM model) and by a value of 255 in byte 7 (indicates that the values are stored on a grid that is not one of the standard NCEP or AWIPS girds). Unless indicated by superscripts, all variables have been available since the archive began (0000 UTC, April 1, 1997).

CCCFFFBDD	VLLLLUUUU	Plain Language	Units	Prec.	GRIB ID							Forecast Projection[3]
					9	10	11/12	18	19	20	21	0 to 78 at 3-hr intervals
001000008	000000300	300 MB HGT AVN	m	1.0	07	64	012C	01	00	tt	0A	All
001000008	000000500	500 MB HGT AVN	m	1.0	07	64	01F4	01	00	tt	0A	All
001000008	000000600	600 MB HGT AVN[1]	m	1.0	07	64	0258	01	00	tt	0A	All
001000008	000000700	700 MB HGT AVN	m	1.0	07	64	02BC	01	00	tt	0A	All
001000008	000000750	750 MB HGT AVN[1]	m	1.0	07	64	02EE	01	00	tt	0A	All
001000008	000000800	800 MB HGT AVN[1]	m	1.0	07	64	0320	01	00	tt	0A	All
001000008	000000850	850 MB HGT AVN	m	1.0	07	64	0352	01	00	tt	0A	All
001000008	000000900	900 MB HGT AVN[1]	m	1.0	07	64	0384	01	00	tt	0A	All
001000008	000000925	925 MB HGT AVN	m	1.0	07	64	039D	01	00	tt	0A	All
001000008	000000950	950 MB HGT AVN[1]	m	1.0	07	64	03B6	01	00	tt	0A	All
001000008	000000975	975 MB HGT AVN[1]	m	1.0	07	64	03CF	01	00	tt	0A	All
001000008	000001000	1000 MB HGT AVN	m	1.0	07	64	03E8	01	00	tt	0A	All
001100008	000000000	SFC PRES AVN	Pa	10.0	01	01	0000	01	00	tt	0A	All
001200008	000000000	MSL(STND) PRES AVN	Pa	10.0	02	66	0000	01	00	tt	0A	All
002000008	000000300	300 MB TEMP AVN	K	0.1	0B	64	012C	01	00	tt	0A	All
002000008	000000500	500 MB TEMP AVN	K	0.1	0B	64	01F4	01	00	tt	0A	All
002000008	000000600	600 MB TEMP AVN[1]	K	0.1	0B	64	0258	01	00	tt	0A	All
002000008	000000700	700 MB TEMP AVN	K	0.1	0B	64	02BC	01	00	tt	0A	All
002000008	000000750	750 MB TEMP AVN[1]	K	0.1	0B	64	02EE	01	00	tt	0A	All
002000008	000000800	800 MB TEMP AVN[1]	K	0.1	0B	64	0320	01	00	tt	0A	All
002000008	000000850	850 MB TEMP AVN	K	0.1	0B	64	0352	01	00	tt	0A	All

002000008	000000900	900 MB TEMP AVN[1]	K	0.1	0B 64 0384	01 00 tt 0A	All
002000008	000000925	925 MB TEMP AVN	K	0.1	0B 64 039D	01 00 tt 0A	All
002000008	000000950	950 MB TEMP AVN[1]	K	0.1	0B 64 03B6	01 00 tt 0A	All
002000008	000000975	975 MB TEMP AVN[1]	K	0.1	0B 64 03CF	01 00 tt 0A	All
002000008	000001000	1000 MB TEMP AVN	K	0.1	0B 64 03E8	01 00 tt 0A	All
002001008	000000002	2 M TEMP AVN	K	0.1	0B 69 0002	01 00 tt 0A	All
002006008	000000995	.995 SIGMA TEMP AVN[2]	K	0.1	0B 6B26DE	01 00 tt 0A	All
002007008	000000970	B.L.(30 MB) TEMP AVN	K	0.1	0B 74 1E00	01 00 tt 0A	All
002011008	000000002	6-H MAX TEMP AVN	K	0.1	0F 69 0002	01 yy tt 02	tt = 6, 12, 18, etc.
002021008	000000002	6-H MIN TEMP AVN	K	0.1	10 69 0002	01 yy tt 02	tt = 6, 12, 18, etc.
002031008	000000002	3-H MAX TEMP AVN	K	0.1	0F 69 0002	01 xx tt 02	tt = 3, 9, 15, etc.
002041008	000000002	3-H MIN TEMP AVN	K	0.1	10 69 0002	01 xx tt 02	tt = 3, 9, 15, etc.
003000008	000000300	300 MB RH AVN	%	1.0	34 64 012C	01 00 tt 0A	All
003000008	000000500	500 MB RH AVN	%	1.0	34 64 01F4	01 00 tt 0A	All
003000008	000000600	600 MB RH AVN[1]	%	1.0	34 64 0258	01 00 tt 0A	All
003000008	000000700	700 MB RH AVN	%	1.0	34 64 02BC	01 00 tt 0A	All
003000008	000000750	750 MB RH AVN[1]	%	1.0	34 64 02EE	01 00 tt 0A	All
003000008	000000800	800 MB RH AVN[1]	%	1.0	34 64 0320	01 00 tt 0A	All
003000008	000000850	850 MB RH AVN	%	1.0	34 64 0352	01 00 tt 0A	All
003000008	000000900	900 MB RH AVN[1]	%	1.0	34 64 0384	01 00 tt 0A	All
003000008	000000925	925 MB RH AVN	%	1.0	34 64 039D	01 00 tt 0A	All
003000008	000000950	950 MB RH AVN[1]	%	1.0	34 64 03B6	01 00 tt 0A	All
003000008	000000975	975 MB RH AVN[1]	%	1.0	34 64 03CF	01 00 tt 0A	All
003000008	000001000	1000 MB RH AVN	%	1.0	34 64 03E8	01 00 tt 0A	All
003001008	000000002	2 M RH AVN	%	1.0	34 69 0002	01 00 tt 0A	All
003002008	000000440	MEAN RH(1.-.44SGY) AVN	%	1.0	34 6C 2C64	01 00 tt 0A	All
003006008	000000995	.995 SIGMA RH AVN[2]	%	1.0	34 6B 26DE	01 00 tt 0A	All
003007008	000000970	B.L.(30 MB) RH AVN	2%	1.0	34 74 1E00	01 00 tt 0A	All
003031008	000000002	2 M SPF HUM AVN[2]	kg/kg	0.00001	33 69 0002	01 00 tt 0A	All
003205008	000000000	3-H TOTAL PRECIP AVN	mm	0.1	3D 01 0000	01 xx tt 04	tt = 3, 9, 15, etc.
003210008	000000000	6-H TOTAL PRECIP AVN	mm	0.1	3D 01 0000	01 yy tt 04	tt = 6, 12, 18, etc.
003235008	000000000	3-H CONV PRECIP AVN	mm	0.1	3F 01 0000	01 xx tt 04	tt = 3, 9, 15, etc.
003240008	000000000	6-H CONV PRECIP AVN	mm	0.1	3F 01 0000	01 yy tt 04	tt = 6, 12, 18, etc.
003350008	000000000	TOTAL P. WATER AVN	kg/m^2	0.1	36 C8 0000	01 00 tt 0A	All
004000008	000000300	300 MB U GRD AVN	m/s	0.1	21 64 012C	01 00 tt 0A	All
004000008	000000500	500 MB U GRD AVN	m/s	0.1	21 64 01F4	01 00 tt 0A	All
004000008	000000600	600 MB U GRD AVN[1]	m/s	0.1	21 64 0258	01 00 tt 0A	All
004000008	000000700	700 MB U GRD AVN	m/s	0.1	21 64 02BC	01 00 tt 0A	All
004000008	000000750	750 MB U GRD AVN[1]	m/s	0.1	21 64 02EE	01 00 tt 0A	All
004000008	000000800	800 MB U GRD AVN[1]	m/s	0.1	21 64 032	01 00 tt 0A	All
004000008	000000850	850 MB U GRD AVN	m/s	0.1	21 64 0352	01 00 tt 0A	All
004000008	000000900	900 MB U GRD AVN[1]	m/s	0.1	21 64 0384	01 00 tt 0A	All

004000008	000000925	925 MB U GRD AVN	m/s	0.1	21 64 039D	01 00 tt 0A	All
004000008	000000950	950 MB U GRD AVN[1]	m/s	0.1	21 64 03B6	01 00 tt 0A	All
004000008	000000975	975 MB U GRD AVN[1]	m/s	0.1	21 64 03CF	01 00 tt 0A	All
004000008	000001000	1000 MB U GRD AVN	m/s	0.1	21 64 03E8	01 00 tt 0A	All
004020008	000000010	10 M U GRD AVN	m/s	0.1	21 69 000A	01 00 tt 0A	All
004040008	000000000	TROP U GRD AVN	m/s	0.1	21 07 0000	01 00 tt 0A	All
004905008	000000000	TROP VW SH AVN[2]	1/s	0.0001	88 07 0000	01 00 tt 0A	All
004100008	000000300	300 MB V GRD AVN	m/s	0.1	22 64 012C	01 00 tt 0A	All
004100008	000000500	500 MB V GRD AVN	m/s	0.1	22 64 01F4	01 00 tt 0A	All
004100008	000000600	600 MB V GRD AVN[1]	m/s	0.1	22 64 0258	01 00 tt 0A	All
004100008	000000700	700 MB V GRD AVN	m/s	0.1	22 64 02BC	01 00 tt 0A	All
004100008	000000750	750 MB V GRD AVN[1]	m/s	0.1	22 64 02EE	01 00 tt 0A	All
004100008	000000800	800 MB V GRD AVN[1]	m/s	0.1	22 64 0320	01 00 tt 0A	All
004100008	000000850	850 MB V GRD AVN	m/s	0.1	22 64 0352	01 00 tt 0A	All
004100008	000000900	900 MB V GRD AVN[1]	m/s	0.1	22 64 0384	01 00 tt 0A	All
004100008	000000925	925 MB V GRD AVN	m/s	0.1	22 64 039D	01 00 tt 0A	All
004100008	000000950	950 MB V GRD AVN[1]	m/s	0.1	22 64 03B6	01 00 tt 0A	All
004100008	000000975	975 MB V GRD AVN[1]	m/s	0.1	22 64 03CF	01 00 tt 0A	All
004100008	000001000	1000 MB V GRD AVN	m/s	0.1	22 64 03E8	01 00 tt 0A	All
004120008	000000010	10 M V GRD AVN	m/s	0.1	22 69 000A	01 00 tt 0A	All
004140008	000000000	TROP V GRD AVN	m/s	0.1	22 07 0000	01 00 tt 0A	All
005000008	000000300	300 MB VV AVN	Pa/s	0.001	27 64 012C	01 00 tt 0A	All
005000008	000000500	500 MB VV AVN	Pa/s	0.001	27 64 01F4	01 00 tt 0A	All
005000008	000000600	600 MB VV AVN[1]	Pa/s	0.001	27 64 0258	01 00 tt 0A	All
005000008	000000700	700 MB VV AVN	Pa/s	0.001	27 64 02BC	01 00 tt 0A	All
005000008	000000750	750 MB VV AVN[1]	Pa/s	0.001	27 64 02EE	01 00 tt 0A	All
005000008	000000800	800 MB VV AVN[1]	Pa/s	0.001	27 64 0320	01 00 tt 0A	All
005000008	000000850	850 MB VV AVN	Pa/s	0.001	27 64 0352	01 00 tt 0A	All
005000008	000000900	900 MB VV AVN[1]	Pa/s	0.001	27 64 0384	01 00 tt 0A	All
005000008	000000925	925 MB VV AVN	Pa/s	0.001	27 64 039D	01 00 tt 0A	All
005000008	000000950	950 MB VV AVN[1]	Pa/s	0.001	27 64 03B6	01 00 tt 0A	All
005000008	000000975	975 MB VV AVN[1]	Pa/s	0.001	27 64 03CF	01 00 tt 0A	All
005000008	000001000	1000 MB VV AVN	Pa/s	0.001	27 64 03E8	01 00 tt 0A	All
007002008	000000000	BEST LI(4BDY) AVN	K	0.1	84 01 0000	01 00 tt 0A	All
007010008	000000000	SFC LI AVN	K	0.1	83 01 0000	01 00 tt 0A	All
007100008	000000000	SFC CAPE AVN	J/kg	1.0	9D 01 0000	01 00 tt 0A	All
007120008	000000000	SFC CNV INHB AVN	J/kg	1.0	9C 01 0000	01 00 tt 0A	All

[1] These variables were added effective 0000 UTC, October 1, 1998.
[2] These variables were removed effective 0000 UTC, October 1, 1998.
[3] Projections 75- and 78-hours were added effective 0000 UTC, October 1, 1998.

E. Medium-Range Forecast Model Archive

Description of Grid and Variables Archived

Initial and forecast fields from NCEP's Medium-Range Forecast (MRF) run of the Global Spectral Model (GSM) are saved daily from the 0000 UTC forecast cycle over the area encompassed by the grid shown in Fig. 12-3. As discussed later in this section, two different grid resolutions are used for archiving the MRF data, corresponding approximately to the resolution of the model used to generate the forecasts. The fine resolution grid has a resolution of 95.25 km at 60° N. The coarse resolution grid has a resolution of 190.50 km at 60° N. Both grids cover the area shown in Fig. 12-3 and are polar stereographic projections oriented 105° W with the aforementioned grid resolutions. Figure 12-3 shows the grid points for the coarse resolution grid; the fine resolution grid can be obtained by adding another row and another column line midway between each row and column, that is, halving the distance between each row and each column. The MRF fine resolution archive consists of data at 11,097 grid points, with 137 points in the x-direction (from left to right in the diagram, or approximately west to east over the contiguous U.S.) and 81 points in the y-direction (from bottom to top in the diagram). The coarse resolution archive consists of data at 2760 grid points, with 69 points in the x-direction (from left to right in the diagram) and 40 points in the y-direction (from bottom to top in the diagram).

Prior to April 1, 1997, all MRF archive data were saved on the coarse resolution grid because the data were generated by the T62 version of the GSM (see below). Beginning with data archived for April 1, 1997, MRF data for projections out to 192 hours were saved on the fine resolution grid; for forecast projections beyond 192 hours, data were saved on the coarse resolution grid. This approach to the data archive corresponds approximately to the resolution of the GSM used in the MRF (as of January 1999, T126 out to 168 hours and T62 beyond that projection to 384 hours). All archive data are packed in TDLPACK format as described in Chapter 5 by subroutine PACK2D beginning at the lower left-hand corner grid point (defined as x=1, y=1) and moving in order of increasing x value when y is odd-valued and in order of decreasing x value when y is even-valued. The rows are processed in order of increasing y value (from bottom to top). The data are unpacked by subroutine UNPACK and stored in an array starting at the lower left-hand corner, proceeding left to right (in order of increasing x), and then bottom to top (in order of increasing y). The data archived are extracted from NCEP files (see below) which contain the data on a 1 degree latitude/longitude grid for projections out to 168 hours and on a 2.5 degree latitude/longitude grid for projections beyond 168 hours. Note that these NCEP files contain the data in GRIB format. NCEP software (program copygb) is used to convert the data from the latitude/longitude grids to the appropriate rectangular grid discussed above. Different interpolation schemes are used for non-precipitation and precipitation variables. Limitations in the GRIB structure hinder identification of precipitation amount forecasts beyond 252 hours. To circumvent this problem for projections beyond 252 hours, TDL software is used to convert precipitation rate forecasts to a forecast of precipitation accumulation over a 12-h period. While the horizontal resolution of the archive grids approximately corresponds to the resolution of the operational MRF model, the MRF computational grid has varied during the history of the archive (see references below for further details).

Table 12-4 lists the MOS-2000 identifier for each of the fields archived, the corresponding GRIB identifier of the field in the NCEP files, the precision of the data, and the units in which the variables were saved. Note that standard GRIB units are used in archiving all the variables.

Archive Process

At the end of the 0000 UTC MRF forecast cycle, a job is run on the Cray to access the NCEP MRF GRIB files, interpolate the data to the appropriate rectangular grid discussed above, read the fields of interest, extract the data for the area shown in Fig. 12-3, and put the data into TDLPACK format.[4] The TDLPACK data are then stored for each day in a daily file. Once a week, the daily files are concatenated and stored on both disk and tape. After a month's data are collected, the disk and tape files are complete. The disk files are not retained for an extended period of time, but are removed as disk space is required for more current data. The tape files are structured so that a volume-set contains a year of data divided into 12 files corresponding to the 12 months of the year. This convention was initiated for data from April 1997 to the present. Data prior to April 1997 are not currently on tape, but are archived to two different disk systems structured so that a year of data resides in a single file with no monthly subdivision (see below for further details). Note that the entire archive process and the storage of the data are done on the Cray. Note, also, that all of these files are written in binary format with standard Unix blocking (that is, the data are not blocked according to the Cray Operating System, but are, in fact, non-COS blocked) so that the data can be read on either the Cray main-frame or the HP workstation platform.

File Names

The following files are used in the archive process described above:

NCEP GRIB files: /com/mrf/prod/mrf.**yyyymmdd**/drfmr.T**hh**Z.PGrbF**xx**, where **yyyy** equals the four digit year, **mm** equals the two digit month, **dd** equals the two digit day of the month, **hh** equals the cycle (00), and **xx** equals the forecast projection (00, 12, 24, 36, 48, 60, 72, 84, 96, 108, 120, 132, 144, 156, 168, 180, 192, 204, 216, 228, 240, 252, 264, 276, 288, 300, 312, 324, 336, 348, 360, 372, and 384).

Daily rotating file: /tdl4/we21ar/rotate/mrx/**yyyymmddhh**, where **yyyy** equals the four digit year, **mm** equals the two digit month, **dd** equals the two digit day of the month, and **hh** equals the cycle (00).

Monthly file: /tdl4/we21ar/archive/mrx/mrx**yyyymm**, where **yyyy** equals the four digit year, and **mm** equals the two digit month. This file is stored on disk; a copy of the file is also stored on tape as described in the yearly file below.

Yearly file: The volume set name of the tape is designated as: tdlmrx**yyyys**, where **yyyy** corresponds to the four digit year and **s** corresponds to the tape status, that is, a **p** for the primary tape set and a **b** for the backup set. Each monthly file within the yearly volume set is designated as: MRX**yyyymm**, where **yyyy** equals the four digit year and **mm** equals the two digit month. Note that only data subsequent to March 1997 are archived on tape.

Yearly files prior to April 1997: Files are stored on Cray5 in the /dm (data migration) system with the following naming convention:

---

[4]NCEP is in the process of converting from the CRAYs to an IBM mainframe. This writeup reflects the current situation.

/dm/we21ar/rotate/reanl/**yyyy**, where **yyyy** equals the four digit year. These files contain the data for an entire year. Unlike the data subsequent to March 1997, only projections to 192 hours are available (see below for further details).

Archive Tapes

The archived MRF model data are eventually stored on cartridge tapes and are listed in the Cray Reel Librarian (CRL) system. That database system can be queried by the Cray user to determine the names and availability of the various volume sets. Generally, the tapes are stored in a separate room from the Cray, and a request must be made to the tape librarian to bring the tapes back to the Cray silos in order to be used. This retrieval of the database nominally takes 24 hours; in reality, the time required can take more than a week. Once the tapes are placed back into the silo, the user can copy the data from tape to disk and then proceed with a specific task. A file on the Cray named: /jdsk41/we/tdllib/log/tapes/we21ar.list gives the names of the MRF model files. However, this list is not necessarily current so that the interested user should query the CRL system. Use the "man" facility for the command "rlr" for further details on querying CRL. As mentioned previously, only data from April 1997 to the present are currently on cartridge tape.

As of November 1998, all of the archived MRF model data have been moved back from tape to the Cray4 /dm (data migration) disk system for purposes of development. These data are stored in the following file systems: /dm/we21ar/modeldata/mrx/mrx**yyyymm_nn**, (data from Jan. 1992 - March 1997); /dm/we21ar/modeldata/mrx/mrx**yyyymm**, (data from April 1997 to the present) where **yyyy** equals the four digit year, **mm** equals the two digit month for the beginning of the period of record, and **nn** equals the two digit month for the ending of the period of record. Thus, for example, data from 1992 are stored in a file labeled mrx199201_12. Data from April 1998 are stored in a file labeled mrx199804 and so forth. The user is encouraged to use the “ls” command to determine the names of the various files. Data are available from January 1992 to the present in the /dm system on Cray4; thus, in fact, the data for the period of January 1992 through March 1997 are located in the /dm system on both Crays. However, the user is reminded to check the status of these files by issuing the "ls -l" command. If the files have been migrated to the silo tape system, the user can retrieve the files within a minute or so by issuing a "dmget" command. Alternatively, the submission of a batch job to Cray4 which requires the /dm files will force the system to retrieve the files back to disk.

Period of Record

The MRF archive begins with 0000 UTC data on January 1, 1992, and continues to the present. The archive was initiated on the Cray in mid-September 1997; data prior to that time but after March 31, 1997, were obtained from NCEP's run-history archive. Data from January 1, 1992, through March 31, 1997, were obtained from NCEP’s re-analysis archive. Note that for this latter period, forecast data were only available from initial conditions on January 1, January 6, January 11, and so forth, that is, every fifth day through the year. Also note that these data were produced from the T62 model, were unavailable for the initial time (hour 00), and were only generated for forecast projections out to 192 hours. Because the T62 version of the MRF model was used in producing the forecasts, the TDL archive for the period of January 1992 through March 1997 is available on the coarse resolution grid described above. Table 12-4 describes the variables available in the archive.

Information Resources

Various versions of the global spectral model have been operational since 1980. The interested user may wish to look at the following:

http://nic.fb4.noaa.gov:8000/research/mrf.html for information about the model status as well as appropriate references. Unfortunately, at the time this chapter was written, this page was not being kept current and so only included model changes through October 1995. The reader might also want to look at NWS Technical Procedures Bulletins Nos. 442 and 443 for more current information concerning the GSM.

Two additional sources can be checked:

http://nic.fb4.noaa.gov:8000/research/global2.html - contains information about NCEP's Global Modeling Branch as well as links to GSM information;

http://sgi62.wwb.noaa.gov:8080/web2/homepage.html - contains documentation on the GSM that is currently being revised.

Table 12-4. Fields archived from the MRF run of the GSM. The columns labelled CCCFFFBDD and VLLLLUUUU indicate the first two words of the MOS-2000 identifier, respectively. Though not shown in the table, the third word of the identifier contains the forecast projection; the fourth word is always set to 0. The column labeled PLAIN LANGUAGE gives the plain language stored in the 32 bytes allowed for variable description in the TDLPACK format. The UNITS and the PRECISION columns describe the units and subsequent precision of the field after the data are unpacked. Note that the precision shown here is simply the inverse of the scaling factor used to pack the data. The GRIB ID columns lists the bytes by number which are contained in the GRIB Product Definition Section (PDS) and which precisely specify the variable. These values are included in hexidecimal format and are essential to the archive of the data. Note that the value of tt corresponds to the ending forecast projection and yy equals tt-12. While not indicated in the table below, all variables in the MRF archive currently created on the Cray are identified by a value of 82 (decimal) in byte 6 for analysis data, 78 in byte 6 for forecasts to 168 hours (identifying the T126 model), 80 in byte 6 for forecasts beyond 168 hours (identifying the T62 model), and by a value of 255 in byte 7 (indicates that the values are stored on a grid that is not one of the standard NCEP or AWIPS grids). Unless indicated by superscripts, for data archived from April 1997 to the present, all variables are available at projections every 12 hours from 0 to 384 hours after 0000 UTC. For data prior to April 1997, the variables are available at projections every 12 hours from 12 to 192 hours after 0000 UTC.

CCCFFFBDD	VLLLLUUUU	Plain Language	Units	Prec.	GRIB ID							Forecast Projection[3]
					9	10	11/12	18	19	20	21	0-384 at 12-hr intervals
001000009	000000300	300 MB HGT MRF	m	1.0	07	64	012C	01	00	tt	0A	all
001000009	000000500	500 MB HGT MRF	m	1.0	07	64	01F4	01	00	tt	0A	all
001000009	000000700	700 MB HGT MRF	m	1.0	07	64	02BC	01	00	tt	0A	all
001000009	000000850	850 MB HGT MRF	m	1.0	07	64	0352	01	00	tt	0A	all
001000009	000000925	925 MB HGT MRF	m	1.0	07	64	039D	01	00	tt	0A	all
001000009	000001000	1000 MB HGT MRF	m	1.0	07	64	03E8	01	00	tt	0A	all
001100009	000000000	SFC PRES MRF	Pa	10.0	01	01	0000	01	00	tt	0A	all
001200009	000000000	MSL(STND) PRES MRF	Pa	10.0	02	66	0000	01	00	tt	0A	all
002000009	000000300	300 MB TEMP MRF	K	0.1	0B	64	012C	01	00	tt	0A	all
002000009	000000500	500 MB TEMP MRF	K	0.1	0B	64	01F4	01	00	tt	0A	all
002000009	000000700	700 MB TEMP MRF	K	0.1	0B	64	02BC	01	00	tt	0A	all
002000009	000000850	850 MB TEMP MRF	K	0.1	0B	64	0352	01	00	tt	0A	all
002000009	000000925	925 MB TEMP MRF	K	0.1	0B	64	039D	01	00	tt	0A	all
002000009	000001000	1000 MB TEMP MRF	K	0.1	0B	64	03E8	01	00	tt	0A	all
002001009	000000002	2 M TEMP MRF	K	0.1	0B	69	0002	01	00	tt	0A	all
002007009	000000970	B.L.(30 MB) TEMP MRF	K	0.1	0B	74	1E00	01	00	tt	0A	all
002051009	000000002	12-H MAX TEMP MRF[1]	K	0.1	0F	69	0002	01	00	tt	02	all
002061009	000000002	12-H MIN TEMP MRF[1]	K	0.1	10	69	0002	01	00	tt	02	all
003000009	000000300	300 MB RH MRF	%	1.0	34	64	012C	01	00	tt	0A	all

003000009	000000500	500 MB RH MRF	%	1.0	34 64 01F4	01 00 tt 0A	all
003000009	000000700	700 MB RH MRF	%	1.0	34 64 02BC	01 00 tt 0A	all
003000009	000000850	850 MB RH MRF	%	1.0	34 64 0352	01 00 tt 0A	all
003000009	000000925	925 MB RH MRF	%	1.0	34 64 039D	01 00 tt 0A	all
003000009	000001000	1000 MB RH MRF	%	1.0	34 64 03E8	01 00 tt 0A	all
003001009	000000002	2 M RH MRF[2]	%	1.0	34 69 0002	01 00 tt 0A	all
003002009	000000440	MEAN RH(1.-.44SGY) MRF	%	1.0	34 6C 2C64	01 00 tt 0A	all
003007009	000000970	B.L.(30 MB) RH MRF	%	1.0	34 74 1E00	01 00 tt 0A	all
003220009	000000000	12-H TOTAL PRECIP MRF[3]	mm	0.1	3D 01 0000	01 yy tt 04	all except 00
003250009	000000000	12-H CONV PRECIP MRF[3]	mm	0.1	3F 01 0000	01 yy tt 04	all except 00
003350009	000000000	TOTAL P. WATER MRF	kg/m^2	0.1	36 C8 0000	01 00 tt 0A	all
004000009	000000300	300 MB U GRD MRF	m/s	0.1	21 64 012C	01 00 tt 0A	all
004000009	000000500	500 MB U GRD MRF	m/s	0.1	21 64 01F4	01 00 tt 0A	all
004000009	000000700	700 MB U GRD MRF	m/s	0.1	21 64 02BC	01 00 tt 0A	all
004000009	000000850	850 MB U GRD MRF	m/s	0.1	21 64 0352	01 00 tt 0A	all
004000009	000000925	925 MB U GRD MRF	m/s	0.1	21 64 039D	01 00 tt 0A	all
004000009	000001000	1000 MB U GRD MRF	m/s	0.1	21 64 03E8	01 00 tt 0A	all
004020009	000000010	10 M U GRD MRF	m/s	0.1	21 69 000A	01 00 tt 0A	all
004040009	000000000	TROP U GRD MRF	m/s	0.1	21 07 0000	01 00 tt 0A	all
004100009	000000300	300 MB V GRD MRF	m/s	0.1	22 64 012C	01 00 tt 0A	all
004100009	000000500	500 MB V GRD MRF	m/s	0.1	22 64 01F4	01 00 tt 0A	all
004100009	000000700	700 MB V GRD MRF	m/s	0.1	22 64 02BC	01 00 tt 0A	all
004100009	000000850	850 MB V GRD MRF	m/s	0.1	22 64 0352	01 00 tt 0A	all
004100009	000000925	925 MB V GRD MRF	m/s	0.1	22 64 039D	01 00 tt 0A	all
004100009	000001000	1000 MB V GRD MRF	m/s	0.1	22 64 03E8	01 00 tt 0A	all
004120009	000000010	10 M V GRD MRF	m/s	0.1	22 69 000A	01 00 tt 0A	all
004140009	000000000	TROP V GRD MRF	m/s	0.1	22 07 0000	01 00 tt 0A	all
005000009	000000300	300 MB VV MRF	Pa/s	0.001	27 64 012C	01 00 tt 0A	all
005000009	000000500	500 MB VV MRF	Pa/s	0.001	27 64 01F4	01 00 tt 0A	all
005000009	000000700	700 MB VV MRF	Pa/s	0.001	27 64 02BC	01 00 tt 0A	all
005000009	000000850	850 MB VV MRF	Pa/s	0.001	27 64 0352	01 00 tt 0A	all
005000009	000000925	925 MB VV MRF	Pa/s	0.001	27 64 039D	01 00 tt 0A	all
005000009	000001000	1000 MB VV MRF	Pa/s	0.001	27 64 03E8	01 00 tt 0A	all

[1] These variables are not available at the 0-h projection; also, for projections of 264 hours and beyond, the GRIB ID's for words 18-21 become 01 tt 00 0A, respectively.

[2] The 2-m relative humidity is unavailable in the archive prior to April 1997.

[3] These variables are not available at the 0-h projection; also for projections of 264 hours and beyond, the precipitation amounts in the archives are obtained by multiplying the precipitation rate forecast (in mm/sec) by 43200 (number of seconds in 12 hours). Note, also that the GRIB IDs (words 9-12 and 18-21) are 3B 01 00 00 01 tt 00 0A and D4 01 00 00 01 tt 00 0A for the total precipitation rate and the convective precipitation rate, respectively, when the TDL archive program extracts the data from the NCEP files.

F. Radar Data Archive

Description of Grid and Variables Archived

The MDL radar data archive is derived from national radar reflectivity mosaics, which are generated every 15 min by WSI Corp on an approximate 2-km latitude-longitude grid for the CONUS. The data are specified in terms of discrete radar reflectivity levels (or categories) that range from 0 to 15, with each pertaining to a reflectivity range (see Table 12-5). The 2-km raw data are processed, and then two derived radar variables are archived on an approximate 10-km polar stereographic grid (11.906 km mesh at standard latitude of 60° N) whose standard longitude is 105° W (Fig. 12-4). The archive grid contains 481 points in the x-direction (left to right) and 385 points in the y-direction (bottom to top), totaling 185,185 points. The lower-left corner of the grid is positioned at 21.553° N, 120.018° W, which coincides with AVN gridpoint (53,1) (x,y). The data are stored in TDLPACK format (as described in Chapter 5) by subroutine PACK2D beginning with the lower-left corner gridpoint (defined as x=1, y=1). The data are unpacked and returned by subroutine UNPACK starting with the lower-left-corner gridpoint, proceeding left-to-right (in the order of increasing x), then bottom-to-top (in the order of increasing y).

As noted, two derived fields extracted from the raw radar reflectivity data are contained in the archive. The first field is the maximum reflectivity level (0 – 15) among the 2-km gridpoints that lie within a 10-km gridbox. The second field, called fractional reflectivity coverage, is the fraction (0.0-1.0) of the 2-km points lying within a 10-km box that have a nonzero reflectivity level. Each of these variables is derived (in the archiving code) following pre-processing of the raw data (see Archive Process below).
Since the data are archived four times per hour and the MOS-2000 system does not accommodate minutes, four variable identifiers are given to each of the two gridded variables, one for each 15-min period (see Table 12-6).

Archive Process

The archive process involves several steps. First, WSI tar files containing one day of compressed radar data are uncompressed. This results in separate HDF (Hierarchical Data Format) files, each valid for a particular 15-min time. The HDF files, containing raw 2-km gridded radar reflectivity data, are read by the archiving code. Then two types of diagnostic data are applied to change the raw reflectivity values at some gridpoints to “missing” values. The first data type is a constant 2-km (gridded) occultation field that involves geographic coverage by the radar network and locations of topograhic-obstruction contamination in the radar measurements. The second information type, which is included with each HDF file, is textual information that describes the status of real-time operations at each radar site. Following this pre-processing step, the two fields on the 10-km grid are derived for the MDL archive and output in TDLPACK format.

File Names

The following radar data files are used in the archive process described above:

Daily raw data files (compressed):
/lamp2000/radar/U2krefyy.ddd_daily.hdf.tar.gz, where yy is the two-digit year and ddd is the three-digit Julian Day. These file types are the compressed raw radar data files extracted from the tape obtained from WSI.

Raw data (individual) .hdf files: /lamp2000/radar/UhhmmZyy.ddd_refl_2km.hdf, where hh and mm denote hours and minutes, respectively, and yy and ddd are as before. These files are extracted from the .tar files (described above) after being uncompressed.

Monthly radar-data archive files: /lamp2000/arc/radar/yyyymm, where yyyy is the four-digit year, and mm is the two-digit month. These file types contain the MDL archived radar data on the 10-km grid, which is in TDLPACK format.

Radar Archive Files

The archive of the 10-km radar data is stored on on "ice" in directory /lamp2000/arc accessible from MDL's workstations. A backup copy of the archive can be transported to the IBM mainframe for storage on tapes.

Period of Record

The currently-available radar data archive begins at 00Z April 1, 1997 and extends through 23Z March 31, 2001.

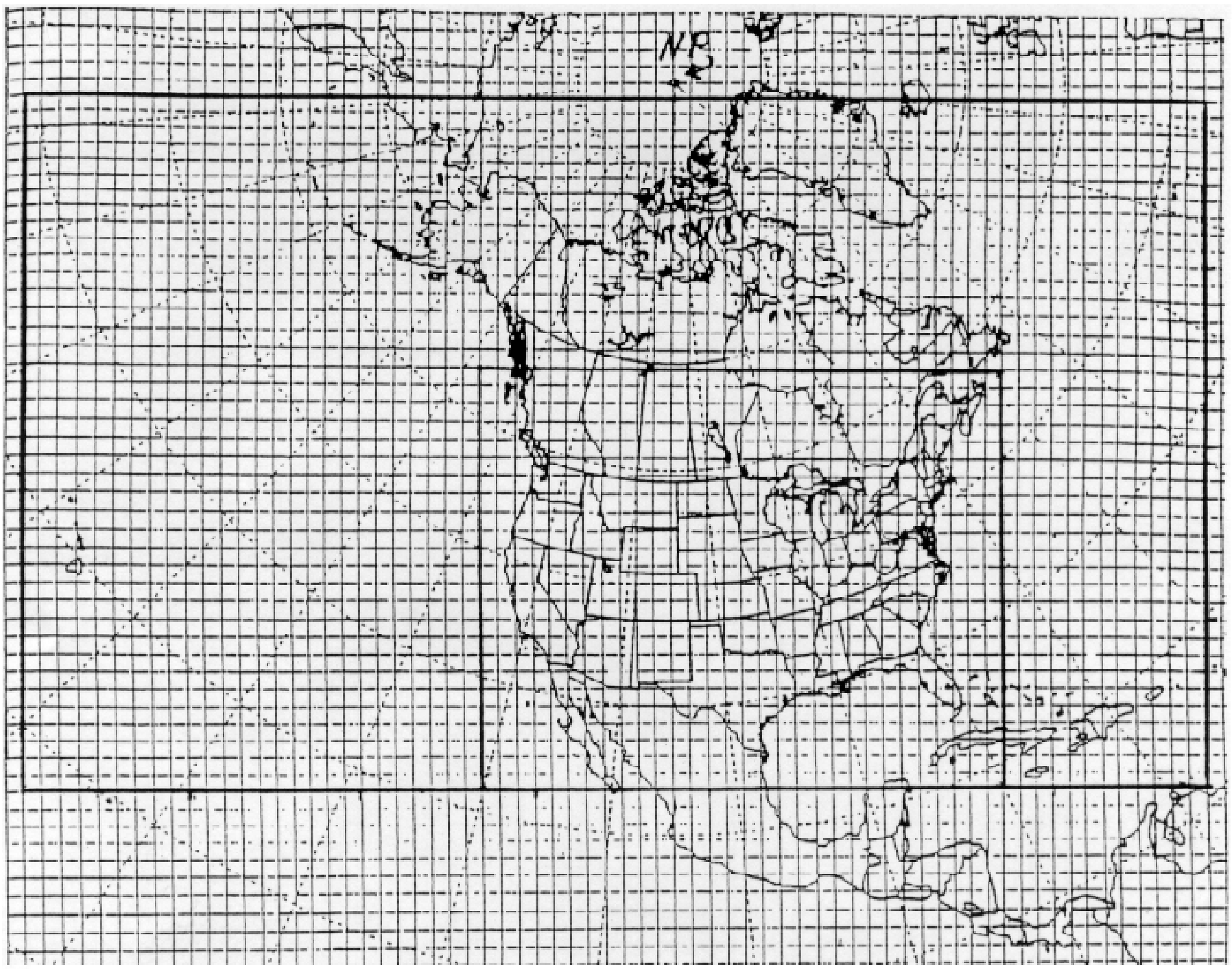

Figure. 12-4. Boundary of the 481 X 385 radar archive grid (inner rectangle bounded by heavy lines) within AVN archive grid (outer rectangle bounded by heavy lines). The upper-right point and the North Pole (NP) of the radar archive grid is (481,385) and (177,657), respectively, relative to the lower-left point (1,1). Relative to the lower-left point (1,1) of the AVN grid, the lower-left point (1,1) of the radar archive grid is at (53,1) and the North Pole is at (75,83).

Table 12-5. Radar reflectivity levels together with the corresponding reflectivity range.

Reflectivity Level	Reflectivity Range(dbZ)
0	≤4.5
1	5.0 - 9.5
2	10.0 - 14.5
3	15.0 - 19.5
4	20.0 - 24.5
5	25.0 - 29.5
6	30.0 - 34.5
7	35.0 - 39.5
8	40.0 - 44.5
9	45.0 - 49.5
10	50.0 - 54.5
11	55.0 - 59.5
12	60.0 - 64.5
13	65.0 - 69.5
14	70.0 - 74.5
15	≥75.0

Table 12-6. Radar variables archived. The columns labeled CCCFFFBDD and ID (words 2-4) refer to the four-word MOS-2000 variable identifier. In the first word, labeled CCCFFFBDD, characters 7-8 refer to the time of the data. For example, the variable 707615004 has a data time of 15 minutes after the hour. The column labeled Plain Language gives the plain language in the 32-bytes allowed for the variable description in the TDLPACK format. The Units (NA denotes not applicable), Range, and Precision apply to the data after it is unpacked.

CCCFFFBDD	ID (words 2-4)	Plain Language	Units	Range	Precision
707600004	0 0 0	REFLECT.(+00MIN)	NA	0 - 15	1.0
707615004	0 0 0	REFLECT.(+15MIN)	NA	0 - 15	1.0
707630004	0 0 0	REFLECT.(+30MIN)	NA	0 - 15	1.0
707645004	0 0 0	REFLECT.(+45MIN)	NA	0 - 15	1.0
707700004	0 0 0	REFLCT. RF(+00MIN)	NA	0.0 - 1.0	0.01
707715004	0 0 0	REFLCT. RF(+15MIN)	NA	0.0 - 1.0	0.01
707730004	0 0 0	REFLCT. RF(+30MIN)	NA	0.0 - 1.0	0.01
707745004	0 0 0	REFLCT. RF(+45MIN)	NA	0.0 - 1.0	0.01

1 # 13. VECTOR DATA ARCHIVES

A. Introduction

TDL archives various types of observational data including hourly surface observations, satellite cloud data, reports from the National Lightning Detection Network, and severe weather observations for use in developmental activities. Sections C, D, and E of this chapter describe the current archive of the hourly surface observations. Three types of data are included in the description. First, since December 1996, we've collected the METAR surface observations and put them into an ASCII (tabular) format. A number of quality-control checks, described in Section C, are performed on the METAR observations in building these tables. Second, snowfall and precipitation amount information available in the Supplemental Climatic Data (SCD's) reports that are received from the NWS Weather Forecast Offices are decoded; observations from the SCD's are added to the METAR tables. Third, the format and contents of the MOS-2000 hourly data archive, created from the METAR tables after additional quality control checks, is described. This particular data set provides the observational data for most of the MOS-2000 development. In Section F, archives of the Satellite Cloud Product (SCP) are described; reports of cloud cover are saved, edited, packed into MOS-2000 format, and later used to complement the ASOS cloud cover reports for the developmental system. Section G describes the older (MOS-1974) archive of hourly data obtained from Surface Aviation Observations (SAO's) as well as synoptic reports. Both the unedited and edited data are described. Note that the differences between the SAO and METAR archives are significant; the users must be aware of these differences if the data from the two archives are to be combined and used intelligently. Finally, Section H describes the lightning and severe weather data that are obtained and used in the developmental process.

B. Format

The format of the packed vector data is explained in Section C of Chapter 6 "Sequential Data Archive Files." Not all of the sections that follow refer to MOS-2000 packed data sets. As appropriate, the ASCII structure is given.

C. METAR Surface Weather Observations

<u>Summary of Archival Process</u>

Surface METAR observations are saved hourly for stations which send in surface reports to the NWS Telecommunications Gateway. These reports are transmitted to NCEP. Upon receipt, the METAR observations are decoded for all stations in NCEP's METAR dictionary (/nwprod/dictionaries/metar.tbl on the Cray) and are saved in the /dcom directory on the Cray in Binary Universal Form for the Representation of meteorological data (BUFR). Operationally, TDL runs software once each hour, at thirty minutes past the hour, to extract the observations from the BUFR format. Observations from 15 minutes before the hour to 15 minutes after the hour are retrieved. These reports represent the "observation of record." No specials (SPECI) are retrieved. The data are saved in two file formats in the /dcom file system: an ASCII table defined in Table 13-1 and an ASCII file containing the original METAR reports (rawmetar files). The data for each hour are saved in a separate file. The software also performs a limited number of quality control checks on the decoded hourly data. These checks are detailed below.

Each day, at approximately 4:00 a.m. LST, software is run to copy the previous calendar day's data from the operational file system to TDL's archive file system. This software also checks the volume of data in each hourly report, and if low, attempts to extract the data from the BUFR files once more. This process allows us to recover data which were not received in a timely manner. In addition, data contained in the Supplementary Climatological Data (SCD) reports are extracted and placed in the ASCII tables at this time. The extraction of the SCD data is described in Section D.

Weekly, the surface table and rawmetar files for the previous week are each concatenated and merged with the existing monthly file. These files (one each for the surface table and raw METAR data) are saved on both disk and tape. The original hourly files as well as the merged files remain on disk for 1 to 3 months depending on the availability of disk space.

Once a full month's data are collected, the disk and tape files for the surface table and raw METAR data are complete. Remember that both the tables and the raw METAR data at this stage are still in ASCII format. The tapes are currently managed through the Cray Reel Librarian tape management system. One year of data is saved on each volume set, and this volume contains 12 files corresponding to the 12 months of the year.

<u>Description of Variables Archived</u>

The following lists each element saved, including units if applicable, and quality checks performed. The weather elements are given in the order they appear in the hourly tables. The mnemonics in parentheses are those used as the headers to the table (Table 13-1).

ICAO METAR call letters (CALL):
Up to 8 characters are allowed for the station identifier. The program currently archives reports from stations only if the first letter of the ICAO ID is “K”, “P”, “M”, “C”, or “T”. Stations beginning with a K include the contiguous United States. Stations beginning with a “P” include Alaska (usually PA) and Hawaii (usually PH). Stations beginning with “M” are in Mexico. Stations beginning with a “C” are in Canada. Finally, stations beginning with a “T” are in the Caribbean which includes Puerto Rico. Stations beginning with a different letter are not processed, even though they may be in the NCEP files.

Station type (TYPE):
There are 11 possible station types:

AUTO = Fully automated
AO1 = ASOS with no precipitation reported
AO1A = ASOS with no precipitation reported, with human intervention
AO2 = ASOS with precipitation reported
AO2A = ASOS with precipitation reported, with human intervention
AMOS = AMOS - Automatic Meteorological Observing Station (full parameter)
RMOS = RAMOS - Remote Automatic Meteorological Observing Station (full parameter)
AUTB = AUTOB - Automatic observation (full parameter)

AWOS = AWOS - Automatic Weather Observing Station
MANU = Manual reporting station
WFO = Weather Service Forecast Office transmitting SCD's

If a station type other than one of the 11 listed is indicated, an "UNKN" is placed in the table to represent unknown station type.

Station latitude (LAT):
Latitude in degrees and hundredths of degrees. North is considered positive. No checking is performed on these values.

Station longitude (LON):
Longitude in degrees and hundredths of degrees. The values retrieved from BUFR are negative for west longitude; the sign is reversed before the longitude is written to the ASCII table.

Time of the METAR observation (TIME):
Two-digit hours and minutes are written in the table. A format specifier is used to retain the leading zeros.

Dry bulb temperature (TMP):
Units are °F. The data are stored in the NCEP BUFR files in Kelvin, so the temperatures are converted back to Fahrenheit and some rounding errors occasionally occur. Quality control checks are performed on the temperature before writing the data into the tables. The year is divided into four 3-month seasons, namely, spring (March-May), summer (June-August), fall (September-November), and winter (December-February). In addition to the seasonal stratification, the stations are divided into five geographic regions based on the latitude and longitude, with each region having a set of maximum and minimum allowed values for each season. If a station's temperature falls outside of the allowable range, the temperature is set to a missing value, namely, 999. The following limits are used for the temperature ranges:

	Winter	Spring	Summer	Fall
Southeast	-26, 102	-20, 117	25, 130	-11, 128
Southwest	-38, 105	-40, 126	12, 130	-25, 128
Northeast	-60, 102	-50, 116	8, 130	-42, 120
Northwest	-62, 105	-50, 119	0, 130	-49, 128
Far North	-80, 81	-66, 100	-6, 106	-67, 94

Southeast: lat ≤ 35°N and lon ≤ 100°W
Southwest: lat ≤ 35°N and lon > 100°W
Northeast: lat > 35°N and lat ≤ 50°N and lon ≤ 100°W
Northwest: lat > 35°N and lat ≤ 50°N and lon > 100°W
Far North: lat > 50°N

Dew point temperature (DEW):
Units are °F. This observation is processed in the same way as the dry bulb temperature. An additional set of allowable dew point temperatures have been created to quality check the dew point temperatures. Again, these ranges have been stratified by season and by geographic region. The dry bulb and dew point temperatures are also checked for consistency. If the dry bulb temperature is less than the dew point temperature, they are both set to missing.

	Winter	Spring	Summer	Fall
Southeast	-31, 85	-25, 85	20, 85	-16, 85
Southwest	-43, 85	-45, 85	-7, 85	-30, 85
Northeast	-65, 85	-55, 85	3, 85	-47, 85
Northwest	-67, 85	-55, 85	-5, 85	-54, 85
Alaska	-85, 74	-71, 74	-11, 74	-72, 74

The regions are the same as those used in the dry bulb temperature check.

Present weather (PRWX1, PRWX2, PRWX3):
METAR allows for up to six different types of present weather at a manual station and three present weather types at automated stations. These are reported in order of decreasing predominance. It is rare to ever have more than three reports at any station so only three are processed and archived.

NCEP decodes the METAR present weather character strings into WMO BUFR descriptions (see WMO Manual on Codes Table 020003). Some information is lost, particularly with mixed precipitation events, and intensity of the precipitation. This happens because BUFR combines several different present weather types into the same BUFR code. For instance, moderate freezing drizzle and heavy freezing drizzle are given the same BUFR code. When the TDL processing program runs, the distinction between these possible intensities is not available. The TDL processing program assigns a character string back to the BUFR number based on the event that is most common. (The original raw METAR reports are also archived so that we are able to check the original present weather reports.) A quality control check is performed here to ensure that only valid BUFR values have been assigned METAR character strings. Any unrecognizable BUFR values are set to blank in the table.

Horizontal visibility (VIS):
The visibility is reported in statute miles, stored by NCEP in meters, and then converted back to miles before being written in the table. Some rounding differences occasionally occur, but the conversion factors were tested to eliminate as many discrepancies as possible. Some quality control checks are performed. If the visibility is at least 3 miles, it is rounded to the nearest whole number. If the visibility is < 3 miles it will include the hundredths. If the visibility is greater than 100 miles, it is set to missing, represented by a blank. If the visibility is greater than 39 miles and less than or equal to 100 miles, the visibility is set to 40 miles. ASOS can report visibility as M1/4 SM which means < 1/4 mile horizontal visibility. A value of M1/4 is stored in the ASCII table as .20 to distinguish it from 1/4 mile which is written as .25. If a vertical visibility is reported, it is used as the cloud height of an obscuration.

Wind direction (WDR):
Wind direction is reported in whole degrees from 0 to 360. If the direction is > 360, the direction is assumed to be in error and is changed to missing, represented by a blank in the table. Calm winds are indicated by a wind direction of 0, and a wind speed of 0. If the winds are reported as variable, the wind direction is stored by NCEP as 0. A check is performed to see if the wind speed is > 0. If the wind speed is > 0 and the wind direction = 0, then the wind direction is changed by the TDL processing program to -9 which indicates a variable wind.

Wind speed (WSP):
Wind speed is reported in METAR in knots, converted by NCEP to m/s, and then converted back to knots by the TDL processing program. The wind speed value is rounded to the nearest integer before it is written in the ASCII table. Some quality control is done on the wind speed. Values > 75 knots are set to missing which is represented as a blank in the table. As mentioned in the description of the wind direction, if the winds are calm, the wind speed and direction are both set to 0. If the winds are variable, the wind direction is written as -9 in the table, and the wind speed is written in the table with the same wind speed reported in the METAR observation.

Wind gusts (GST):
Wind gusts are reported in knots. As in the wind speed, NCEP converts the gusts to m/s and the TDL processing program converts them back to knots. The resulting number is then rounded to the nearest integer. A quality control check is performed on the wind gust to change any values > 200 knots to missing, which is represented as a blank in the ASCII table.

Mean Sea level pressure (MSL):
The sea-level pressure is reported in METAR in millibars and tenths as a 3-digit integer which does not include the leading 9 or 10. NCEP converts this value to Pascals, and the TDL processor converts it back to millibars, out to the tenths place and written with the leading 9 or 10. The only quality control performed is to set any reports greater than 9999.9 to missing, which is represented by a blank.

Altimeter setting (ALT):
The altimeter setting is reported in METAR in hundredths of inches of mercury, stored by NCEP in Pascals, and then converted to inches of mercury by the TDL processor. The value is checked so that if it is reported to be greater than 99.9 inches of mercury, it is set to missing, which is represented as a blank in the table. The altimeter setting is written in the table out to two decimal places.

Cloud amount of layers (CA1, CA2, CA3, CA4, CA5, CA6):
Cloud amounts are reported in METAR in categories representing octas for as many as six possible layers. Clear (SKC or CLR) is 0 octas, few (FEW) is > 0 and ≤ 2 octas, scattered (SCT) is 3 - 4 octas, broken (BKN) is 5 - < 8 octas, and overcast (OVC) is 8 octas. For clear skies, SKC is used at manual sites while CLR is used at ASOS sites to mean clear below 12,000 ft. NCEP converts the cloud amount category to an integer and the TDL processor converts it back to the category. Another possible value for the cloud amount is partial obscuration (POB) which occurs when the category is followed by three slashes in the METAR report, for example, FEW///. NCEP assigns the same number to all partial obscurations so that the TDL processor no longer has the cloud category information, and a value of “POB” is written to the table. Total obscuration is another category of cloud amount. A total obscuration is indicated in METAR by the presence of a vertical visibility followed by a height. If the value from BUFR indicates a total obscuration, the TDL processor will set the cloud amount to “OB”, and the vertical visibility is used as the cloud height. The vertical visibility must be < 999 or the cloud height is set to missing. Once the TDL processor program encounters a total obscuration or an overcast layer, processing stops on the cloud values, and no more layers are allowed in the table. All clouds reported in METAR are assumed to be opaque. Three cloud layers are normally reported, and another

three layers may be manually appended. The program will allow up to six layers to be reported from any station type.

Cloud height for layers (CH1, CH2, CH3, CH4, CH5, CH6):
Cloud heights are reported in METAR in hundreds of feet for up to six layers. NCEP stores them in meters, and the TDL processor program converts them back to hundreds of feet. If the cloud amount indicates a partial obscuration, the cloud height is set to 0. As described in the previous paragraph detailing the cloud amount, if an obscuration is encountered, the vertical visibility is used as the cloud height. If a total obscuration or an overcast layer is encountered, the program will not allow any cloud heights above that layer. Any cloud heights greater than 999 (99,900 feet) are changed to missing which is represented by a blank value in the table.

Hourly precipitation amount (1PCP):
The hourly precipitation is reported in hundredths of inches by ASOS sites. Currently, only AO2 and AO2A hourly precipitation reports are being archived by TDL. NCEP stores this value in kg/m^2, and the TDL processing program converts it back to hundredths of inches. A report in METAR of P0000 indicates a trace. A “-4" is used to represent a trace in the ASCII tables. An hourly precipitation amount is always expected for AO2 and AO2A sites, so if the value is stored by NCEP as a missing (> 9999), it is assumed to be an indeterminate amount and written as -9. In METAR, an indeterminate amount is represented with 4 slashes “////”. If no value is received from NCEP, then we assume that precipitation did not occur, and a 0 is put in the table. For all other station types, a blank (that is, a missing indicator) is put in the table for the hourly precipitation amount.

3-hour precipitation amount (3PCP):
All station types are expected to report precipitation amounts for the past 3 hours at 0300, 0900, 1500, and 2100 UTC observation times. As mentioned in the description of the hourly precipitation amounts, the values are reported in hundredths of inches, stored in kg/m^2, and converted back to hundredths of inches. If a station does not report any precipitation amount, we assume there was no precipitation, and a 0 is written to the table. As described earlier, a -9 is used to represent an indeterminate amount, and a -4 is used to represent a trace in the table. A check is done on the reporting hour so that the 3-h precipitation amounts are allowed only at the expected observation times. All other hours will assume the 3-h amounts are missing, and will be represented with a blank in the table.

6-hour precipitation amount (6PCP):
All station types are expected to report precipitation amounts for the past six hours at 0000, 0600, 1200, and 1800 UTC observation times. As mentioned in the description of the hourly precipitation amounts, the values are reported in hundredths of inches, stored in kg/m^2, and converted back to hundredths of inches. If a station does not report any precipitation amount, we assume there was no precipitation, and a 0 is written to the table. Again, a -9 is used to represent an indeterminate amount, and a -4 is used to represent a trace in the table. A check is done on the reporting hour so that the 6-h precipitation amounts are only valid at the expected observation times. All other hours will assume the 6-h amounts are missing, and will be represented with a blank in the table.

24-hour precipitation amount (24PP):
All station types are expected to report precipitation amounts for the past 24 hours at the 1200 UTC observation time. As mentioned in the description of the hourly precipitation amounts, the values are reported in hundredths of inches, stored in kg/m^2, and converted back to hundredths of inches. If a station does not report any precipitation amount, we assume there was no precipitation, and a 0 is written to the table. Again, a -9 is used to represent an indeterminate amount, and a -4 is used to represent a trace in the table. A check is done on the reporting hour so that the 24-h precipitation amount is only valid at the expected observation time of 1200 UTC. All other hours will assume the 24-h amounts are missing, and will be represented with a blank in the table.

Minutes of Sunshine (SUN):
The duration of sunshine in minutes is reported at a small number of observing sites at 0800 UTC. The observation of sunshine is coded as the number of minutes during the previous calendar day. If the hour is not 0800 UTC, or the number of minutes exceeds 999, then the sunshine is set to missing which is represented by a blank.

6-hour maximum temperature (MX6):
The 6-h maximum temperatures are reported in METAR to tenths of °C at the synoptic times of 0000, 0600, 1200, and 1800 UTC. NCEP stores these temperatures in Kelvin and the TDL processor converts them to °F before writing them to the ASCII table. A check is done to ensure that these reports are received only at 0000, 0600, 1200, or 1800 UTC; at other hours, the value is set to missing which is represented with a blank in the table. Quality control is also performed on this observation by using the same method described earlier for the temperature. If a 6-h maximum temperature is outside the valid range for a given season and geographical region, the value is set to missing.

6-hour minimum temperature (MN6):
The 6-h minimum temperatures are reported in METAR to tenths of °C at the synoptic times of 0000, 0600, 1200, and 1800 UTC. NCEP stores these temperatures in Kelvin and the TDL processor converts them to °F before writing them to the ASCII table. The same time checks described in the 6-h maximum temperature are used. Quality control is also performed on this observation by using the same method described in the temperature section. If a 6-h minimum temperature is outside the valid range for a given season and geographical region, the value is set to missing.

Calendar day maximum temperature (X24):
The 24-h maximum temperature is reported in METAR to tenths of °C for the past 24 hours, at midnight local standard time. NCEP stores these temperatures in Kelvin, and the TDL processor converts them to °F before writing them to the ASCII table. Quality control is also performed on this observation by using the same method described in the temperature section. If the 24-h maximum temperature is outside the valid range for a given season and geographical region, the value is set to missing.

Calendar day minimum temperature (N24):
The 24-h minimum temperature is reported in METAR to tenths of °C for the past 24 hours, at midnight local standard time. NCEP stores these temperatures in Kelvin, and the TDL processor converts them to °F before writing them to the ASCII table. Quality control is also performed on this observation by using

the same method described in the temperature section. If the 24-h minimum temperature is outside the valid range for a given season and geographical region, the value is set to missing.

Snow depth (SND) on the ground:
The snow depth on the ground is reported in inches and comes from the METAR 4/sss group. It is stored by NCEP in meters and converted back to inches by the TDL processor program. Very few reports of snow depth come from the METAR data, so beginning in December 1998, another process was added to the hourly scripts to get the snow depth from the Supplemental Climatological Data (SCD). This process is described in more detail in Section D. If a value exists in the METAR data for the snow depth, that value is kept only if the SCD data contain no report of snow. If a value exists in the SCD data, that value will overwrite any snow depth data that came from the METAR reports.

Snowfall amount (SNF):
This is the depth of freshly fallen snow which is reported in METAR in the snow increasing rapidly group (SNINCR). The value is reported as inches/hr. Beginning in December 1998, we replaced this value with snowfall amounts obtained from the SCD data. While the SCD data report 6-h snowfall at the hours of 0000, 0600, 1200, and 1800 UTC, the METAR observations contain snowfall whenever more than .5 inches falls in a single hour. After December 1998, all values of snowfall from the METAR reports are set to missing which is represented by a blank. All snowfall values archived in the hourly surface tables now come from the SCD observations.

Satellite cloud amount category (SAE) from GOES-East:
These observations are needed to complement the ASOS cloud reports which do not detect clouds about 12,000 ft. The satellite cloud product (SCP) reports a cloud amount for mid- and high-level clouds in categories of clear (CLR), scattered (SCT), broken (BKN), or overcast (OVC). These data are currently being archived in a separate program and are not now merged into the hourly ASCII tables. See Section D which describes the SCP.

Satellite Effective cloud amount (SEE) from GOES-East:
The effective cloud amount (ECA) is a percentage value that represents the combined effects of the areal cloud coverage and the opacity of the clouds. In the past, this value has been useful for determining whether the clouds are thin or opaque. These data are currently being archived in a separate program and are not merged into the hourly ASCII tables. See Section D which describes the SCP.

Satellite cloud amount category (SAW) from GOES-West:
See the description of the satellite cloud amount category for GOES-East.

Satellite Effective cloud amount (SEW) from GOES-West:
See the description of the satellite effective cloud amount for GOES-East.

ASCII Table Structure

The first record in the ASCII file contains a header record with the valid date and time. The second record describes the types of observations in the data fields. Next are the observations, sorted alphabetically by ICAO station identifiers. Finally, a terminator record is used to end the file. The table contains 211 characters per station record, and will wrap to fit the size of the user’s screen. The following is a sample hourly observation table valid December 10, 1999, at 1200 UTC. The field descriptions that follow define the observational elements and were described previously.

```
 TDL HOURLY METAR TABLE VALID 1999121012  HOUR: 12Z

CALL    :TYPE:  LAT :   LON :TIME:TMP:DEW: PRWX1 : PRWX2 :PRWX3  : VIS
:WDR:WSP:GST:  MSL : ALT :CA1:CH1:CA2:CH2:CA3:CH3:CA4:CH4
:CA5:CH5:CA6:CH6:1PCP:3PCP:6PCP:24PP:SUN:MX6:MN6:X24:N24
:SND:SNF:SAE:SEE:SAW:SEW:

KDCA    :AO2A: 38.85:  77.03:1151: 48: 44:       :       :      :10.00:190:
7:   :1015.5:29.99:FEW: 80:SCT:100:BKN:250:   :
:   :   :   :   :   0:    :   0:   0:   : 51: 47:   :
:   :   :   :   :   :   :
~
~
~
ZZZZZZZZ: END-OF-HOUR 12Z
```

Table 13-1. Definition of the fields in the ASCII observation table.

Field	Field Description
CALL	ICAO station identifier
TYPE	Station type
LAT	Station latitude in degrees, positive is N
LON	Station longitude in degrees, positive is W
TIME	Hour and minute (UTC) of report
TMP	Dry bulb temperature in °F
DEW	Dew-point temperature in °F
PRWX1	First present weather
PRWX2	Second present weather
PRXW3	Third present weather
VIS	Visibility reported in statute miles
WDR	Wind direction in whole degrees
WSP	Wind speed in whole knots
GST	Wind gusts in whole knots
MSL	Mean Sea-level pressure in millibars
ALT	Altimeter in inches of mercury
CA1	Cloud amount category for 1st layer
CH1	Cloud height for 1st layer in hundreds of feet
CA2	Cloud amount category for 2nd layer
CH2	Cloud height for 2nd layer in hundreds of feet
CA3	Cloud amount category for 3rd layer
CH3	Cloud height for 3rd layer in hundreds of feet
CA4	Cloud amount category for 4th layer
CH4	Cloud height for 4th layer in hundreds of feet

Field	Field Description
CA5	Cloud amount category for 5th layer
CH5	Cloud height for 5th layer in hundreds of feet
CA6	Cloud amount category for 6th layer
CH6	Cloud height for 6th layer in hundreds of feet
1PCP	Hourly precipitation amount in hundredths of inches (AO2 and AO2A
3PCP	3-h precipitation amount in hundredths of inches, 0300, 0900, 1500,
6PCP	6-h precipitation amount in hundredths of inches, 0000, 0600, 1200,
24PP	24-h precipitation amount in hundredths of inches, reported at 1200
SUN	Duration of sunshine in minutes, reported at 0800 UTC
MX6	6-h max temperature in whole °F, 0000, 0600, 1200, 1800 UTC
MN6	6-h min temperature in whole °F, 0000, 0600, 1200, 1800 UTC
X24	24-h calendar day maximum temperature in whole °F, reported at
N24	24-h calendar day minimum temperature in whole °F, reported at
SND	Snow Depth on the ground in inches
SNF	Inches of freshly fallen snow from the past 6 hours (SCD data
SAE	Satellite cloud amount category for GOES-EAST
SEE	Satellite effective cloud amount for GOES-EAST
SAW	Satellite cloud amount category for GOES-WEST
SEW	Satellite effective cloud amount for GOES-WEST

Archive Process and File Names

As discussed at the beginning of Section C, the TDL processing program is run every hour to retrieve observations from the NCEP BUFR files and reformat them for use by TDL. Two output files are generated by this program every hour. The first file is referred to as the raw METAR file. These files are written to the directory **/dcom/us007003/yyyymmdd/mobsbfr** and given filenames of **rawmetar.hh**, where **yyyymmdd** represents the year, month, and day, and **hh** represents an hour from 00 to 23 UTC. The files consist of the raw METAR reports in their original form. The file is arranged alphabetically by the station's ICAO identifier, with reports of up to 255 characters long per line for each station.

The second set of files are the ASCII hourly surface tables. These are also saved to **/dcom/us007003/yyyymmdd/mobsbfr** with filenames of **sfctbl.hh**, where again the **hh** represents an hour from 00 to 23 UTC. The TDL processor program retrieves a single station's report from the tank used to store the dumped BUFR data, then decodes the BUFR values for each of the desired weather observations and writes the station's observations as an ASCII record. This process is repeated for every North American station in the tank. These reports are then sorted alphabetically with a UNIX sort command and appended to a file containing header information.

Once a day a recovery process is run on the METAR data so as to archive data that were too late to be picked up by the hourly jobs, to merge supplemental climatological data (SCD) for snow and precipitation into the appropriate hourly METAR files, and to write the raw METAR and ASCII hourly surface tables into TDL directories where they can be processed and archived. The files are written to the directories: /tdl4/we21ar/rotate/mtr and /tdl4/we21ar/rotate/hry.

File Names

The following files are used in the archive process described above:

NCEP BUFR files: /dcom/us007003/**yyyymmdd**/b000/xx007, where **yyyy** is the four digit year, **mm** is the two digit month, and **dd** is the two digit day of the month. These files are located on the Cray.

Operational Hourly files: /dcom/us007003/**yyyymmdd**/mobsbfr/sfctbl.**hh**; /dcom/us007003/**yyyymmdd**/mobsbfr/rawmetar.**hh**, where **yyyy** is the four digit year, **mm** is the two digit month, **dd** is the two digit day of the month, and **hh** is the hour of the day (00-23). The first of these files (sfctbl.**hh**) contains the hourly tables; the second of these files (rawmetar.**hh**) contains the raw metar reports. These files are located on the Cray.

Archive Daily files: /tdl4/we21ar/rotate/hry/**yyyymmddhh;** /tdl4/we21ar/rotate/mtr/**yyyymmddhh**, where **yyyy** is the four digit year, **mm** is the two digit month, **dd** is the two digit day of the month, and **hh** is the hour of the day (00-23). These files are located on the Cray and contain the tables and raw metar data, respectively, as described earlier.

Archive Monthly files: /tdl4/we21ar/archive/hry/hry**yyyymm**; /tdl4/we21ar/archive/mtr/mtr**yyyymm**, where **yyyy** is the four digit year, **mm** is the two digit month, **dd** is the two digit day of the month, and **hh** is the hour of the day (00-23). These files are stored on the Cray and contain the tables and raw metar data, respectively, as described earlier. The same files are also stored on tape as described below in the section "Archive Yearly files."

Archive Yearly files: The volume set names for the tapes are designated as: tdlhry**yyyys** and tdlmtr**yyyys**, where **yyyy** is the four digit year and **s** corresponds to the tape copy--**p** for primary, and **b** for backup. Each monthly file within the yearly volume set is designated as: **HRYyyyymm** or **MTRyyyymm** where **yyyy** corresponds to the four digit year and **mm** equals the two digit month.

Archive Tapes

The archived hourly data are stored on cartridge tapes and are listed in the Cray Reel Librarian (CRL) system. That system can be queried by the Cray user to determine the names and availability of the various volume sets. Generally, the tapes are stored in a separate room from the Cray, and a request must be made to the tape librarian to bring the tapes back to the Cray silos in order to be used. This retrieval nominally takes 24 hours; in reality, the time required can take more than a week. Once the tapes are placed back into the silo, the user can copy the data from tape to disk and then proceed with a specific task. A file on the Cray named: /jdsk41/we/tdllib/log/tapes/we21ar.list gives the names of the hourly files. This list is not necessarily current so that the interested user should query the CRL system. Use the "man" facility for the command "rlr" for further details on querying CRL.

As of December 1999, all of the archived hourly data had been moved back from tape to the Cray4 /dm (data migration) disk system for purposes of development. These data are stored in the files listed previously. The user is reminded to check the status of these files by issuing the "ls -l" command. If the files have been migrated to the silo tape system, the user can retrieve the files within a minute or so by issuing the "dmget" command. Alternatively, the submission of a batch job to Cray4 which requires the /dm files will force the system to retrieve the files back to disk.

Period of Record

The archive of the METAR data began on the CRAY in December 1996. In December 1998, we began archiving the SCD data in an effort to collect snow depth and snowfall reports. The SCD data are merged into the hourly tables and are not archived separately.

D. Supplemental Climatic Data (SCD) Reports

Summary of Archival Process

As explained in Section C, data contained in the Supplementary Climatological Data (SCD) reports are extracted and placed in the ASCII tables at this time. Weekly, the surface table and rawmetar files for the previous week are each concatenated and merged with the existing monthly file. These files (one each for the surface table and raw METAR data) are saved on both disk and tape. The original hourly files as well as the merged files remain on disk for 1 to 3 months depending on the availability of disk space.

Once a full month's data are collected, the disk and tape files for the surface table and raw METAR data are complete. Remember that both the tables and the raw METAR data at this stage are still in ASCII format. The tapes are currently managed through the Cray Reel Librarian tape management system. One year of data is saved on each volume set, and this volume contains 12 files corresponding to the 12 months of the year.

Description of Data and Variables Archived

According to Weather Service Observing Handbook No. 7, Weather Forecast Offices (WFOs) and Data Collection Offices shall report supplemental climatological data (SCD) at the synoptic times of 0000, 0600, 1200, and 1800 UTC. These SCD reports provide manual observations of various weather elements that either are not detected by ASOS or are not reported in METAR observations. Since some of these elements are useful for development and verification activities, TDL supplements its archive of METAR reports by using data from SCD reports.

Depth of New Snow (Snowfall)
One of the SCD weather elements archived by TDL is the depth of new snow during the past 6 hours (snowfall). Specifically, at 0000, 0600, 1200, and 1800 UTC, stations report a 6-h snowfall value when any amount of snow has fallen in the past 6 hours. The "931nnn" code group in the remarks section of SCD reports indicates snowfall, where "nnn" is in tenths of inches. This amount, which is stored in the snowfall (SNF) column of the TDL METAR tables, is the actual amount that has fallen, even if some (or all) of it has melted. Trace amounts are coded 931000 and are stored by TDL as -4 in the tables. If

no new snow has fallen and snowfall is not reported in a station's SCD, the snowfall is assumed to be zero at that site and is stored as 0. If a WFO does not transmit an SCD, the snowfall is assumed to be missing (see the Archive Process below).

Depth of Snow on the Ground
Another weather element that TDL archives is total depth of snow on the ground (snow depth). At 0000 and 1200 UTC, stations report snow depth for values greater than a trace. They do likewise at 0600 and 1800 UTC whenever more than a trace of snow is on the ground and when more than a trace of precipitation (water equivalent) has fallen in the past six hours. The "4/sss" code group in the remarks section of SCD reports indicates snow depth, where "sss" is in whole inches. If snow depth is a trace or less at 0000 or 1200 UTC, snow depth is not reported and is assumed to be zero. At 0600 and 1800 UTC, snow depth is considered missing at all stations which do not report a value. Snow depth is stored in the SND column of the TDL METAR table.

6-Hour Precipitation Amount
The third weather element that TDL obtains from SCD reports is the 6-h precipitation amount. At 0000, 0600, 1200, and 1800 UTC, WFOs that are not collocated with an ASOS report a 6-h precipitation amount (water equivalent) when any amount of precipitation has fallen in the past 6 hours. The "6RRRR" code group in the remarks section of SCD reports indicates precipitation amount, where "RRRR" is in hundredths of inches. This amount is stored in the 6-h precipitation (6PCP) column of the TDL METAR tables. A trace is coded as "60000" and stored as -4 by TDL, while an indeterminable amount is coded as "6////" and stored as -9 by TDL.

<u>Archive Process</u>

Every hour, TDL creates a table containing the standard hourly METAR observations. At the end of each day (approximately 0400 UTC), TDL runs a script to recover any hours that are missing data or that have low station counts. This script also calls a routine to add SCD reports to the METAR tables.

If a station transmitted both a METAR and an SCD for an hour (i.e., the WFO is collocated with an ASOS site), any snow depth or snowfall data in the SCD replace the METAR entry within the table. If an SCD report does not match a station in the METAR table, a new record containing only snowfall and snow depth is added to the table for that station. If the WFO is not collocated with an ASOS, the 6-h precipitation also is added to the table. In such cases, TDL designates the station as type "WFO" in the table.

Sometimes more than one METAR or SCD report is valid for the same station and hour. If multiple SCD records exist for the same station and hour, the latest SCD is used. If multiple METAR reports exist for the same station and hour, the SCD snow data are added to each METAR report. If a station sent a METAR report but no SCD, then the SND and SNF columns of the METAR report are left blank for that station and are considered missing.

<u>Information Resources</u>

The interested reader is referred to "Part IV Supplementary Observations" of the NWS Observing Handbook No. 7.

E. Hourly Surface Weather Observations

After a full month of hourly ASCII tables has been collected, we edit and pack the data into the MOS-2000 format, and write edited monthly files. The editing software, U520, reads the data contained in the hourly surface tables and performs numerous quality control checks. Appendix I to this chapter enumerates all the data extremes and inconsistencies the code currently looks for, and describes the modifications made to the data in each case. Additional information about the reporting and editing of specific weather elements is given later in this section. Once the data are edited, the new monthly MOS-2000 files are written out in the standard UNIX blocking structure. This format can be read on both the Cray and HP UNIX platforms.

An abbreviated version of the TDL station dictionary is used in the U520 editing process. Prior to editing a month of data, the reporting stations are surveyed, and new stations are flagged. New stations are included in the data once they begin reporting fairly consistently.

Description of Data and Variables Archived

As described earlier in Section C, the METAR observations are decoded from NCEP files, stored in ASCII tables, and later edited and packed into MOS-2000 format by U520. This section describes the edited, packed data. Note that the editing and packing process only saves one report per hour, and that this report represents the “observation of record.” No specials are saved; none are available in the ASCII tables.

Observed data are saved each hour for 38 variables and approximately 2000 stations. Table 13.2 lists the MOS-2000 identifiers, the plain language descriptor, the precision, and the units of the hourly data. Table 13.3 indicates the variables that are only available at specific hours. Unless listed in this table, each record in Table 13.2 is available at each hour of the day.

Hourly data for stations in the United States, Canada, Mexico, and the Caribbean are saved. The specific region in which these stations are located is identified by the leading letter in the ICAO station identifier. The above areas correspond to the letters K/P, C, M, and T, respectively. Only those stations which report in a given hour are included in the archived data for that hour. Consequently, the list of stations in the MOS-2000 header record varies from hour to hour. For a comprehensive list of stations which are currently reporting, or have reported in the past, see the TDL station dictionary which is described in Chapter 10 “Station Directory.”

The elements described in Table 13.2 are defined below. The first line of the sub-section gives the MOS-2000 identifier (CCCFFFBDD).

400006000
Latitude in degrees, to ten-thousandths.

400007000
Longitude in degrees, to ten-thousandths. West longitude is positive; East longitude is negative.

Table 13.2. Hourly observations prepared by U520 from the METAR reports. The column labeled CCCFFFBDD indicates the first word of the MOS-2000 identifier. The second, third, and fourth words of the identifier equal 0. The column labeled Plain Language gives the plain language stored in the 32 bytes allowed for variable description in the MOS-2000 format. The Units and the Precision columns describe the units and subsequent precision of the data after the vector data are unpacked. Unless listed in Table 13.3, all variables are available every hour.

CCCFFFBDD	Plain Language	Units	Precision
400006000	LATITUDE	degrees	0.0001
400007000	LONGITUDE	degrees	0.0001
700001000	OBSERVATION TIME	coded	1.0
700002000	STATION TYPE	coded	1.0
701200000	OBS SEA LEVEL PRES	mb	0.1
701250000	OBS ALTIMETER SETTING	in Hg	0.01
702000000	OBS TEMPERATURE	deg F	1.0
702006000	OBS CALENDAR DAY MAX	deg F	1.0
702016000	OBS CALENDAR DAY MIN	deg F	1.0
702100000	OBS 6-H MAX TEMP	deg F	1.0
702110000	OBS 6-H MIN TEMP	deg F	1.0
703100000	OBS DEW POINT	deg F	1.0
703200000	OBS 1-H PRECIP AMT	in	0.001
703205000	OBS 3-H PRECIP AMT	in	0.001
703210000	OBS 6-H PRECIP AMT	in	0.001
703315000	OBS 24-H PRECIP AM	in	0.001
704200000	OBS WIND DIRECTION	degrees	10.0
704210000	OBS WIND SPEED	kt	1.0
704211000	OBS MAX WIND GUST SPD	kt	1.0
708100000	OBS VISIBILITY	mi	0.01
708320000	OBS CLOUD AMT-LYR 1	coded	1.0
708321000	OBS CLOUD HGT-LYR 1	100's ft	1.0
708322000	OBS CLOUD AMT-LYR 2	coded	1.0
708323000	OBS CLOUD HGT-LYR 2	100's ft	1.0
708324000	OBS CLOUD AMT-LYR 3	coded	1.0
708325000	OBS CLOUD HGT-LYR 3	100's ft	1.0
708326000	OBS CLOUD AMT-LYR 4	coded	1.0
708327000	OBS CLOUD HGT-LYR 4	100's ft	1.0
708328000	OBS CLOUD AMT-LYR 5	coded	1.0
708329000	OBS CLOUD HGT-LYR 5	100's ft	1.0
708330000	OBS CLOUD AMT-LYR 6	coded	1.0
708331000	OBS CLOUD HGT-LYR 6	100's ft	1.0
708401000	OBS 6-H SNOWFALL AMT	in	0.01
708403000	OBS SNOW DEPTH	in	1.0
708500000	OBS WEATHER, GROUP 1	coded	1.0
708510000	OBS WEATHER, GROUP 2	coded	1.0
708520000	OBS WEATHER, GROUP 3	coded	1.0
709310000	OBS DLY SUNSHINE AMT	min	1.0

Table 13.3. Variables in the hourly surface observation data set described in Table 13.2 that are available only at specific hours. The column labeled CCCFFFBDD indicates the first word of the MOS-2000 identifier. The second, third, and fourth words of the identifier are always set to 0. The column labeled Plain Language gives the plain language stored in the 32 bytes allowed for variable description in the MOS-2000 format. The Hours Available column lists the hours at which the variable is available.

CCCFFFBDD	Plain Language	Hours Available
702006000	OBS CALENDAR DAY MAX	[1]
702016000	OBS CALENDAR DAY MIN	[1]
702100000	OBS 6-H MAX TEMP	0000, 0600, 1200, 1800
702110000	OBS 6-H MIN TEMP	0000, 0600, 1200, 1800
703205000	OBS 3-H PRECIP AMT	0300, 0900, 1500, 2100
703210000	OBS 6-H PRECIP AMT	0000, 0600, 1200, 1800
703315000	OBS 24-H PRECIP AMT	1200
708401000	OBS 6-H SNOWFALL AMT	0000, 0600, 1200, 1800
708403000	OBS SNOW DEPTH	0000, 0600, 1200, 1800
709310000	OBS DLY SUNSHINE AMT	0800

[1]The station reports this observation at local midnight; as a result, this variable is stored with the hour of local midnight and is found in the archive between the hours of 0300 and 1000 UTC.

700001000

Observation Time is coded as (hours * 100 + minutes) where the hours and minutes represent the time of the observation according to UTC. For example, if the observation is taken at 0950 UTC, the observation time is coded as 950. Reports are accepted up to 15 minutes before and after the hour.

NOTE: In archiving the METAR reports, we often encounter multiple record observations from a single site for the same hour. Multiple observations are handled in the following manner: manual observations are chosen over automated ones; if the station type is the same for the multiple reports, the observation closest to the official reporting time of 10 minutes before the hour is chosen. No specials (SPECI) are archived.

700002000

Station type, in coded form.
1 = Manual Station (MANU)
2 = Augmented NWS ASOS (AO2A)
3 = Stand-alone NWS ASOS (AO2)
4 = Augmented FAA ASOS (AO1A)
5 = Stand-alone FAA ASOS (AO1)
6 = Automated station (AUTO), includes AMOS, AWOS, RAMOS, AUTOB
7 = NWS Weather Forecast Office which sends SCD reports (WFO)
9 = Unknown Type

NOTE: Reports for WFO’s are included in the archive because WFO’s not collocated with an observing site report snowfall, snow depth, and 6-h precipitation amounts in the Supplemental Climatic Data (SCD). See 13.C.ii for further information on the SCD reports.

701200000

Sea Level Pressure in millibars, to tenths.
9999 = missing.

701250000

Station altimeter setting in inches and hundredths of mercury, to nearest hundredth.
9999 = missing.

702000000

Temperature in whole degrees Fahrenheit.
9999 = missing.

702006000

Maximum Temperature for previous calendar day in whole degrees Fahrenheit.
9999 = missing.

Reports are available at midnight LST, which means that the report is stored at any hour from 0300 to 1000 UTC, according to the station's time zone.

702016000

Minimum Temperature for previous calendar day in whole degrees Fahrenheit.
9999 = missing.

Reports are available at midnight LST, which means that the report is stored at any hour from 0300 to 1000 UTC, according to the station's time zone.

702100000

Maximum Temperature for previous 6 hours in whole degrees Fahrenheit.
9999 = missing.

Reports are available at 0000, 0600, 1200, and 1800 UTC only.

702110000

Minimum Temperature for previous 6 hours in whole degrees Fahrenheit.
9999 = missing.

Reports are available at 0000, 0600, 1200, and 1800 UTC only.

703100000

Dew Point in whole degrees Fahrenheit.
9999 = missing.

NOTE: ASOS stations can not report dew points below -30° F. Hence, when the air temperature is below -30° F, the dew points are recorded as missing (9999) in the MOS-2000 packed data. When preparing the hourly observations for predictand purposes, a subroutine (CORDP) is available in the u201 program to "correct" these extremely low dew points by setting the missing dew point value to that of the air temperature.

703200000

1-h Precipitation Amount in inches and hundredths.
0.004 = a trace of precipitation
0.0 = no precipitation
9999 = erroneous report or the station never reports precipitation.

NOTE: All precipitation amounts for the Canadian stations are set to 9999 because they do not report precipitation amounts reliably, if at all. Only stations of type AO2 and AO2A can report 1-hr precipitation.

703205000

3-h Precipitation Amount in inches and hundredths.
0.004 = a trace of precipitation
0.0 = no precipitation
9999 = erroneous report or the station never reports precipitation.

These reports are available only at 0300, 0900, 1500, and 2100 UTC

NOTE: All precipitation amounts for the Canadian stations are set to 9999 because they do not report precipitation amounts reliably, if at all.

703210000

6-h Precipitation Amount in inches and hundredths.
0.004 = a trace of precipitation
0.0 = no precipitation
9999 = erroneous report or the station never reports precipitation.

These reports are available only at 0000, 0600, 1200, and 1800 UTC

NOTE: All precipitation amounts for the Canadian stations are set to 9999 because they do not report precipitation amounts reliably, if at all.

703315000

24-h Precipitation Amount in inches and hundredths.
0.004 = a trace of precipitation
0.0 = no precipitation
9999 = erroneous report or the station never reports precipitation.

These reports are available only at 1200 UTC.

NOTE: All precipitation amounts for the Canadian stations are set to 9999 because they do not report precipitation amounts reliably, if at all.

704200000

Wind Direction in whole degrees, reported to nearest 10 degrees.
9999 = missing.

704210000

Wind Speed in whole knots.
9999 = missing.

704211000

Maximum Wind Gust Speed in whole knots as reported in hourly G-group.
9999 = missing.

708100000

Visibility in miles and hundredths, converted from the observing code, for example:

.12 = 1/8   1.00 = 1
.18 = 3/16   1.50 = 1 ½ etc.
.20 = < 1/4 (ASOS)
9999 = missing.

Note: In the U520 quality-control program, some visibility checks are not done for Mexican stations and non-Puerto Rican Caribbean stations because their visibility reports are not reliable (see Appendix I, subroutine VISWXQC for the specific checks not done).

708320000

Amount of lowest cloud layer, in coded form.

0 = CLR (Clear skies, automated observation)
1 = SKC (Clear skies, manual observation)
2 = FEW (> 0 but ≤ 2/8 coverage)
3 = SCT (3/8 - 4/8 coverage)
6 = BKN (5/8 - < 8/8 coverage)
8 = OVC (8/8 coverage)
9 = POB (< 8/8, sky partially obscured)
10 = OB (Sky totally obscured)
9999 = missing. A missing value indicates that either the report was incomplete, or questionable.

708321000

Cloud base of lowest layer, in hundreds of feet.
888 = clear conditions at this level and all levels above.
9999 = missing.

708322000

Amount of second lowest cloud layer, in coded form.
Code values are same as for lowest cloud layer, except that the codes for clear, partial obscuration, or total obscuration are never used.
9999 = missing. A missing value, with a valid report at the lowest level, implies no report was received for this layer, and the remaining cloud layer amount reports will also be 9999.

708323000

Cloud base of second lowest layer, in hundreds of feet.
9999 = missing. A missing value, with a valid report at the lowest level, implies no report was received for this layer, and the remaining cloud layer base reports will also be 9999.

708324000

Amount of third lowest cloud layer, in coded form.
Code values are same as for second lowest cloud layer.
9999 = missing. A missing value, with a valid report at a lower level, implies no report was received for this layer, and the remaining cloud layer amount reports will also be 9999.

708325000

Cloud base of third lowest layer, in hundreds of feet.
9999 = missing. A missing value, with a valid report at a lower level, implies no report was received for this layer, and the remaining cloud layer base reports will also be 9999.

708326000

Amount of fourth lowest cloud layer, in coded form.
Code values and processing are same as for third lowest cloud layer.

708327000

Cloud base of fourth lowest layer, in hundreds of feet.
Code values and processing are same as for third lowest cloud layer.

708328000

Amount of fifth lowest cloud layer, in coded form.
Code values and processing are same as for third lowest cloud layer.

708329000

Cloud base of fifth lowest layer, in hundreds of feet.
Code values and processing are same as for third lowest cloud layer.

708330000

Amount of sixth lowest cloud layer, in coded form.
Code values and processing are same as for third lowest cloud layer.

708331000

Cloud base of sixth lowest layer, in hundreds of feet.
Code values and processing are same as for third lowest cloud layer.

708401000

6-hour Snowfall Amount in inches and tenths.
0.0 = no snow reported in the SCD
0.04 = trace
9999 = missing.

The data are obtained from supplemental climatic data (SCD) and are available only at 0000, 0600, 1200, and 1800 UTC. We began the process to use the SCD reports as of 0000 UTC on December 18, 1998. Prior to that time, snowfall data were obtained from the METAR “SNINCR” report; these data were dropped and were deleted from the earlier archives once the SCD reports became available. Unless the station is known to send an SCD report, the snowfall amounts are set to missing in all cases.

NOTE: Although the SCD observations record snowfall to the nearest tenth of an inch, as of this time, NCEP only provides data for categories of 0.4, 0.8, etc., inches.

708403000

Snow depth, in inches.
0 = no snow depth reported in the SCD
9999 = missing.

Data are available only at 0000, 0600, 1200, and 1800 UTC. If snow depth is available in both the METAR and SCD reports, the SCD report is kept.

Unless the user knows that a station is an SCD site and reports reliably, it not possible to discern whether the data came from the SCD or METAR reports. Unless the station is known to send an SCD report, the snowfall amounts are set to missing in all cases.

NOTE: If snow did not fall within the previous 6 hours, the snow depth will usually not be reported at 0600 and 1800 UTC, even though snow may be present.

708500000

Weather in coded form (Group 1)
Note that the abbreviations which follow are those used in the METAR code. The code numbers are those found in the BUFR code tables and written out in the MOS-2000 format. (The code numbers processed from the BUFR tables are more extensive than the list shown below; however, many of the numbers distinguish between manual and automated observations. While we process all the numbers that we encounter, we categorize all the events according to the list below.) Note, also, that when the hourly METAR reports are decoded from the NCEP BUFR files and put back into an alphanumeric format, only the element in **BOLD** type below is written into the ASCII table. For example, the code number 7 in the BUFR files represents both BLDU and BLSA; the plain language “BLDU” is written in the ASCII table; the value 7 is stored in the packed data.

0 =	No weather reported	
4 =	**FU**	(Smoke)
5 =	**HZ**	(Haze)
6 =	**DU**	(Dust)
7 =	**BLDU**, BLSA	(Blowing dust, blowing sand)
8 =	**PO**, VCPO	(Dust whirls or sand whirls, dust or sand between 5 and 10 miles from station)
9 =	**VCDS**, VCSS	(Duststorm between 5 and 10 miles from the station, also sandstorm in the vicinity)
10 =	**BR**	(Mist with visibility $\geq$ 5/8 sm and $\leq$ 6 sm)
11 =	**PY**	(Spray)
16 =	**VCSH**	(Showers within sight, between 5 and 10 miles from the station)
17 =	**TS**, VCTS	(Thunderstorms with no precipitation, also thunderstorm in the vicinity)
18 =	**SQ**	(Squalls)
19 =	**+FC**, FC	(Tornado or waterspout, also funnel cloud)
31 =	**DS**, -DS, SS, -SS	(Moderate duststorm, also light duststorm or light or moderate sandstorm)
34 =	**+DS**, +SS	(Heavy duststorm, also heavy sandstorm)
36 =	**DRSN**	(Drifting snow)
37 =	**+DRSN**	(Heavy drifting snow)
38 =	**BLSN**, -BLSN, BLSA, DRSN	(Blowing snow)
39 =	**+BLSN**	(Heavy blowing snow (MANU))
40 =	**VCFG**	(Fog between 5 and 10 miles from the station)
41 =	**BCFG**	(Patchy fog)
44 =	**PRFG**	(Partial fog)
45 =	**FG**, MIFG	(Fog, visibility < 5/8 sm)
51 =	**-DZ**	(Light continuous drizzle)
53 =	**DZ**	(Moderate continuous drizzle)
55 =	**+DZ**	(Heavy continuous drizzle)
56 =	**-FZDZ**	(Light freezing drizzle)

57 =	**FZDZ**, +FZDZ	(Moderate or heavy freezing drizzle)
58 =	**-RADZ**, -DZRA	(Light rain and drizzle)
59 =	**RADZ**, +RADZ, DZRA, +DZRA	(Moderate or heavy rain and drizzle)
61 =	**-RA**	(Light rain)
63 =	**RA**	(Moderate rain)
65 =	**+RA**	(Heavy rain)
66 =	**-FZRA**	(Light freezing rain)
67 =	**FZRA**, +FZRA	(Moderate or heavy freezing rain)
68 =	**-RASN**, -SNRA, -DZSN, -SNDZ	(Light rain or drizzle and snow mixed)
69 =	**RASN**, SNRA, SNDZ, +RASN, +SNRA, DZSN, +DZSN, +SNDZ	(Moderate rain or drizzle and snow mixed, also heavy rain or drizzle and snow mixed)
71 =	**-SN**	(Light continuous snowfall)
73 =	**SN**	(Moderate continuous snowfall)
75 =	**+SN**	(Heavy continuous snowfall)
76 =	**IC**	(Ice crystals)
77 =	**SG**, +SG, -SG	(Snow grains)
79 =	**PE**, PL, +PE, +PL, -PL	(ice pellets) (**PL** after 11/05/98)
80 =	**-SHRA**	(Light rain shower)
81 =	**SHRA**, +SHRA	(Moderate or heavy rain shower)
83 =	**-SHRASN**, -SHSNRA	(Light showers of rain and snow mixed)
84 =	**SHRASN**, SHSNRA, +SHRASN, +SHSNRA	(moderate or heavy showers of rain/snow mixed)
85 =	**-SHSN**	(Light snow showers)
86 =	**SHSN**, +SHSN	(Moderate or heavy snow showers)
88 =	**GS**	(Small hail < 1/4 inch in diameter or snow pellets)
90 =	**GR**	(Hail ≥ 1/4 inch in diameter)
95 =	**TSRA**, -TSRA, -TSSN, TSSN	(Thunderstorms with light or moderate rain or snow)
96 =	**TSGR**, TSGS	(Thunderstorm with hail)
97 =	**+TSRA**, +TSSN	(Thunderstorm with heavy rain or snow)
98 =	**TSDS**, TSSS, +TSSS, +TSDS	(Thunderstorm with dust or sand)
121 =	**UP**	(Unknown precipitation from ASOS)
156 =	**+FZDZ**	(Heavy freezing drizzle, AUTO only)
166 =	**+FZRA**	(Heavy freezing rain, AUTO only)
174 =	**-PE**, -PL	(Light ice pellets, AUTO only)
176 =	**+PE**, +PL	(Heavy ice pellets, AUTO only)
183 =	**+SHRA**	(Heavy rain showers, augmented ASOS only)
187 =	**+SHSN**	(Heavy snow showers, augmented ASOS only)
196 =	**+TSGR**	(Thunderstorm, heavy with hail)
204 =	**VA**	(Volcanic ash)
207 =	**BLPY**	(Blowing spray)
208 =	**DRDU**, DRSS	(Drifting dust, also drifting sand)
9999 =	missing	(station unable to report present weather)

NOTE: For stations that can report present weather (MANU, AO2, AO2A), no reported weather is coded as a 0 in the first group, and 9999 in the second and third groups. For stations that cannot report present weather, all three present weather groups are set to 9999.

708510000

Weather in coded form (Group 2)
Codes used are the same as for weather group 1.

708520000
Weather in coded form (Group 3)
Codes used are the same as for weather group 1.

709310000
Daily Sunshine Amount in minutes.
9999 = missing.

Quality Control and Archive Process

The process used to obtain and quality control these data is described at the beginning of Section C; quality-control checks used in the U520 process are summarized in Appendix I to this chapter. As mentioned earlier, rudimentary quality control is also done prior to U520 in creating the ASCII tables containing hourly observations. Hourly observations prior to December 1, 1996 (termed MOS-1974 data for ease of discussion), were archived by a different process and quality control program (see Section G for specific details). The MOS-2000 and MOS-1974 processes differ significantly in some of the checks that were performed, due to both the implementation of METAR and the larger number of ASOS sites. Care was taken, however, to keep as many as possible of the quality-control checks consistent between the two processes. The data records contained in the two archive data sets also differ, as indicated in Tables 13.2 and 13.4 (see Section G). Differences exist in the way data were reported and stored between the two systems. Users must be aware of these differences in order to use both data sets intelligently.

Differences in methods of processing the data include the following:

- **Wind Gusts**
  MOS-1974: if no gust reported, value = 0.
  MOS-2000: if no gust reported, value = 9999.

- **Clouds**
  MOS-1974: if clear skies, cloud amount for every level = 0; cloud height for every level = 888.
  MOS-2000: if clear skies, cloud amount for first level = 0; cloud height for first level = 888. For all other levels, cloud amount and cloud height = 9999.
  MOS-1974: at every level above last level at which scattered or broken clouds are reported, cloud amount = 0; cloud height = 888. (If overcast is reported at lower level, both cloud amount and height = 9999 at all higher levels.)
  MOS-2000: at every level above last level at which scattered or broken clouds are reported, cloud amount = 9999; cloud height = 9999. (If overcast is reported at lower level, both cloud amount and height = 9999 at all higher levels.)
  MOS-1974: no distinction between clear skies reported at manual or automated station.
  MOS-2000: clear skies reported at automated sites (CLR) = 0; clear skies reported at manual sites = 1.

- **Variable Wind**
  MOS-1974: if variable wind direction, wind direction and speed = 9999.
  MOS-2000: if variable wind direction, wind direction = 990; wind speed = reported value.

- **Present Weather**
  MOS-1974: limited weather phenomena code values; obscurations to vision reported separately; one group of present weather phenomena allowed.
  MOS-2000: extensive weather phenomena code values; obscurations to vision reported as weather phenomena; three groups of present weather phenomena allowed.

- **Obscuration in Cloud Amount**
  MOS-1974: partial obscuration (POB) or total obscuration (OB) reported as total sky cover, not in cloud amount.
  MOS-2000: partial obscuration (POB) or total obscuration (OB) reported in first level of cloud amount; cloud amounts and heights for all levels above the first = 9999.

- **Sunshine Amount**
  MOS-1974: units of hours.
  MOS-2000: units of minutes.

File Names

At the time this section was written, the packed MOS-2000 hourly data had been stored in the Cray data migration system (/dm). In addition, data used for ongoing developmental work had been moved to files on the HP work stations located in SSMC2 and in the NOAA Science Center.

Monthly files on the Cray: /dm/we21ar/hourlydata/hre**yyyymm**, where **yyyy** equals the four digit year, and **mm** equals the two digit month. These are the edited hourly data files after processing by U520. One file is available for every month beginning in December 1996 (yyyymm=199612) and ending approximately in the month previous to the current month (for example, in November 1999 (yyyymm=199911)).

Monthly files on the HP known as blizzard: /mos1/mos/hourly/**yyyymm**, where **yyyy** equals the four digit year, and **mm** equals the two digit month. These are the edited hourly data files after processing by U520. On this file system, one file is available for every month beginning in December 1996 (yyyymm=199612) and ending in November 1999 (yyyymm=199912).

Monthly files on the HP known as chinook: /mos3/mos/hourly/hre**yyyymm**, where **yyyy** equals the four digit year, and **mm** equals the two digit month. These are the edited hourly data files after processing by U520. On this file system, one file is available for every month beginning in December 1996 (yyyymm=199612) and ending in November 1999 (yyyymm=199911).

Period of Record

The packed MOS-2000 archive of hourly data begins on December 1, 1996, and continues to the present.

Information Resources

The interested reader is referred to Federal Meteorological Handbook No. 1 describing the format and content of the METAR reports, NWS Observing Handbook No. 7, and the ASOS User’s Guide.

F. Satellite Cloud Product (SCP) Reports

Description of Data and Variables Archived

Satellite Cloud Product (SCP) bulletins are generated every hour by the National Environmental Satellite, Data, and Information Service (NESDIS) and are collected and archived every 6 hours by TDL. The GOES-EAST (GOES-8) satellite covers the eastern United States and includes Puerto Rico, while the GOES-WEST satellite (GOES-10) covers the western United States and includes Hawaii. There is currently no coverage for Alaska. The SCP bulletins are intended to be used to complement the ASOS cloud reports for heights above 12,000 feet.

Separate bulletins are generated for stations in the NWS Eastern, Central, Southern, and Western Regions, Hawaii, and Puerto Rico. For the GOES-EAST products, WMO headers of TCUS40-43 are used for satellite cloud estimates for the Eastern, Central, Southern, and Western Regions, respectively. The WMO header of TCUS62 identifies the satellite cloud estimate for Puerto Rico obtained from the GOES-EAST satellite. For the GOES-WEST products, headers of TCUS51-54 identify the satellite cloud estimates for the Central Region, Southern Region, Western Region, and Hawaii, respectively. The products created for the contiguous U.S. are based on the sounder instrument, and the products covering Hawaii and Puerto Rico are based on the imager instrument. NCEP receives an alert from NESDIS when the bulletins are ready. The bulletins are then retrieved and stored in ASCII files on Cray4 for our use. These files are concatenated into a single file for each satellite, edited for format, and packed into MOS-2000 format by TDL processes.

The information archived from the SCP for a particular station, date and time, are the categorical cloud coverage (CLR, SCT, BKN, OVC) at the mid level (between 631 and 400 mb) and high level (above 400 mb), and the ECA (average Effective Cloud Amount in percent). Cloud top data are also given in the SCP bulletins, but are not archived by TDL. The cloud coverage and ECA are available for approximately 930 stations for GOES-EAST and 310 stations for GOES-WEST, with some overlap in the central U.S.

The following CCCFFFBDD identifiers are used in the MOS-2000 packed data:

708350000
SCP Cloud Amount - GOES EAST

708351000
SCP Cloud Amount - GOES WEST

708360000
SCP Cloud ECA - GOES EAST

708361000
SCP Cloud ECA - GOES WEST

The cloud amount values are coded according to the following:
0 = CLR, MCLR (Clear, mostly clear)
1 = FEW (few clouds; not currently used)
3 = SCT (scattered)
6 = BKN (broken)
8 = OVC (overcast)
9999 = missing

The ECA values are stored as whole percent. Missing values are denoted by 9999.

Archive Process

The TDL hourly processor script described in Section C submits the SCP archive script, fmtscp.sc, every 6 hours (at 0000, 0600, 1200, and 1800 UTC). The SCP files are located on Cray4 in the directory **/pcom/foreign/text** and are only stored in operations for 12 hours before being removed to make room for newer files. This fmtscp.sc script formats and concatenates the bulletins and then puts them into the TDL daily archive directory **/tdl4/we21ar/rotate/sce(w).** At the end of the month, all of the SCP raw bulletins for GOES-EAST or WEST are concatenated into one large monthly file (**tdlsceyyyymmp** and **tdlscwyyyymmp**, where **yyyy** is the 4-digit year and **mm** is the 2-digit month). We then run the script scpedit.sc to get the data in the desired format with some quality control to check for valid dates and duplicate bulletins. Finally, the data are packed by running the script scppack.sc and placed into the archive directory.

File Names

The following files are used in the archive process described above:

NESDIS ASCII SCP Bulletins:
/pcom/foreign/text/ASOS.GOES**xx**.HOURLY.**yyyymmdd.hhnnssbb**, where **xx** equals 8 for GOES-8 and 10 for GOES-10, **yyyy** equals the 4-digit year, **mm** equals the 2-digit month, **dd** equals the 2-digit date of the month, **hhnnss** equals the 6-digit processing time of the bulletin, including the hour in UTC, minutes and seconds, and **bb** equals the bulletin number.

Daily rotating files:
/tdl4/we21ar/rotate/sc**r**/GOES**RRRR**SCP.**yyyymmddhh**, where **r** is e for GOES-EAST and w for GOES-WEST, **RRRR** is EAST for GOES-EAST and WEST for GOES-WEST, **yyyy** is a 4-digit year, **mm** is a 2-digit month, **dd** is a 2-digit date and **hh** is a 2-digit hour in UTC (the final hour processed).

Concatenated (unedited and unpacked) ASCII SCP monthly files:
/dm/we21kl/scp**cccc**/tdlsc**ryyyymm**p on Cray5
/mos1/mos/scp**cccc**/tdlsc**ryyyymm**p on chinook
/mos2/mos/scp**cccc**/tdlsc**ryyyymm**p on blizzard, where **cccc** is east for GOES-EAST and west for GOES-WEST, **r** is e for GOES-EAST and w for GOES-WEST, **yyyy** is a 4-digit year and **mm** is a 2-digit month.

Edited (ASCII) SCP monthly files:
/dm/we21kl/scp**cccc**/tdlse**ryyyymm**p on Cray5
/mos1/mos/scp**cccc**/tdlse**ryyyymm**p on chinook
/mos2/mos/scp**cccc**/tdlse**ryyyymm**p on blizzard, where **cccc** is east for GOES-EAST and west for GOES-WEST, **r** is e for GOES-EAST8 and w for GOES-WEST, **yyyy** is a 4-digit year and **mm** is a 2-digit month.

TDLPACK (edited and packed) SCP monthly files:
/dm/we21kl/scp**cccc/yyyymm** on Cray5
/mos1/mos/scp**cccc/yyyymm** on chinook
/mos2/mos/scp**cccc/yyyymm** on blizzard, where **cccc** is east for GOES-EAST and west for GOES-WEST, **yyyy** is a 4-digit year and **mm** is a 2-digit month.

Period of Record

The SCP archive for GOES-EAST and GOES-WEST began in September 1995. The first archive was generated by GOES-8 for the eastern U.S. and GOES-7 for the western U.S. In January 1996, GOES-7 was replaced by GOES-9. These earliest archives were run on the HDS mainframe and contained a mix of sounder-based and imager-based products. There were no bulletins available for Hawaii and Puerto Rico.

The unedited and unpacked files from the HDS mainframe are stored on the HP workstations (blizzard, chinook) in one file for the east and one file for the west for about 19 months of data from 19950901 to 19970331. They are designated in the workstation directories by the filenames tdlsce**hds**p and tdlscw**hds**p. These files were edited from September 1995 through December 1996 to remove SCP reports for stations that were still manual at the end of December 1996. Beginning in January 1997, all SCP reports were saved in the SCP files. This modification in the process was made because the hourly surface archive from the NCEP METAR BUFR files (see Section E) contains a station-type indicator. As a consequence, in preparing predictand data for development of sky cover equations, the manual cloud observations are not complemented. Without this indicator, no method existed to determine when to apply the complementing algorithm. The edited and packed files follow the naming convention described earlier.

In March 1997, the mainframe archives were turned off and the data were processed on Cray4. These files also follow the naming convention described earlier. In July 1998, GOES-9 stopped producing the SCP bulletins, and in August 1998, GOES-10 was brought online to take over for the western U.S.

Information Resources

A number of articles describing the SCP data have been published in AMS Preprint articles and Bulletins of the AMS. For a succinct description, the interested reader is referred to NWS Technical Procedures Bulletin No. 410.

G. Hourly Surface Aviation Observations (SAO’s)

Description of Data and Variables Archived

Prior to the implementation of the METAR reporting standard in July 1996, the hourly surface observations were transmitted in a format known as the surface aviation observations or SAO’s. When the data reached NCEP, the reports were decoded and stored in a special set of hourly files. TDL began its archive of hourly surface observations on December 14, 1976. After the conversion to METAR in July 1996, NCEP continued to generate these SAO-lookalike files and at the same time generated a new set of BUFR files that contained the METAR data. In December 1996, TDL converted its archive process to save the observations from the new files. The archive of SAO data continued through December 31, 1996. The SAO archive was later converted to the MOS-2000 format for the period of January 1, 1977, through December 31, 1996. Because of problems in the NCEP decoders in creating the SAO-lookalike files during the July 1996 to December 1996 period, the developer should be wary of the data quality and should use the METAR archive data (see Section E) for December 1996.

The following notes describe data that were archived from the SAO format, stored in an older packed format (henceforth, termed MOS-1974), and later converted to the MOS-2000 format. All of the details of the MOS-1974 format as well as known problems in the data may be found in Chapter X of TDL Office Note 74-14. The following description is a synopsis of the data. Except where noted, the descriptions refer to both unedited and edited data.

Table 13.4 lists the MOS-2000 identifiers, the plain language descriptor, the precision, and the units of the hourly data. Table 13.5 indicates the variables that are only available at specific hours. Unless listed in this table, each record is available at each hour of the day. Note that the 12-h precipitation amount (CCCFFFBDD = 703220000) is only available in the edited data because this variable was computed during the quality-control process.

The elements described in Table 13.4 are described further in the following. The first line of the sub-section gives the MOS-2000 identifier (CCCFFFBDD).

400006000

Latitude in degrees, to hundredths.

400007000

Longitude in degrees, to hundredths. West longitude is positive; East longitude is negative.

700001000

Observation Time and Synoptic Update Indicator

Number of minutes before (negative) or after (positive) the hour that the report was transmitted (prior to January 12, 1979) or that the report was observed (since January 12, 1979). In addition, if the original report was partially updated with synoptic data, 30 is either subtracted from or added to the observation time, depending on whether the time was before or after the hour. For example, if the report was observed at 1155 UTC and had been synoptically updated, this value would be set to -35. Reports are accepted up to 30 minutes before the hour and up to 29 minutes after the hour. To avoid conflict with total synoptic observations, any report observed exactly 30 minutes early will cause this value to be set to -29. For reports that consist entirely of synoptic data, the observation time is set to -60.

The observation time record has only been available since November 1, 1978. Note that errors in the archive software eliminated actual times until December 14, 1978.

Table 13.4. Hourly observations originally archived in the MOS-1974 format and later converted to the MOS-2000 format. The column labeled CCCFFFBDD indicates the first word of the MOS-2000 identifier. Note that the second, third, and fourth words of the identifier are always set to 0. The column labeled Plain Language gives the plain language stored in the 32 bytes allowed for variable description in the MOS-2000 format. The Units and the Precision columns describe the units and subsequent precision of the data after the vector data are unpacked. Unless listed in Table 13.5, all variables are available every hour.

CCCFFFBDD	Plain Language	Units	Precision
400006000	LATITUDE	degrees	0.01
400007000	LONGITUDE	degrees	0.01
700001000	OBSERVATION TIME	min	1.0
701100000	OBS STATION PRESSURE	mb	0.1
701200000	OBS SEA LEVEL PRES	mb	0.1
702000000	OBS TEMPERATURE	deg F	1.0
702004000	OBS 12-H MAX TEMP	deg F	1.0
702005000	OBS 24-H MAX TEMP	deg F	1.0
702014000	OBS 12-H MIN TEMP	deg F	1.0
702015000	OBS 24-H MIN TEMP	deg F	1.0
703100000	OBS DEW POINT	deg F	1.0
703210000	OBS 6-H PRECIP AMT	in	0.001
703220000	OBS 12-H PRECIP AMT	in	0.001
704010000	OBS U-WIND (EARTH)	kt	0.1
704110000	OBS V-WIND (EARTH)	kt	0.1
704200000	OBS WIND DIRECTION	degrees	10.0
704210000	OBS WIND SPEED	kt	1.0
704211000	OBS MAX WIND GUST	kt	1.0
707310000	OBS SEVERE WEATHER	coded	1.0
708000000	OBS CEILING HGT	100's ft	1.0
708100000	OBS VISIBILITY	mi	0.01
708250000	OBS OBSTRUCT VISION	coded	1.0
708300000	OBS OPAQUE SKY COVER	tenths	1.0
708310000	OBS TOTAL SKY COVER	coded	1.0
708320000	OBS CLOUD AMT-LYR 1	octa	1.0
708321000	OBS CLOUD HGT-LYR 1	100's ft	1.0
708322000	OBS CLOUD AMT-LYR 2	octa	1.0
708323000	OBS CLOUD HGT-LYR 2	100's ft	1.0
708324000	OBS CLOUD AMT-LYR 3	octa	1.0
708325000	OBS CLOUD HGT-LYR 3	100's ft	1.0
708400000	OBS SNOW DEPTH	coded	1.0
708401000	OBS 6-H SNOWFALL AMT	inches	0.01
708500000	OBS WEATHER	coded	1.0
709310000	OBS DLY SUNSHINE AMT	hr	0.001

Table 13.5. Variables in the hourly surface observation data set that are available only at specific hours. The column labeled CCCFFFBDD indicates the first word of the MOS-2000 identifier. Note that the second, third, and fourth words of the identifier are always set to 0. The column labeled Plain Language gives the plain language stored in the 32 bytes allowed for variable description in the MOS-2000 format. The Hours Available column lists the hours at which the variable is available.

CCCFFFBDD	Plain Language	Hours Available (UTC)
702004000	OBS 12-H MAX TEMP	0000
702005000	OBS 24-H MAX TEMP	0600
702014000	OBS 12-H MIN TEMP	1200
702015000	OBS 24-H MIN TEMP	1800
703210000	OBS 6-H PRECIP AMT	0000, 0600, 1200, 1800
703220000	OBS 12-H PRECIP AMT	0000, 0600, 1200, 1800
708400000	OBS SNOW DEPTH	0000, 0600, 1200, 1800
708401000	OBS 6-H SNOWFALL AMT	0000, 0600, 1200, 1800
709310000	OBS DLY SUNSHINE AMT	0800

701100000

Station Pressure in millibars, to tenths.
9999 = missing (either altimeter setting or station elevation is missing).

This value is derived from the station elevation in whole meters and the altimeter setting in inches and hundredths of mercury; therefore, the tenths may not be correct.

701200000

Sea Level Pressure in millibars to tenths.
9999 = missing.

702000000

Temperature in whole degrees Fahrenheit.
9999 = missing.

702004000

Maximum Temperature for previous 12 hours in whole degrees Fahrenheit.
9999 = missing.

Reports are available at 0000 UTC only.

702005000

Maximum Temperature for previous 24 hours in whole degrees Fahrenheit.
9999 = missing.

Reports are available at 0600 UTC only.

702014000

Minimum Temperature for previous 12 hours in whole degrees Fahrenheit.
9999 = missing.

Reports are available at 1200 UTC only.

702015000

Minimum Temperature for previous 24 hours in whole degrees Fahrenheit.
9999 = missing.

Reports are available at 1800 UTC only.

703100000

Dew Point in whole degrees Fahrenheit.
9999 = missing.

703210000

6-h Precipitation Amount in inches and hundredths.
0.004 = trace
0.0 = no precipitation
9999 = missing (the editing process eliminated the report or the station never reports precipitation).

These reports are available only at 0000, 0600, 1200, and 1800 UTC

703220000

12-hour Precipitation Amount in inches and hundredths.
0.004 = trace
0.0 = no precipitation
9999 = missing (the station never reports precipitation or either of the component 6-h precipitation values were missing).

These data were obtained by summing the 6-h precipitation amount values; reports are only available in the edited hourly datasets at 0000, 0600, 1200, and 1800 UTC.

704010000

U-Wind Component in knots to the nearest tenth, computed from the observed wind speed and direction.
9999 = missing.

704110000

V-Wind Component in knots to the nearest tenth, computed from the observed wind speed and direction.
9999 = missing.

704200000

Wind Direction in whole degrees, reported to nearest 10 degrees.
9999 = missing.

704210000

Wind Speed in whole knots.
9999 = missing.

704211000

Maximum Wind Gust Speed in whole knots as reported in hourly G-group.
0 = no report.

707310000

Severe Weather in coded form.
0 = none of the following reported

1 = Squall (Q)
2 = Thunderstorm (T)
3 = Severe Thunderstorm (T+)
4 = Hail (A)
5 = Tornado (not encoded from hourly reports)

708000000
Ceiling in hundreds of feet.
888 = unlimited
9999 = missing.

708100000
Visibility in miles and hundredths, converted from the observing code, for example:
.12 = 1/8 1.00 = 1
.18 = 3/16 1.50 = 1 ½ etc.
All observed values > 40 miles are set = 40.
9999 = missing.

708250000
Obstruction to vision in coded form.
0 = none of the following reported
1 = Smoke (K)
2 = Haze (H) or Smoke and Haze (KH)
3 = Blowing Obstructions (BD, BN, BS, BY)
4 = Fog (F)
5 = Ice Fog (IF)
6 = Ground Fog (GF)
9999 = missing.

708300000
Opaque sky cover in tenths.
0 = none
1 = one tenth
.
.
.
10 = ten tenths (includes obscured)
9999 = missing.

708310000
Total Sky Cover in coded form.
0 = clear
1 = partial obscuration
2 = thin scattered
3 = thin broken
4 = thin overcast
5 = scattered
6 = broken
7 = overcast
8 = obscured
9999 = missing.

708320000
Amount of lowest cloud layer, in octas.

9999 = missing.

A missing value indicates that either the report was incomplete, or this level could not be seen due to obscuring phenomena at the surface. If obscured, the Total Sky Cover (CCCFFFBDD = 708310000) code value for this observation has been set to 8.

This information has only been available since January 1, 1979.

708321000

Cloud base of lowest layer, in hundreds of feet.
888 = clear at this level and all levels above
9999 = missing.

This information has only been available since January 1, 1979.

708322000

Amount of second lowest cloud layer, in octas.
9999 = missing.

A missing value, with a valid report at the lowest level, implies an overcast at the lowest level, and the remaining cloud layer reports should also be 9999.

This information has only been available since January 1, 1979.

708323000

Cloud base of second lowest layer, in hundreds of feet.
888 = clear at this level and all levels above
9999 = missing.

This information has only been available since January 1, 1979.

708324000

Amount of third lowest cloud layer, or if more than 3 layers exist, the lowest layer constituting a ceiling, in octas.
9999 = missing.

A missing value, with a valid report at a lower level, implies an overcast at the lower level.

This information has only been available since January 1, 1979.

708325000

Cloud base of third lowest layer, in hundreds of feet.
888 = clear at this level
9999 = missing.

This information has only been available since January 1, 1979.

708400000

Snow Depth in coded form.
0 = none
1 = trace
2 = 1 inch
3 = 2 inches

4 = 3 inches
5 = 4 inches
6 = 5 inches
7 = 6-10 inches
8 = 11-20 inches
9 = 21 inches or more
9999 = missing.

These data were obtained from synoptic reports and are available only at 0000, 0600, 1200, and 1800 UTC.

708401000

<u>6-hour Snowfall Amount</u> in whole inches.
0.04 = trace
0.0 = no report.

The data were obtained from synoptic reports and are available only at 0000, 0600, 1200, and 1800 UTC.

708500000

<u>Weather</u> in coded form (values since May 16, 1979)
Note that the abbreviations are those used in the older SAO format, namely, L=drizzle, R=rain, RW=rain shower, ZL=freezing drizzle, ZR=freezing rain, IP=ice pellets, IPW=showers with ice pellets, S=snow, SP=snow pellets, SG=snow grains, SW=snow shower, and IC=ice crystals.

0 = none of the following reported

1 = L-, L--	16 = any combination of freezing and frozen precipitation
2 = L	
3 = L+	17 = IP-, IP--, IPW-, IPW--
4 = R-, R--	18 = IP, IPW
5 = R	19 = IP+, IPW+
6 = R+	20 = S-, S--, SP-, SP--, SG-, SG--
7 = RW-, RW--	21 = S, SP, SG
8 = RW	22 = S+, SP+, SG+
9 = RW+	23 = SW-, SW--
10 = ZL-, ZL--	24 = SW
11 = ZL	25 = SW+
12 = ZL+	26 = R and IP, L and IP (all intensities)
13 = ZR-, ZR--	27 = R and S, L and S (all intensities)
14 = ZR	28 = IC (all intensities)
15 = ZR+	

Prior to May 16, 1979, the following code values were used:
0 = none of the following reported

1 = L-, L--	16 = any combination of liquid and frozen precipitation
2 = L	
3 = L+	17 = IP-, IP--, IPW-, IPW--
4 = R-, R--	18 = IP, IPW
5 = R	19 = IP+, IPW+
6 = R+	20 = S-, S--, SP-, SP--, SG-, SG--
7 = RW-, RW--	21 = S, SP, SG
8 = RW	22 = S+, SP+, SG+
9 = RW+	23 = SW-, SW--
10 = ZL-, ZL--	24 = SW

11 = ZL   25 = SW+
12 = ZL+
13 = ZR-, ZR−-
14 = ZR
15 = ZR+

709310000

Daily Sunshine Amount in hours and thousandths.
9999 = missing.

Reports begin on April 16, 1980, and are only available at 0800 UTC.

Archive Process

As indicated above, these data were originally archived in MOS-1974 format. Data were saved in unedited format as well as in an edited format after using the quality control program known as M520. The checks available in M520 are documented in MOS OP No. 106 of TDL Office Note 75-2 (MOS Development System). Software was later written to convert the data from MOS-1974 to MOS-2000 format, and the old archives (both unedited and edited data) were converted in 1998. At that time, the archives were written to tapes in the CRAY Reel Librarian system. Tape archiving was done in 1-year segments (12 monthly files per volume set) for both unedited and edited data. A primary and backup tape were generated for each data type as well.

The original archive process for the MOS-1974 data was as follows. After a month of raw data was collected, the data were checked for errors. TDL's error checking program first converted all hourly reports to consistent units of measurement. This was accomplished through the use of a dictionary of hourly stations. TDL maintained this dictionary to identify specific characteristics of each station's reports and to indicate stations that did not report certain weather elements such as current weather or precipitation. Otherwise, no report of these elements would be interpreted as no significant weather at the time of observation or no precipitation having occurred over the previous 6 hours, respectively. The dictionary also provided information about automatic stations (in the era prior to ASOS) which not only did not report certain elements, but also did not conform to some of the standards for manual stations. For example, the automatic stations reported a continuous range of values of wind direction and visibility, while manual stations were limited to certain "reportable" values. The program also decoded maximum and minimum temperatures not in the range of 0°F - 100°F. When observations outside of this range occurred, the reported value in the raw hourly file was the observed value plus 100 for temperatures below 0°F and was the observed value minus 100 for temperatures above 100°F. The archived hourly data were then passed through a series of checks to determine whether each element fell within an allowable range of values. The permissible values varied depending on the location of the station. The program also monitored the internal consistency of the weather elements in each station's report. Included in these checks were comparisons of temperature-dew point, temperature-weather, weather-precipitation, weather-cloud cover, weather-visibility, and cloud cover-ceiling values. In addition, the consistency of values between hours was checked. Comparisons were made between the reported maximum/minimum temperatures and hourly values of temperature and between the hourly reports of current weather and the 6-h precipitation amount. The hourly trends of temperature, dew point, and pressure were monitored to check for consistency in these elements. Finally, when possible, the program eliminated or corrected cases where errors occurred in NCEP's processing or in TDL's archiving of the reports. Known errors that could not be corrected by M520 were noted in Chapter X of TDL Office Note 74-14. Both the unedited and edited data were saved on magnetic tape.

Three other caveats are important to using these older data. First, the information included in the hourly dictionary was not always current. Consequently, a station may have been automated, or may have changed from automated to manned, and yet the editing software (M520) was unaware of that fact. Secondly, we found that the data collected from July 1, 1996, through December 31, 1996, were of questionable quality because of the NCEP decoders in use at that time. Finally, the TDL archive and quality-control process of the SAO data as well as the maintenance of the hourly dictionary were not modified to account for the introduction of ASOS in the mid-1990's. As a result, the reports from ASOS sites may not have been adequately checked for quality until the new quality-control process described in Sections C and E and Appendix I was implemented.

File Names

At the time this section was written, the hourly data had been stored in the Cray data migration system (/dm) as well as on tapes accessible to the Cray. In addition, data used for ongoing developmental work had been moved to files on the HP work stations located in SSMC2 and in the NOAA Science Center.

Monthly files on the Cray: /dm/we21ar/hourlydata/hre**yyyymm**, where **yyyy** equals the four digit year, and **mm** equals the two digit month. These are the edited hourly data files and one file is available for every month beginning in January 1977 (yyyymm=197701) and ending in December 1996 (yyyymm=199612)

Yearly files on the Cray: The volume set name of the tape is designated as tdlhr**xyyyys**, where **x** corresponds to the level of processing of the data (**u** for unedited data; **e** for edited data), **yyyy** corresponds to the four digit year, and **s** corresponds to the tape status, that is a **p** for the primary tape set and a **b** for the backup set. Each monthly file within the yearly volume set is designated as: **HRXyyyymm**, where **X** corresponds to the level of processing of the data (**U** for unedited data; **E** for edited data), **yyyy** corresponds to the four digit year, and **mm** equals the two digit month.

Monthly files on the HP known as blizzard: /mos1/mos/hourly/**yyyymm**, where **yyyy** equals the four digit year, and **mm** equals the two digit month. On this file system, one file of edited data is available for every month beginning in January 1994 (yyyymm=199401) and ending in November 1996 (yyyymm=199611). Starting in December 1996, the data are taken from the METAR archive (see Section C). The unedited data are not available on this platform.

/mos3/mos/hourly/**yyyymm**, where **yyyy** equals the four digit year, and **mm** equals the two digit month. On this file system, one file of edited data is available for every month beginning in January 1992 (yyyymm=199201) and ending in December 1993 (yyyymm=199312).

Monthly files on the HP known as chinook: /mos3/mos/hourly/hre**yyyymm**, where **yyyy** equals the four digit year, and **mm** equals the two digit month. On this file system, one file of edited data is available for every month beginning in January 1992 (yyyymm=199201) and ending in November 1996 (yyyymm=199611). Starting in December 1996, the data are taken from the METAR archive (see Section C). The unedited data are not available on this platform.

Archive Tapes

As indicated above, the archived hourly data (both unedited and edited) are stored on cartridge tapes and are listed in the Cray Reel Librarian (CRL) system. That system can be queried by the Cray user to determine the names and availability of the various volume sets. Generally, the tapes are stored in a separate room from the Cray, and a request must be made to the tape librarian to bring the tapes back to the Cray silos in order to be used. This retrieval nominally takes 24 hours; in reality, the time required can take more than a week. Once the tapes are placed back into the silo, the user can copy the data from tape to disk and then proceed with a specific task. A file on the Cray named: /jdsk41/we/tdllib/log/tapes/we21ar.list gives the names of the hourly files. This list is not necessarily current so that the interested user should query the CRL system. Use the “man” facility for the command “rlr” for further details on querying CRL.

As of December 1999, all of the archived hourly data had been moved back from tape to the Cray4 /dm (data migration) disk system for purposes of development. These data are stored in the files listed previously. The user is reminded to check the status of these files by issuing the “ls -l” command. If the files have been migrated to the silo tape system, the user can retrieve the files within a minute or so by issuing the “dmget” command. Alternatively, the submission of a batch job to Cray4 which requires the /dm files will force the system to retrieve the files back to disk.

Period of Record

The archive of hourly observations obtained from the SAO format begins with 0000 UTC data on January 1, 1977, and ends at 2300 UTC on December 31, 1996.

Information Resources

The interested reader is referred to TDL Office Note 74-14 described earlier and to Federal Meteorological Handbook No.1 describing the format and content of the SAO’s.

F. Lightning and Severe Weather Reports

Description of Data and Variables Archived

Cloud-to-ground (cg) lightning data from the National Lightning Detection Network and reports of severe weather collected by the Office of Meteorology are used in the development of thunderstorm and severe thunderstorm forecast equations. The lightning data contain information on the time of the flash, the total number of cg flashes, the number of cg flashes that were negative, the number of cg flashes that were positive, the maximum signal strength, the number of strokes, and the associated geographic location in degrees of latitude and longitude. The severe weather data contain the number of tornadoes, the maximum tornado F-scale, the number of hail reports, the maximum hail size, the number of damaging wind reports, and the maximum wind speed. These severe weather reports also have an associated time and geographic location.

The lightning data are provided by the Global Hydrology Resource Center (GHRC) at NASA's Global Hydrology and Climate Center, in Huntsville, Alabama. We send a written request to GHRC about every 6 months and the most recent 6 months worth of data are provided on a cartridge tape.

The severe weather observations are collected by NWS forecast offices and sent to the Office of Meteorology (OM). These are the same reports used in the Storm Data publication maintained by the National Climatic Data Center (NCDC). A written or oral request is made to OM for the severe weather data. In the past, OM gave us the data on a computer disk, but since 1999, they have sent the data through electronic mail as an attached text file for the entire requested period, which is normally 6 months.

Archive Process

An HP workstation is used to read the cartridge tape and extract the daily lightning data files from tarred files. The daily lightning files are ftp'd to the IBM SP where they are checked for missing hours, and concatenated into monthly and yearly files. The severe weather data are ftp'd from a personal computer to the IBM SP, where the reports are put in chronological order, duplicate reports are removed, and the time of the reports is changed from Central Standard Time to UTC. The program U523 is run to create MOS-2000 data sets from the lightning and severe weather data. U523 creates packed MOS-2000 records for each hour of the year for 10,057 grid points covering the contiguous United States and adjacent areas.

File Names

daily lightning files:
HP's: /mos11/mos/lightning/upload/Nflash**yy.ddd**_daily.lit.raw, where **yy** is the last 2-digits of the year, and **ddd** is the Julian day of the year. These daily files (in ASCII) are created and named when they are extracted from the tapes provided by GHRC. To save space, the files are not stored permanently on the workstation since they can be recovered from the original tapes.

concatenated monthly lightning files:
IBM SP: /gpfsuser/g06/we21kh/data/lightning/ltg.**yyyymm**, where **yyyy** is a 4-digit year and **mm** is a 2-digit month. This ASCII file is created with the script catyymm.sc by using the daily lightning files as input.

edited monthly lightning files:
IBM SP: /gpfsuser/g06/we21kh/data/lightning/ltgarc.**yyyymm**, where **yyyy** is a 4-digit year and **mm** is a 2-digit month. The edited monthly files are created with the script order.sc which checks the data for errors and chronological order. The order script also changes the format slightly to prepare the data for use in the U523 program. The concatenated monthly lightning files (in ASCII) are used as input to U523.

yearly lightning files:
HP's: /mos11/mos/lightning/ltgyear.**yyyy**.Z (compressed to save space)

IBM SP: /gpfsuser/g06/we21kh/data/lightning/ltgyear.**yyyy**, where **yyyy** is a 4-digit year. These yearly files (in ASCII) are created with the script catyyyy.sc, which concatenates the edited monthly files.

edited yearly severe weather files:
HP's: /mos11/mos/svr/svrraw/svrstm**yyyy**.txt
IBM SP: /gpfsuser/g06/we21kh/data/severe/svrstm**yyyy**.txt, where **yyyy** is the 4-digit year. These ASCII files are created by the program SVRFMT by using the OM text files as input.

]**packed lightning and severe weather files**:
HP's: /mos11/mos/tdlpack/U523**yyyy**.packed
IBM SP: /tdl4/we21kh/thunderstorms/packed/U523**yyyy**.out, where **yyyy** is a 4-digit year. These files contain the severe weather and cg lightning data packed in the MOS-2000 format with a report valid for each hour during the year. These files are generated by the program U523 and contain data for 10,057 grid points (or "stations") covering the contiguous United States and adjacent areas.

Archive Tapes

The yearly cg lightning files are available on 2.0 GB data cartridges.

Period of Record

The lightning data and severe weather data are available in TDL packed format since April 1, 1994, and are current to September 30, 1999.

MOS2000 ID's for thunderstorms and severe weather

CCCFFFBDD	Packed Variables for Thunderstorms and Severe Weather
400006000	Latitude
400007000	Longitude
707200000	Occurrence of a thunderstorm (lightning and/or severe weather)
707300000	Unconditional occurrence of severe weather
707310000	Conditional occurrence of severe weather
707320000	Number of tornadoes
707330000	Maximum tornado f-scale
707340000	Number of hail reports
707350000	Maximum hail size
707360000	Number of damaging wind reports
707370000	Maximum wind speed
707380000	Total cloud-to-ground lightning flashes
707390000	Negative cloud-to-ground lightning flashes
707400000	Positive cloud-to-ground lightning flashes
707410000	Signal strength
707420000	Number of strokes

Information Resources

See Appendix II for a description of the cloud-to-ground lightning data provided by GHRC.

Global Atmospherics, Inc., Lightning Information, http://www.glatmos.com/lightinfo/lightinfo.html

Lightning & Atmospheric Electricity Research at the Global Hydrology and Climate Center (GHCC), http://thunder.nsfc.nasa.gov

NCDC Publications, Storm Data, http://www5.ncdc.noaa.gov/pubs/publications.html#SD

National Weather Service Training Center, AWIPS Informational Series: Operational Uses of Lightning Data, http://www.nwstc.noaa.gov/d.HMD/Lightning/Ltng_home.htm

Severe Thunderstorm Climatology, http://www.nssl.noaa.gov/hazard

APPENDIX I TO CHAPTER 13

Explanation of Errors and Output of Hourly Data Checking Program U520

Error output from U520 is written to one file and contains 7 columns. They are as follows:

1. Station ICAO
2. Date of error.
3. Observation instrument type.
4. Error code ("Nerr").
5. Old data value to be changed due to error.
6. New data value.
7. English explanation of error.

The tables below indicate the quality control checks, actions taken, and error code for each subroutine of u520.

SUBROUTINE TMPQC

Temperature	Report error	Nerr
Consistency check over 3 hours. If there are 2 consecutive hours of an increase/decrease in temperature by 10 degrees, check difference between the average of temperatures 1 and 3 and temperature 2. If +/- 10 degrees.	Change temperature to 9999.	9901
If the reported synoptic 6-h max (min) is out of tolerance (tolerance is 6+(2*the number of hours missing)).	Change 6-h max (min) to 9999.	9915 (max) 9925 (min)
If the 6-h max (min) calculated from the hourlies is greater (less) than the reported 6-h max (min) by 2 degrees or more and this occurred more than once during the synoptic period.	Change 6-h max (min) to 9999.	9911 (max) 9921 (min)
If the 6-h max (min) calculated from the hourlies is greater (less) than the reported 6-h max (min) by only one degree and this occurred more than once during the synoptic period or if same but occurred only once during the period (with the exception of the first or last hour).	Error is a rounding problem, change the hourly temperatures that reflect this to the correct temperature.	9902

If the 6-h max (min) calculated from the hourlies is greater (less) than the reported 6-h max (min) and this occurred only once during the period on either the first or last hour of the synoptic period.	Change the 6-h max (min) to reflect the correct temperature.	9912/ 9913; (max) 9922/ 9923 (min)
If the 6-h max (min) calculated from the hourlies is greater(less) than the reported 6-h max (min) by 2 degrees or more on any hour with the exception of first or last hour of the synoptic period.	Change the 6-h max (min) to 9999.	9914 (max) 9924 (min)
If the 6-h max is less than the 6-h min.	Change the 6-h max and min to 9999.	9930
If the 24-h max is less than the 24-h min.	Change the 24-h max and min to 9999.	9931

Dewpoint	Report error	Nerr
If the dewpoint is less than -30 degrees F, and the dewpoint depression is greater than 15 degrees.	Change dewpoint to 9999.	9800
Consistency check over 3 hours. If there are 2 consecutive hours of an increase/decrease in dewpoint by 10 degrees, check difference between the average of dewpoints 1 and 3 and dewpoint 2. If +/- 10 degrees.	Change dewpoint 2 to 9999.	9802
If an hourly temperature was changed due to rounding, cross check the temperatures with the dewpoints to assure that the dewpoint will not exceed the temperature. If it does,	Change the dewpoint to the new temperature so that the dewpoint depression is 0.	9803

SUBROUTINE PCPNQC

Precipitation amount	Report error	Nerr
Out of range 1-, 3-, 6-, and 24-h amounts. (1-h > 3.75 in, 3-h > 9 in, 6-h > 12 in, and 24-h > 24 in)	Change to 9999.	9700

If present weather was reported, but no 1-h precip amount was reported.	Change 1-h precip to 9999.	9705
If present weather was reported, but no 3-h precip amount was reported (MANU, AO2, AO2A).	Change 3-h precip to 9999.	9706
If present weather was reported, but no 6-h precip amount was reported (MANU, AO2, AO2A).	Change 6-h precip to 9999.	9707
If 3-h precipitation amount is greater than 6-h amount.	Change both to 9999.	9710
If 3-h precipitation amount reported and 6-h report is missing and no additional precipitation occurred in the last three hours of the synoptic period.	Change the 6-h amount to equal the 3-h amount.	9711
For ASOS, if 6-h precipitation was reported (amount must be .01 or .02), the dewpoint depression was less than 2, FG (visibility less than ½ mile) was reported, no precipitation was observed and the sky was either partially or totally obscured then, ASOS dew.	Change the 1-, 3-, and 6-h amounts to 0. Keep track of the amount of precipitation eliminated for one day.	9720
For ASOS, if 6-h precipitation amount was reported, no weather observed, sky was clear at least 4 of the hours during the period, and the temperature rose from lower or equal to 30 to a temperature greater than or equal to 35 (unless the temperature has remained above freezing allowing for additional snowmelt from earlier periods, max of 4 unless the temperature falls below freezing again), then ASOS snowmelt.	Change the 1-, 3-, and 6-h amounts to 0. Keep track of the amount of precipitation eliminated for one day.	9730
If precipitation amounts have been eliminated due to dew or snow melt and no other precip was reported.	Change 24-h precip to 0.	9740
If precipitation amounts have been eliminated due to dew or snow melt and other precip was reported.	Subtract the amounts eliminated from the 24-h period ending 12z	9741
SCD snowfall amount reported without a 6-h liquid equivalent.	Change 6-h precip amount to 9999.	9750

SCD snowfall amount out of range (> 20 inches).	Change snowfall to 9999.	9760
Indeterminant precip amount (-9) reported.	Change to 9999. **DOES NOT WRITE ANYTHING TO ERROR FILE.**	Nothing

SUBROUTINE PRESQC

Pressure	Report error	Nerr
Out of range, MSL > 1075 mb or < 875 mb.	Change pressure to 9999.	9601
Out of range, altimeter > 32.00 in or < 24.00 in.	Change altimeter to 9999.	9602
Consistency check over 3 hours. If there is a decrease/increase of barometric pressure in MSL of 3.4 mb for 2 consecutive hours then, compare the average of pressures 1 and 3 against pressure 2. If +/- 3.4 mb.	Change pressure 2 to 9999.	9603
Consistency check over 3 hours. If there is a decrease/increase of barometric pressure in ALT of 0.10 in for 2 consecutive hours then, compare the average of pressures 1 and 3 against pressure 2. If +/- 0.10 in.	Change altimeter 2 to 9999.	9604

SUBROUTINE CLDSQC

Cloud height and amount	Report error	Nerr
Cloud height out of range (>45,000 ft).	Change cloud height and amount at all layers to 9999.	9500
Invalid cloud height reported. (Not checked at Canadian stations.)	Change cloud height and amount at all layers to 9999.	9501
Cloud height not increasing though groups.	Change cloud height and amount at all layers to 9999.	9510
Cloud amounts not increasing through groups.	Change cloud height and amount at all layers to 9999.	9511
Missing cloud height but amount present or missing amount but height present occurring at any level. (This will not check those cases fixed by 9521).	Change cloud height and amount at all layers to 9999.	9520

Cloud report contains 4 or more layers and the first cloud height was set to 9999 due to a BUFR problem (only happened in data through Nov. 1997).	Change cloud heights at all layers to 9999.	9521
CLR or SKC reported.	Change cloud height at layer 1 to 888 and all others to 9999. **DOES NOT WRITE ANYTHING TO ERROR FILE.**	9530
POB reported in first group. (After 11/10/97 this is done in the program that creates the hourly tables.)	Change cloud height at layer 1 to 0.	9531
KMWN or KMWS report POB in first and second cloud amounts (represents fog in the valley below the station).	Delete the cloud height (POB) and amount (0) for the first layer. Move the data for the remaining layers down one layer and set the height and amount for layer 6 to 9999	9540
Any station (except KMWN or KMWS) reports POB in any cloud amount besides layer 1.	Change cloud height and amount at all layers to 9999.	9541

SUBROUTINE WNDQC

Wind speed, direction, and gusts	Report error	Nerr
Wind direction not to the nearest 10 degrees.	Change wind speed, direction and gust to 9999.	9401
Wind gust not greater than or equal to 10 kts and not between (wind speed + 3 kts) and (wind speed + 40 kts).	Change wind gust to 9999.	9402
Calm report does not have 0 for all three wind components	Change wind speed, direction, and gust to 9999.	9403
Variable wind direction (-9) reported.	Change wind direction to 990. **DOES NOT WRITE ANYTHING TO ERROR FILE.**	Nothing

SUBROUTINE VISWXQC

Visibility and present weather	Report error	Nerr
Invalid visibility value reported. (Not checked at Canadian and Mexican stations.)	Change the visibility to 9999.	9300

If the visibility is between 7 and 10 miles and the present weather group is an obstruction to vision.	Change the visibility to 6 miles.	9310
If the visibility is greater than 10 miles and the present weather is something that would reduce the visibility.	Change the visibility and all present weather groups to 9999.	9312/ 9212
If visibility is greater than ½ mile and the present weather is FG.	Change the present weather group to BR.	9210
If visibility is less than or equal to ½ mile and the present weather is BR.	Change the present weather group to FG.	9211
MANU or AO2 reports visibility less than or equal to 6 miles but doesn't report present weather. (Not checked at Canadian and Mexican stations.)	Change the present weather groups to 9999.	9213
Intensity of snow/drizzle reported does not match visibility reported, and no other weather was reported.	Change the present weather to the proper intensity as dictated by the visibility.	9214
Intensity of ice pellets reported does not match visibility reported, and no other weather was reported.	Change the present weather to the proper intensity as dictated by the visibility.	9215
If the order of the present weather groups is incorrect.	Correct order.	9220
If VCSH is reported with any other precipitation occurring.	Eliminate VCSH.	9221
If thunderstorm with precipitation is reported and is accompanied by thunderstorm with hail.	Change groups so that they indicate just one thunder qualifier.	9222
If duplicate reports of the same precipitation type.	Keep the first report and change all other present weather groups to 9999.	9223
If liquid precipitation is reported and temperature is below 30 degrees.	Change present weather groups to 9999.	9230
If freezing precipitation or ice crystals reported with temperature greater than 40 degrees.	Change present weather groups to 9999.	9231
If frozen precipitation or ice pellets reported with temperature greater than 44 degrees.	Change present weather groups to 9999.	9232

Blowing phenomenon is reported, but there isn’t enough wind to blow things around (wind less than 9 knots without a gust reported).	Change blowing present weather group and any after that to 9999.	9240
MANU or AO2 reports OB but does not report any present weather.	Change present weather groups to 9999.	9250
MANU reports clear sky but also reports present weather (except for IC).	Change present weather groups and cloud amount layer 1 to 9999.	9251

<u>SUBROUTINE CNVTWX</u>

Present weather, station type, and cloud amount	Report error	Nerr
Weather element not recognized.	Change present weather group to 9999.	9101
Station type not recognized.	Change station type to 9.	9102
Cloud amount not recognized.	Change cloud amount to 9999.	9103

Appendix II TO CHAPTER 13

Description of the Cloud-to-Ground Lightning Data Provided by GHRC

A. Introduction

This appendix contains information on cloud-to-ground lightning data produced by the U.S. National Lightning Detection Network (NLDN) and obtained by TDL from NASA's Global Hydrology Resource Center (GHRC). The cloud-to-ground lightning product, the file format, and pertinent scientific references are provided here. The information was obtained from a README file written by GHRC.

The NLDN is a commercial lightning detection network operated by Global Atmospherics, Inc. (GAI), of Tucson, Arizona. The network is comprised of 106 sensors connected to a central processor that records the time, polarity, signal strength, and number of strokes of each cloud-to-ground lightning flash detected over the United States to a distance of about 400 km beyond the border. A combination of time of arrival and direction finding technology is used to locate each flash. The NLDN uses 47 Advanced Lightning Direction Finders (ALDFs) in combination with 59 LPATS III electric field sensors. Depending on the location within the network, GAI claims a location accuracy of 500 meters, with a detection probability between 80-90 percent, varying by region. These data are ingested in real-time and stored in a raw data file. Distribution of the data is restricted by request of GHRC.

B. File Format for the Raw Cloud-to-Ground Data

The raw data file is in ASCII format. The raw data file up to January 12, 1995, is in extended format (see below). However, the raw data for dates after January 12, 1995, is generally in a different format, called the standard format (also, see below). The temporal resolution of the standard format data is in seconds while the temporal resolution of the extended format is in milliseconds. There may be times when the GHRC receives raw data in the extended format from GAI and those data will have a resolution of milliseconds. The GHRC does not have control over which format of data is received and archived; therefore, users should be prepared for both formats of data in the files.

C. Flash Format Information For Extended Format

This section provides information on the format of the flash data for the extended format. The fields that comprise the extended format are listed with their associated lengths. A sample line from a data file in extended format follows.

Sample data line:
```
07/22/94 00:00:00.934 37.480 -111.591 -293.2 2 126 2.0 5.0 1
```

Field name	Length of field	Example
Date	9	07/22/94
Time (UTC)	13	00:00:00.934
Latitude (degrees)	7	37.480
Longitude (degrees)	9	-111.591

Signal Strength (kA)	8	-293.2
Multiplicity (Strokes/flash)	3	2
Ellipse Angle (degrees)	4	126
Semi-major axis (km)	5	2.0
Eccentricity	5	5.0
Chi-square	3	1

The ellipse angle, semi-major axis, and eccentricity describe an error ellipse within which there is a 50% probability that the flash occurred. The error ellipse is described by its semi-major axis, eccentricity (ratio of semi-major to semi-minor axis), and the orientation of the semi-major axis (in degrees relative to north, i.e., North = 0.0 degrees; clockwise). The lightning flash is located at the center of the ellipse. The Chi-square value is a measure of how well the direction finder antennae angles agree. Chi-square values between 0.0 and 3.0 are considered good, while values from 3.0 to 10.0 are acceptable.

D. Flash Format Information for Standard Format

This section provides information on the format of the flash data for the STANDARD format. The fields that comprise the standard format are listed with their associated lengths. A sample line from a data file in standard format follows.

Sample data line:
01/30/95 00:02:33 24.347 -82.469 -80.0 1

Field name	Length of field	Example
Date	9	01/30/95
Time (UTC)	9	00:02:33
Latitude (degrees)	7	24.347
Longitude (degrees)	9	-82.469
Signal Strength(kA)	8	-80.0
Multiplicity (Strokes/flash)	3	1

<u>File Names</u>

A raw data file is produced for each day of the year. The naming convention for the raw data file is: Nflash**yy.ddd**_daily.lit.raw, where **yy** is the year and **ddd** is the day of year.

<u>References</u>

Cummins, K. L., M. J. Murphy, E. A. Bardo, W. L. Hiscox, R. B. Pyle, and A. E. Pifer, 1998: A combined TOA/MDF technology upgrade of the U. S. National Lightning Detection Network. <u>J. Geophys. Res.</u>, 103, 9035-9044.

Idone, V. P., D. A. Davis, P. K. Moore, Y. Wang, R. W. Henderson, M. Ries, and P. F. Jamason, 1998: Performance evaluation of the U. S. National Lightning Detection Network in eastern New York, Part 1, Detection Efficiency. <u>J. Geophys. Res.</u>, 103, 9045-9056.

14. CONSTANT FILES

A. Introduction

These external random access files (see Chapter 7) contain data that generally are accessed in a non-sequential, hence random, fashion. Examples that are possible in the MOS-2000 system include files that contain "true" constants (climatic information), such as monthly relative frequencies, daily normal max/min temperatures, or annual rainfall amounts; 1- and 2-dimensional linearization constants used in U201 to create predictor variables; and probability thresholds used in U710 or U910 to create categorical forecasts from a distribution of probability forecasts. Random access files can also contain forecasts made by U900 or U910. This chapter discusses only the files containing climatic information and probability thresholds. At this time, no linearization constant file exists; however, the method of creation would be the same.

B. Format

Chapter 7 describes the format and structure of the TDL external random access files.

C. Preparation

Chapter 7 also discusses the software used to create and access these files. Essentially, U350 creates a file in which the master key record and one empty data key record are written. Either U351 or U352 is then used to define the stations for which data can be contained in the file and to write records into the file. U351 is used when the information to be stored in the file is available in ASCII format. U351 will read the ASCII data, pack them into the MOS-2000 record format, and then write the record into the constant file. U352 is used when the information to be stored in the file is already packed in the MOS-2000 format. At different times, either U351 or U352 can be used to write records into the same constant file. U353 can be used to copy one random access file to another, omitting records as desired. U354 can be used to inventory a random access file. The programs U35x (x non zero) are generally for vector data; U361 can be used for gridpoint data. Fortran Unit No. 44 is reserved for gridpoint data; Nos. 45-49 are reserved for vector data.

D. Variable Identification for Climatic Information

As discussed in Chapter 4, the CCCFFF convention for the first ID word (ID(1)) represents the time interval and the meteorological element for which the climatic information is valid. In this instance, CCC equals $4C_2C_3$ where C_2 defines the time interval and C_3 generally refers to the digit expressing the class of meteorological observation. Thus, for example,

C_2 = 0, if there is no time dependency
C_2 = 1, if daily values are available
C_2 = 2, if values every 5 days are available (max and min)
C_2 = 3, if values are monthly and are to be interpolated to day of year
C_2 = 4, if values are 2-seasonal

In a similar fashion, C_3 takes on the values of 1, 2, 3, 4, 7, or 8 according to whether the meteorological element is classified as mass (pressure), temperature, moisture, wind, thunderstorm or severe weather, or general sky/weather conditions (clouds, ceiling, visibility, etc.) respectively.

The FFF represents the family into which the variable falls and the time-interval to which the variable pertains (e.g., 1 hour, 3 hours, 6 hours, etc., in the case of precipitation). Thus, the FFF further defines the variable in terms of the actual observation or processing done.

Two options exist for the remainder of the IDs. The first option was defined originally, and the second option was defined in March 2004 to reduce the number of different CCCFFFs needed and to better meet other needs. The accessing routine CONST has been modified to accommodate both. It is expected all future ID definitions will follow option 2. The two options are defined below.

Option 1

In addition to FFF defining the family and the time-interval to which the variable pertains (e.g., 1 hour, 3 hours, 6 hours, etc.), it also defines the hour at which the constant value is valid.

The last part (BDD) of the identifier for the climatic normals will generally be 000, although relative frequencies of certain model variables are not precluded. In the case of a model relative frequency, the DD would equal the model number.

The second word of the identifier (ID(2) = VLLLLUUUU) indicates in the LLLL position the season, month, or day for which the climatic variable is valid. Thus, ID(2) = 01xxxUUUU (the "1" is arbitrary and essentially defines this as the first option), where xxx = 001, 002, ..., 012 for monthly values (representing January, February, ..., December), xxx = 017 for the warm season, xxx = 018 for the cool season, and so on. For daily values, ID(2) = 00xxxUUUU, (note the change of the 2nd digit from "1" to "0") where xxx defines the day of the year (005 = January 5, 030 = January 30, 035 = February 4, and so on). UUUU can be used to represent a level or the top, say, of a layer (or bin) when dealing with, for instance, cloud heights and amounts.

The third word of the identifier equals 0. The fourth word of the identifier equals 0 for a climatic normal of a continuous (or quasi-continuous) variable like temperature. For a relative frequency of an event defined relative to some breakpoint, the fourth word contains the breakpoint or threshold formatted according to the MOS-2000 standard. Given that ID(4) = WXXXXYYISG, the value WXXXXYY is non-zero. In other words, if one were considering the relative frequency of the occurrence of 0.01 inches or more of precipitation, then WXXXXYY = 0950052 represents the threshold of 0.01 inches, and ID(4) = 0950052000. Table 14.1 contains the first and fourth ID words for constants as examples that are currently available or are being used as possible predictors in forecast equations. Because only one threshold or breakpoint is carried along with the identifier, the relative frequency is not unambiguously defined by the breakpoint. The user must know whether the breakpoint was used in a cumulative from above, cumulative from below, or discrete fashion when the frequencies were developed in order to know the complete definition.

Option 2:

The FFF defines the time-interval of the variable as it does in option 1, but not the valid hour; the latter is done in Word 3.

As with option 1, the last part (BDD) of the identifier for the climatic normals will generally be 000, although relative frequencies of certain model variables are not precluded. In the case of a model relative frequency, the DD would equal the model number.

For the second word of the identifier, the only difference between Option 1 and Option 2 is that the "0" for daily values is replaced by "2" and the "1" for other than daily values is replaced by "3". This change of values from Option 1 to Option 2 essentially signals that the "HH" in word 3 (see below) is used for Option 2.

In the third word, the "HH" position is used for the ending hour for which the constant is valid. For example, for a climatic value valid at 05Z, HH = 05; for a climatic variable computed over the hours 04Z to 05Z, HH would also be 05; for a climatic variable computed over the hours 03Z, 04Z, and 05Z, HH would also be 05. The difference between 1-h, 2-h and 3-h values is captured by different FFFs. As with Option 1, UUUU can be used to represent a level or the top, say, of a layer (or bin) when dealing with, for instance, cloud heights and amounts.

The fourth word is the same as in option 1.

E. Variable Identification for Threshold Information

As of the present, no thresholds yet exist in the TDL constant files. The convention for identifying them, however, has been established. The ID structure is as follows:

ID(1) = CCCFFFBDD = 8xyyyyBDD, where x = 0 for cumulative probability categories; x = 1 for discrete categories, and the thresholds have been calculated as if cumulative from above; x = 2 for discrete categories, and the thresholds have been calculated as if cumulative from below; and yyyy = the normal identifier of the meteorological variable.

ID(2) = $0L_1L_1L_2L_2$UUUU, where L_1L_1 = forecast cycle for which the thresholds are valid (for example, 12 for the 12Z forecast cycle), and L_2L_2 = coded value indicating the season for which the thresholds are valid. For example L_2L_2 = 13 for March 1 - May 31; = 14 for June 1 - August 31; = 15 for September 1 - November 30; = 16 for December 1 - February 29; = 17 for April 1 - September 30; = 18 for October 1 - March 31; = 19 for January 1 - December 31, and so. UUUU can be used to represent a level or the top, say, of a layer (or bin) when dealing with, for instance, cloud heights and amounts.

ID(3) = 000000ttt, where ttt = forecast projection (of predictand).

ID(4) = WXXXXYYISG extracted from the predictand.

F. Existing Constant Files

At this point, each developer has her/his own constant files. When the various constant records are placed in one central file prior to operational implementation, the file will be stored as a file within the subdirectory:

/home21/tdllib/constants

Table 14.1. Examples of variable identifiers of constants currently in use within the MOS-2000 system. The first and fourth id words are given along with a simple explanation. The second id word contains the time indicator as discussed above. If the explanation describes the variable as a monthly relative frequency (RF), then the relative frequencies are available for each month of the year. Abbreviations in the table are Mon. = monthly, precip. = precipitation, amt. = amount, cond. = conditional, svr. wx. = severe weather, and occur. = occurrence.

ID(1)	ID(4)	Description
422006000	0000000000	Normal calendar day maximum temperature (1961-90 normals)
422016000	0000000000	Normal calendar day minimum temperature (1961-90 normals)
433210000	0950052000	Mon. RF of precip. amt. $\geq$ 0.01 in. for 18-00Z period
433210000	0950051000	Mon. RF of precip. amt. $\geq$ 0.10 in. for 18-00Z period
433210000	0250000000	Mon. RF of precip. amt. $\geq$ 0.25 in. for 18-00Z period
433211000	0950052000	Mon. RF of precip. amt. $\geq$ 0.01 in. for 00-06Z period
433211000	0950051000	Mon. RF of precip. amt. $\geq$ 0.10 in. for 00-06Z period
433211000	0250000000	Mon. RF of precip. amt. $\geq$ 0.25 in. for 00-06Z period
433212000	0950052000	Mon. RF of precip. amt. $\geq$ 0.01 in. for 06-12Z period
433212000	0950051000	Mon. RF of precip. amt. $\geq$ 0.10 in. for 06-12Z period
433212000	0250000000	Mon. RF of precip. amt. $\geq$ 0.25 in. for 06-12Z period
433213000	0950052000	Mon. RF of precip. amt. $\geq$ 0.01 in. for 12-18Z period
433213000	0950051000	Mon. RF of precip. amt. $\geq$ 0.10 in. for 12-18Z period
433213000	0250000000	Mon. RF of precip. amt. $\geq$ 0.25 in. for 12-18Z period
433220000	0950052000	Mon. RF of precip. amt. $\geq$ 0.01 in. for 12-00Z period
433220000	0950051000	Mon. RF of precip. amt. $\geq$ 0.10 in. for 12-00Z period
433220000	0250000000	Mon. RF of precip. amt. $\geq$ 0.25 in. for 12-00Z period
433221000	0950052000	Mon. RF of precip. amt. $\geq$ 0.01 in. for 18-06Z period
433221000	0950051000	Mon. RF of precip. amt. $\geq$ 0.10 in. for 18-06Z period
433221000	0250000000	Mon. RF of precip. amt. $\geq$ 0.25 in. for 18-06Z period

ID(1)	ID(4)	Description
433222000	0950052000	Mon. RF of precip. amt. ≥ 0.01 in. for 00-12Z period
433222000	0950051000	Mon. RF of precip. amt. ≥ 0.10 in. for 00-12Z period
433222000	0250000000	Mon. RF of precip. amt. ≥ 0.25 in. for 00-12Z period
433223000	0950052000	Mon. RF of precip. amt. ≥ 0.01 in. for 06-18Z period
433223000	0950051000	Mon. RF of precip. amt. ≥ 0.10 in. for 06-18Z period
433223000	0250000000	Mon. RF of precip. amt. ≥ 0.25 in. for 06-18Z period
433240000	0950052000	Mon. RF of precip. amt. ≥ 0.01 in. for 00-00Z period
433240000	0950051000	Mon. RF of precip. amt. ≥ 0.10 in. for 00-00Z period
433240000	0250000000	Mon. RF of precip. amt. ≥ 0.25 in. for 00-00Z period
433241000	0950052000	Mon. RF of precip. amt. ≥ 0.01 in. for 06-06Z period
433241000	0950051000	Mon. RF of precip. amt. ≥ 0.10 in. for 06-06Z period
433241000	0250000000	Mon. RF of precip. amt. ≥ 0.25 in. for 06-06Z period
433242000	0950052000	Mon. RF of precip. amt. ≥ 0.01 in. for 12-12Z period
433242000	0950051000	Mon. RF of precip. amt. ≥ 0.10 in. for 12-12Z period
433242000	0250000000	Mon. RF of precip. amt. ≥ 0.25 in. for 12-12Z period
433243000	0950052000	Mon. RF of precip. amt. ≥ 0.01 in. for 18-18Z period
433243000	0950051000	Mon. RF of precip. amt. ≥ 0.10 in. for 18-18Z period
433243000	0250000000	Mon. RF of precip. amt. ≥ 0.25 in. for 18-18Z period
433624000	0000000000	Mon. RF of precip. amt. ≥ 0.01 in. for 00-12Z period (AEV sample)
433626000	0000000000	Mon. RF of precip. amt. ≥ 0.01 in. for 12-00Z period (AEV sample)
437310000	0950000000	Mon. RF of cond. svr. wx. for 18-00Z period
437311000	0950000000	Mon. RF of cond. svr. wx. for 00-06Z period
437312000	0950000000	Mon. RF of cond. svr. wx. for 06-12Z period
437313000	0950000000	Mon. RF of cond. svr. wx. for 12-18Z period
437320000	0950000000	Mon. RF of cond. svr. wx. for 12-00Z period
437321000	0950000000	Mon. RF of cond. svr. wx. for 18-06Z period
437322000	0950000000	Mon. RF of cond. svr. wx. for 00-12Z period
437323000	0950000000	Mon. RF of cond. svr. wx. for 06-18Z period
437330000	0950000000	Mon. RF of cond. svr. wx. for 00-00Z period

ID(1)	ID(4)	Description
437331000	0950000000	Mon. RF of cond. svr. wx. for 06-06Z period
437332000	0950000000	Mon. RF of cond. svr. wx. for 12-12Z period
437333000	0950000000	Mon. RF of cond. svr. wx. for 18-18Z period
437390000	0950000000	Mon. RF of thunderstorms for 18-00Z period
437391000	0950000000	Mon. RF of thunderstorms for 00-06Z period
437392000	0950000000	Mon. RF of thunderstorms for 06-12Z period
437393000	0950000000	Mon. RF of thunderstorms for 12-18Z period
437400000	0950000000	Mon. RF of thunderstorms for 12-00Z period
437401000	0950000000	Mon. RF of thunderstorms for 18-06Z period
437402000	0950000000	Mon. RF of thunderstorms for 00-12Z period
437403000	0950000000	Mon. RF of thunderstorms for 06-18Z period
437410000	0950000000	Mon. RF of thunderstorms for 00-00Z period
437411000	0950000000	Mon. RF of thunderstorms for 06-06Z period
437412000	0950000000	Mon. RF of thunderstorms for 12-12Z period
437413000	0950000000	Mon. RF of thunderstorms for 18-18Z period
438000000	0950001000	Mon. RF of ceiling height < 1000 ft at 00Z
438003000	0950001000	Mon. RF of ceiling height < 1000 ft at 03Z
438006000	0950001000	Mon. RF of ceiling height < 1000 ft at 06Z
438009000	0950001000	Mon. RF of ceiling height < 1000 ft at 09Z
438012000	0950001000	Mon. RF of ceiling height < 1000 ft at 12Z
438015000	0950001000	Mon. RF of ceiling height < 1000 ft at 15Z
438018000	0950001000	Mon. RF of ceiling height < 1000 ft at 18Z
438021000	0950001000	Mon. RF of ceiling height < 1000 ft at 21Z
438300000	0150001000	Mon. RF of clear skies at 00Z
438303000	0150001000	Mon. RF of clear skies at 03Z
438306000	0150001000	Mon. RF of clear skies at 06Z
438309000	0150001000	Mon. RF of clear skies at 09Z
438312000	0150001000	Mon. RF of clear skies at 12Z
438315000	0150001000	Mon. RF of clear skies at 15Z
438318000	0150001000	Mon. RF of clear skies at 18Z

ID(1)	ID(4)	Description
438321000	0150001000	Mon. RF of clear skies at 21Z
438300000	0700001000	Mon. RF of few, scattered, or broken skies at 00Z
438303000	0700001000	Mon. RF of few, scattered, or broken skies at 03Z
438306000	0700001000	Mon. RF of few, scattered, or broken skies at 06Z
438309000	0700001000	Mon. RF of few, scattered, or broken skies at 09Z
438312000	0700001000	Mon. RF of few, scattered, or broken skies at 12Z
438315000	0700001000	Mon. RF of few, scattered, or broken skies at 15Z
438318000	0700001000	Mon. RF of few, scattered, or broken skies at 18Z
438321000	0700001000	Mon. RF of few, scattered, or broken skies at 21Z
438300000	0999905000	Mon. RF of overcast skies at 00Z
438303000	0999905000	Mon. RF of overcast skies at 03Z
438306000	0999905000	Mon. RF of overcast skies at 06Z
438309000	0999905000	Mon. RF of overcast skies at 09Z
438312000	0999905000	Mon. RF of overcast skies at 12Z
438315000	0999905000	Mon. RF of overcast skies at 15Z
438318000	0999905000	Mon. RF of overcast skies at 18Z
438321000	0999905000	Mon. RF of overcast skies at 21Z
438501000	0350001000	Mon. RF of daily cond. occur. of freezing precip.
438501000	0450001000	Mon. RF of daily cond. occur. of frozen precip.
438502000	0350001000	Mon. RF of cond. occur. of freezing precip. (12-00Z)
438502000	0450001000	Mon. RF of cond. occur. of frozen precip. (12-00Z)
438503000	0350001000	Mon. RF of cond. occur. of freezing precip. (18-06Z)
438503000	0450001000	Mon. RF of cond. occur. of frozen precip. (18-06Z)
438504000	0350001000	Mon. RF of cond. occur. of freezing precip. (00-12Z)
438504000	0450001000	Mon. RF of cond. occur. of frozen precip. (00-12Z)
438505000	0350001000	Mon. RF of cond. occur. of freezing precip. (06-18Z)
438505000	0450001000	Mon. RF of cond. occur. of frozen precip. (06-18Z)
438506000	0350001000	Mon. RF of cond. occur. of freezing precip. (03-15Z)
438506000	0450001000	Mon. RF of cond. occur. of frozen precip. (03-15Z)
438507000	0350001000	Mon. RF of cond. occur. of freezing precip. (15-03Z)

ID(1)	ID(4)	Description
438507000	0450001000	Mon. RF of cond. occur. of frozen precip. (15-03Z)
438508000	0350001000	Mon. RF of cond. occur. of freezing precip. (09-21Z)
438508000	0450001000	Mon. RF of cond. occur. of frozen precip. (09-21Z)
438509000	0350001000	Mon. RF of cond. occur. of freezing precip. (21-09Z)
438509000	0450001000	Mon. RF of cond. occur. of frozen precip. (21-09Z)

15. MOS EQUATIONS

A. Format

The equations are produced by U600, U602, or U605 and written to an ASCII file. The format is as follows, where the variable names are those used in U600, U602 and U605:

Record Type 1

OPTION 1 - Format (' 'A60,' CYCLE =',I2.2)

EQNNAM - The name of the file, including a path if desired is written by U600, U602, or U605. This is written to provide a margin of safety. File names can be accidently changed. The file name duplicated as the first record allows verification either visually or with software. U700 and U705 matche this file name with the file name on the .CN control file; they must match.

NCYCLE - The cycle time (taken from the first date of the U600, U602,or U605 run) the equations were developed on is put into columns 70 and 71. This is additional proof, in addition to EQNNAM, of where the equations came from. This feature was added March 2002, and will not disturb follow on programs like U700, U705, U900, or U905.

OPTION 2 Format (1XI4,1XA3,1XI4,1XI4)

ICYCLE - The cycle, or time, to which the equations apply. This would normally be 0000 or 1200; other times may be used such as 0600 and 1800.

UTC - Three ASCII characters to denote time. Normally, these would be 'UTC'.

MONDAS - The starting month and day to which the equations apply. For instance 0301 would mean the equations apply starting on March 1.

MONDAE - The ending month and day to which the equations apply. For instance 0931 would mean the equations apply ending on September 31.

For equations used in operations, the file name is not used, but rather the date/time for which the equations are being evaluated is checked with ICYCLE, MONDAS, and MONDAE to assure the forecasts are made from the correct equations. This is done by U900 or U905.

Record Type 2 - Format (' 'I4)

NTAND - The number of equations (predictands) for each station, or group of stations, in this file.

Record Type 3 - Format (' 'I9.9,2I10,I11,I8)

ID(J,N) - Four ID's (J=1,4) to identify each of the equations (predictands) (N=1,NTAND) and the associated ICAT value, the latter specifying whether or not some post-processing is to be done, and if so, what. The first 4 words are the MOS-2000 ID for the variable.

The following groups of records are repeated for each station or group of stations for which equations are provided.

Record Type 4 - Format (14(' 'A8))

CCALL(K) - The 8-character call letters for each station to which the following equations apply (K=1,LSTA), followed by the terminator '99999999'. The terminator makes it easy to edit the station list without counting the number of stations. Blanks can occur in the station list.

Record Type 5 - Format (' 'I4)

NTERMS - The number of terms (predictors) in each equation.

Record Type 6 - Format (' 'I9.9,2I10,I11)

LP(J,L) - The standard MOS-2000 four-word ID's (J=2,5) of each predictor (L=1,NTERMS).

OUTPUT FROM U600 OR U602

Record Type 7 - Format (' '10E13.7)

AVG(N) - The predictand average for each of the equations (N=1,NTAND).

Record Type 8 - Format (' '10E13.7)

CORR(N) - The multiple correlation of the predictand with all the predictors for each of the equations (N=1,NTAND).

Record Type 9 - Format (' '10E13.7)

A(N) - The equation constant for each of the equations (N=1,NTAND).

Record Type 10 - Format (' '10E13.7)

P(L,N) - The coefficients are written for each equation (N=1,NTAND), then the format is used again for each term (L=1,NTERMS). That is, the write statement is repeated NTERMS times. In this way, the predictand averages, multiple correlations, equation constants, and coefficients will be in a column. If more than 10 equations are present, then there will be two or more rows for each.

OUTPUT FROM U605

Record Type 7 - Format (‘ ’,8E15.9)

AVG(N) - The predictand average for each of the equations (N=1,NTAND).

Record Type 8 - Format (‘ ’,8E15.9)

CORR(N) - The multiple correlation of the predictand with all the predictors for each of the equations (N=1,NTAND).

Record Type 9 - Format (‘ ’,8E15.9)

ESS(N) - The standard errors generated by the regression program for each of the equations (N=1,NTAND).

Record Type 10 - Format (‘ ’,8E15.9)

A(N) - The equation constant for each of the equations (N=1,NTAND).

Record Type 11 - Format (‘ ’,8E15.9)

P(L,N) - The coefficients are written for each equation (N=1,NTAND), then the format is used again for each term (L=1,NTERMS). That is, the write statement is repeated NTERMS times. In this way, the predictand averages, multiple correlations, equation constants, and coefficients will be in a column. If more than 10 equations are present, then there will be two or more rows for each.

Record Type 12 - Format (‘ ’,8E15.9)

P(L,N) - The inverse of the cross product NTERMS+1 X NTERMS+1 matrix (that could have been) used in developing the equations. (This is used with the individual predictor vector and ESS( ) above to determine the error of estimate of the specific forecast.)

ALL EQUATIONS

This complete set of records can be repeated. For instance, if temperature equations were simultaneously derived for projections 6 to 12 hours and another set for 15 to 24 hours, the predictors would be different. These two sets of equations could be sequential on this file, even the file name (OPTION 1) or beginning and ending dates (OPTION 2) being repeated.

B. Preparation and Use

The equations are prepared originally by U600, u602, or U605 in ASCII format for easy reading. The first character of each record is a blank to facilitate possible printing. Records are of such length that they can be easily viewed or printed. The format is such that the station lists can be changed rather easily. User programs should read this station list with subroutine RDC so that blanks can occur in the list (e.g., each station could be a separate record for easy reading); subroutine RDSTAD shows an example of RDC use. Also, if development of equations for stations must be split over runs, the first three record types can be edited out of all of the resulting files except the first, and the files copied together into one larger file.

If further packing or making a binary file is found to be necessary, that can be done from the ASCII files. On the Unix platforms, a fairly large number of files can be dealt with rather easily.

Because a forecast made from an equation is the result of several terms, some of which may be large and to a large extent cancel others, several places of precision in the coefficients, etc., are desirable. Five places (i.e., E12.5) gave less than desirable results. Quite likely, 7 places gives about -3-place accuracy in the forecasts (the 4th place may be in error). This should be sufficient for the purpose to which the forecast will be put, and commensurate with the accuracy of the predictors. For U705, two matrix multiplications are performed, and it was thought higher accuracy was desirable for some variables.

In this regard, it has been noticed that in U700 FCST71 may give sightly different results than FCST72 just because the terms are evaluated and coalesced in a different order, and rounding produces different results. This small difference is not important (and neither can be deemed more correct than the other), but the user might be perplexed as to why different results might occur with exactly the same inputs (except that for a particular day, FCST71 would be used on one run and FCST72 used on the other run). The same discrepancy can occur with U705, which uses FCST75 and FCST76.

16. FORECASTS AND MATCHING OBSERVATIONS ARCHIVE

A. Format

The MOS and AEV (AFOS- and AWIPS-Era Verification) forecasts are stored in the same format as most other vector data as defined in Chapter 6 "Sequential Data Archive Files." Each forecast element, for each issue time and projection is a TDLPACK record. In addition to forecasts, some observations are also present. One or more "directory" records are present in each file, each one except the first preceded by a "trailer" record.

B. MOS Forecasts

The 4-word IDs, the scaling factor used in packing the data (which should agree with the Variable Constants File described in Chapter 11), the plain language description packed with the data (a few of these are truncated to fit on a line), and finally the Code and Category numbers used in the MOS system by which the forecasts were produced (described in TDL Office Note 74-14), are given in the table below. The ttt is the tau (projection) and will vary depending on the projections for which the forecasts were made. The model number DD is 06, 08, or 09 because the forecasts were made from the NGM, AVN, or MRF model, respectively (see Chapter 4). The observations in these files are those used as predictors in the operational forecast equations.

MOS-2000 System 4-Word Id	Scaling	Plain Language	Ncode	Ncat
CCCFFFBDD VLLLLUUUU TRROHHτττ ISG				
202000006 000000000 000000ttt 000	2	NGM MOS SFC TEMP	298	1
202001006 000000000 000000ttt 000	2	NGM MOS MAX TEMP	208	1
202001008 000000000 000000ttt 000	2	AVN DAYTIME MAX	203	1
202001009 000000000 000000ttt 000	2	MRF DAYTIME MAX	201	1
202011006 000000000 000000ttt 000	2	NGM MOS MIN TEMP	218	1
202011008 000000000 000000ttt 000	2	AVN NIGHTTIME MIN	213	1
202011009 000000000 000000ttt 000	2	MRF NIGHTTIME MIN	211	1
203000006 000000000 000000ttt 000	1	RELATIVE HUMIDITY	678	1
203100006 000000000 000000ttt 000	2	NGM MOS SFC DEW PT TEMP	668	1
203101006 000000000 000000ttt 000	2	DEW POINT DEPRESSION	688	1
203210006 000010000 000000ttt 000	3	NGM MOS 6-H QPF CAT 1	708	1
203210006 000020000 000000ttt 000	3	NGM MOS 6-H QPF CAT 2	708	2
203210006 000030000 000000ttt 000	3	NGM MOS 6-H QPF CAT 3	708	3
203210006 000040000 000000ttt 000	3	NGM MOS 6-H QPF CAT 4	708	4
203210006 000050000 000000ttt 000	3	NGM MOS 6-H QPF CAT 5	708	5
203210006 000060000 000000ttt 000	0	NGM MOS 6-H QPF BEST CAT	708	6
203210006 000070000 000000ttt 000	2	NGM MOS 6-H QPF EXP VAL	708	7
203220006 000010000 000000ttt 000	3	NGM MOS 12-H QPF CAT 1	718	1
203220006 000020000 000000ttt 000	3	NGM MOS 12-H QPF CAT 2	718	2
203220006 000030000 000000ttt 000	3	NGM MOS 12-H QPF CAT 3	718	3
203220006 000040000 000000ttt 000	3	NGM MOS 12-H QPF CAT 4	718	4
203220006 000050000 000000ttt 000	3	NGM MOS 12-H QPF CAT 5	718	5
203220006 000060000 000000ttt 000	3	NGM MOS 12-H QPF CAT 6	718	6
203220006 000070000 000000ttt 000	0	NGM MOS 12-H QPF BEST CAT	718	7
203220006 000080000 000000ttt 000	2	NGM MOS 12-H QPF EXP VAL	718	8
203500006 000010000 000000ttt 000	3	NGM MOS 6-H POP	608	1

203505006	000010000	000000ttt	000	3	NGM MOS 12-H POP	618	1
203505008	000010000	000000ttt	000	3	AVN 12-H POP	613	1
203505009	000010000	000000ttt	000	3	MRF 12-H POP	611	1
203510009	000010000	000000ttt	000	3	MRF 24-H POP	621	1
204010006	000000000	000000ttt	000	2	NGM MOS U-WIND	568	1
204016006	000000000	000000ttt	000	2	S.R.L. U-WIND	588	1
204110006	000000000	000000ttt	000	2	NGM MOS V-WIND	568	2
204116006	000000000	000000ttt	000	2	S.R.L. V-WIND	588	2
204200006	000000000	000000ttt	000	2	NGM MOS WIND DIRECTION	568	3
204210006	000000000	000000ttt	000	2	NGM MOS WIND SPEED	518	1
204211006	000000000	000000ttt	000	2	NGM MOS INFL WIND SPEED	519	1
204215009	000000000	000000ttt	000	2	MRF MEAN WIND SPEED	541	1
204216006	000000000	000000ttt	000	2	S.R.L. WSPD	548	1
204226006	000000000	000000ttt	000	2	S.R.L. TURBUENCE/AZ	598	1
204236006	000000000	000000ttt	000	2	S.R.L. TURBUENCE/EL	698	1
207300006	000010000	000000ttt	000	3	NGM MOS 6-H TSTM	348	1
207301006	000010000	000000ttt	000	3	NGM MOS 6-H SVR WX	358	1
207302006	000010000	000000ttt	000	3	NGM MOS 12-H TSTM	368	1
207303006	000010000	000000ttt	000	3	NGM MOS 12-H SVR WX	378	1
207304006	000010000	000000ttt	000	3	NGM MOS 24-H TSTM	388	1
207305006	000010000	000000ttt	000	3	NGM MOS 24-H SVR WX	398	1
208000006	000010000	000000ttt	000	3	NGM MOS CIG CAT 1	108	1
208000006	000020000	000000ttt	000	3	NGM MOS CIG CAT 2	108	2
208000006	000030000	000000ttt	000	3	NGM MOS CIG CAT 3	108	3
208000006	000040000	000000ttt	000	3	NGM MOS CIG CAT 4	108	4
208000006	000050000	000000ttt	000	3	NGM MOS CIG CAT 5	108	5
208000006	000060000	000000ttt	000	3	NGM MOS CIG CAT 6	108	6
208000006	000070000	000000ttt	000	3	NGM MOS CIG CAT 7	108	7
208000006	000080000	000000ttt	000	0	NGM MOS CIG BEST CAT	108	8
208100006	000010000	000000ttt	000	3	NGM MOS VIS CAT 1	158	1
208100006	000020000	000000ttt	000	3	NGM MOS VIS CAT 2	158	2
208100006	000030000	000000ttt	000	3	NGM MOS VIS CAT 3	158	3
208100006	000040000	000000ttt	000	3	NGM MOS VIS CAT 4	158	4
208100006	000050000	000000ttt	000	3	NGM MOS VIS CAT 5	158	5
208100006	000060000	000000ttt	000	0	NGM MOS VIS BEST CAT	158	6
208200006	000010000	000000ttt	000	3	NGM MOS CIG/VIS CAT 1	128	1
208200006	000020000	000000ttt	000	3	NGM MOS CIG/VIS CAT 2	128	2
208200006	000030000	000000ttt	000	3	NGM MOS CIG/VIS CAT 3	128	3
208200006	000040000	000000ttt	000	0	NGM MOS CIG/VIS BEST CAT	128	4
208250006	000010000	000000ttt	000	3	NGM MOS PROB OF BLWG PHNM	858	1
208250006	000020000	000000ttt	000	3	NGM MOS PROB OF HAZE	858	2
208250006	000030000	000000ttt	000	3	NGM MOS PROB OF FOG	858	3
208250006	000040000	000000ttt	000	3	NGM MOS PROB OF NO OBS	858	4
208250006	000050000	000000ttt	000	0	NGM MOS OBVIS BEST CAT	858	5
208300006	000010000	000000ttt	000	3	NGM MOS PROB OF CLEAR SKY	808	1
208300006	000020000	000000ttt	000	3	NGM MOS PROB OF SCAT SKY	808	2
208300006	000030000	000000ttt	000	3	NGM MOS PROB OF BRKN SKY	808	3
208300006	000040000	000000ttt	000	3	NGM MOS PROB OF OVRCST	808	4
208300006	000050000	000000ttt	000	0	NGM MOS OPAQ SKY BEST CAT	808	5
208305008	000000000	000000ttt	000	2	AVN 12-H MEAN CLOUDINESS	823	1
208305009	000000000	000000ttt	000	2	MRF 12-H MEAN CLOUDINESS	821	1
208401006	000010000	000000ttt	000	3	NGM MOS 6-H SN AMT CAT 1	428	1
208401006	000020000	000000ttt	000	3	NGM MOS 6-H SN AMT CAT 2	428	2
208401006	000030000	000000ttt	000	0	NGM MOS 6-H SN AMT BEST	428	3

208402006	000010000	000000ttt	000	3	NGM MOS 12-H SN AMT CAT 1	438	1
208402006	000020000	000000ttt	000	3	NGM MOS 12-H SN AMT CAT 2	438	2
208402006	000030000	000000ttt	000	3	NGM MOS 12-H SN AMT CAT 3	438	3
208402006	000040000	000000ttt	000	3	NGM MOS 12-H SN AMT CAT 4	438	4
208402006	000050000	000000ttt	000	0	NGM MOS 12-H SN AMT BEST	438	5
208501006	000010000	000000ttt	000	3	NGM MOS PROB OF ZP	408	1
208501006	000020000	000000ttt	000	3	NGM MOS PROB OF SN	408	2
208501006	000030000	000000ttt	000	3	NGM MOS PROB OF RN	408	3
208501006	000040000	000000ttt	000	0	NGM MOS PTYPE BEST CAT	408	4
208505008	000000000	000000ttt	000	3	AVN CPOS UNCALIBRATED	413	1
208505009	000000000	000000ttt	000	3	MRF CPOS UNCALIBRATED	411	1
208600006	000010000	000000ttt	000	3	NGM MOS PRECIP CHAR	308	1
208600006	000020000	000000ttt	000	3	NGM MOS PRECIP CHAR	308	2
208600006	000030000	000000ttt	000	3	NGM MOS PRECIP CHAR	308	3
208600006	000040000	000000ttt	000	0	NGM MOS PRECIP CHAR	308	4
209300006	000000000	000000ttt	000	2	NGM MOS INSOL KWH/M**2	778	1
209305006	000000000	000000ttt	000	3	NGM MOS DLY PCNT EXTR RAD	768	1
209310006	000000000	000000ttt	000	1	NGM MOS HOURS OF SUNSHINE	798	1
209315006	000000000	000000ttt	000	3	NGM MOS DLY PCNT POS SUN	788	1
212001008	000000000	000000ttt	000	2	AVN DAYTIME MAX CALIBR	203	2
212001009	000000000	000000ttt	000	2	MRF DAYTIME MAX CALIBR	201	2
212011008	000000000	000000ttt	000	2	AVN NIGHTTIME MIN CALIBR	213	2
212011009	000000000	000000ttt	000	2	MRF NIGHTTIME MIN CALIBR	211	2
213505008	000010000	000000ttt	000	3	AVN 12-H POP CALIBRATED	613	2
213505009	000010000	000000ttt	000	3	MRF 12-H POP CALIBRATED	611	2
213510009	000010000	000000ttt	000	3	MRF 24-H POP CALIBRATED	621	2
214215009	000000000	000000ttt	000	2	MRF MEAN WIND SPEED CALIB	541	2
218305008	000000000	000000ttt	000	2	AVN 12-H MEAN CLOUD CALIB	823	2
218305009	000000000	000000ttt	000	2	MRF 12-H MEAN CLOUD CALIB	821	2
218505008	000000000	000000ttt	000	3	AVN CPOS CALIBRATED	413	2
218505009	000000000	000000ttt	000	3	MRF CPOS CALIBRATED	411	2
224010006	000000000	000000ttt	000	2	MARINE U-WIND MPP	576	1
224011006	000000000	000000ttt	000	2	S. A. U-WIND MPP	586	1
224110006	000000000	000000ttt	000	2	MARINE V-WIND MPP	576	2
224111006	000000000	000000ttt	000	2	S. A. V-WIND MPP	586	2
224210006	000000000	000000ttt	000	2	MARINE WSPD MPP	516	1
224211006	000000000	000000ttt	000	2	MARINE WSPD MPP (INFL)	517	1
224213006	000000000	000000ttt	000	2	S. A. WSPD MPP	526	1
224214006	000000000	000000ttt	000	2	S. A. WSPD MPP (INFL)	527	1
702000000	000000000	000000ttt	000	2	OBSERVED TEMP	900	20
703100000	000000000	000000ttt	000	2	OBSERVED DEW POINT	900	7
704010000	000000000	000000ttt	000	2	OBSERVED U WIND COMPONENT	900	18
704110000	000000000	000000ttt	000	2	OBSERVED V WIND COMPONENT	900	19
704200000	000000000	000000ttt	000	0	OBSERVED WIND DIRECTION	900	16
704210000	000000000	000000ttt	000	2	OBSERVED WIND SPEED	900	17
708000000	000000000	000000ttt	000	0	OBSERVED CEILING HEIGHT	900	3
708100000	000000000	000000ttt	000	2	OBSERVED VISIBILITY	900	5
708250000	000000000	000000ttt	000	0	OBSERVED OBSTR TO VISION	900	2
708300000	000000000	000000ttt	000	0	OBSERVED OPAQUE SKY COVER	900	1
708310000	000000000	000000ttt	000	0	OBSERVED TOTAL SKY COVER	900	4
708500000	000000000	000000ttt	000	0	OBSERVED WEATHER	900	6

C. AEV Forecasts

The 4-word IDs, the scaling factor used in packing the data (which should agree with the Variable Constants File described in Chapter 11), the plain language description packed with the data, and units before packing, are given in the table below. The ttt is the tau (projection) and will vary depending on the projections for which the forecasts were made. The "model number" for MOS forecasts is 80, for local forecasts is 81, and for matching observed data is 82 (see Chapter 4). The observations in these files are those used to verify the local forecasts.

MOS-2000 System 4-Word Id	Scaling	Plain Language	Native Units
CCCFFFBDD VLLLLUUUU TRROHHτττ ISG			
202001080 000000000 000000ttt 000	0	AEV MOS DAYTM MAX TMP	deg F
202001081 000000000 000000ttt 000	0	AEV LCL DAYTM MAX TMP	deg F
202011080 000000000 000000ttt 000	0	AEV MOS NIGHT MIN TMP	deg F
202011081 000000000 000000ttt 000	0	AEV LCL NIGHT MIN TMP	deg F
203505080 000000000 000000ttt 000	2	AEV MOS 12-H POP	value
203505081 000000000 000000ttt 000	2	AEV LCL 12-H POP	value
204200080 000000000 000000ttt 000	-1	AEV MOS WIND DIREC	deg
204200081 000000000 000000ttt 000	-1	AEV LCL WIND DIREC	deg
204202081 000000000 000000ttt 000	-1	AEV LCL SIG WIND DIREC	deg
204210081 000000000 000000ttt 000	0	AEV LCL WIND SPEED	kt
204211080 000000000 000000ttt 000	0	AEV MOS WIND SPEED	kt
204222081 000000000 000000ttt 000	0	AEV LCL SIG WIND SPEED	coded
208000080 000000000 000000ttt 000	0	AEV MOS CEILING HGT	coded
208002081 000000000 000000ttt 000	0	AEV LCL CEILING HGT	100's ft
208100080 000000000 000000ttt 000	0	AEV MOS VISIBILITY	coded
208102081 000000000 000000ttt 000	2	AEV LCL VISIBILITY	mi
208300080 000000000 000000ttt 000	0	AEV MOS OPQ SKY COVER	coded
208310081 000000000 000000ttt 000	0	AEV LCL TOT SKY COVER	coded
208402080 000000000 000000ttt 000	0	AEV MOS 12-H SNOW AMT	coded
208403081 000000000 000000ttt 000	0	AEV LCL 12-H SNOW AMT	in
208501080 000000000 000000ttt 000	0	AEV MOS PRECIP TYPE	coded
208501081 000000000 000000ttt 000	0	AEV LCL PRECIP TYPE	coded
400006000 000000000 000000000 000	4	LATITUDE	deg
400007000 000000000 000000000 000	4	LONGITUDE	deg
702001082 000000000 000000ttt 000	0	AEV OBS DAYTM MAX TMP	deg F
702006082 000000000 000000ttt 000	0	AEV OBS CAL DAY MAX	deg F
702011082 000000000 000000ttt 000	0	AEV OBS NIGHT MIN TMP	deg F
702016082 000000000 000000ttt 000	0	AEV OBS CAL DAY MIN	deg F
703220082 000000000 000000ttt 000	3	AEV OBS 12-H P AMT	in
704200082 000000000 000000ttt 000	-1	AEV OBS WIND DIREC	deg
704201082 000000000 000000ttt 000	-1	AEV OBS WDIR (MAX SPD)	coded
704210082 000000000 000000ttt 000	0	AEV OBS WIND SPEED	kt
704221082 000000000 000000ttt 000	0	AEV OBS MAX WSPD (7HR)	kt
708000082 000000000 000000ttt 000	0	AEV OBS CEILING HGT	100's ft
708100082 000000000 000000ttt 000	2	AEV OBS VISIBILITY	mi
708310082 000000000 000000ttt 000	0	AEV OBS TOT SKY COVER	coded
708402082 000000000 000000ttt 000	2	AEV OBS 12-H SNOWFALL	in
708502082 000000000 000000ttt 000	0	AEV OBS PCP TYPE	coded
708503082 000000000 000000ttt 000	0	AEV OBS PCP TYPE (3HR)	coded

REFERENCES

Carter, G. M., J. P. Dallavalle, and H. R. Glahn, 1989: Statistical forecasts based on the National Meteorological center's numerical weather prediction system. Wea. Forecasting, 4, 401-412.

Dey, Clifford H., 1996: GRIB, Edition 1, the WMO format for the storage of weather product information and the exchange of weather product messages in gridded binary form. National Meteorological Center Office Note 388 (Revised), National Weather Service, NOAA, U.S. Department of Commerce, 75 pp.

Glahn, H. R., 1993: An analysis of some features of the GRIB code. TDL Office Note 92-11, National Weather Service, NOAA, U.S. Department of Commerce, 42 pp.

_____, 1994: An improved algorithm for determining the groups in GRIB second-order packing. TDL Office Note 94-1, National Weather Service, NOAA, U.S. Department of Commerce, 25 pp.

_____, 1995: GRIB product sizes for AWIPS. TDL Office Note 95-3, National Weather Service, NOAA, U.S. Department of Commerce, 29 pp.

_____, 1991: On MOS and perfect prog for interpretive guidance. TDL Office Note 91-3, National Weather Service, NOAA, Department of Commerce, 23 pp.

_____, 1992: On the packing of gridpoint data for efficient transmission. TDL Office Note 92-11, National Weather Service, NOAA, U.S. Department of Commerce, 32 pp.

_____, 1998: Packing of individual radar products. TDL Office Note 98-3, National Weather Service, NOAA, U.S. Department of Commerce, 13 pp.

_____, 1997: Satellite, gridpoint, and vector data packing. TDL Office Note 97-1, National Weather Service, NOAA, U.S. Department of Commerce, 15 pp.

Stackpole, J. D., 1993: GRIB, Edition 1, the WMO format for the storage of weather product information and the exchange of weather product messages in gridded binary form. National Meteorological Center Office Note 388, National Weather Service, NOAA, U.S. Department of Commerce, 75 pp.

Weather Bureau, 1968: Operational forecasts with the Sub-synoptic Advection Model (SAM). Technical Procedures Bulletin No. 16, Environmental Sciences Services Administration, U.S. Department of Commerce, 19 pp.

WMO, 1988: Manual on Codes, Vol. 1, Part B--Binary Codes. WMO No. 306, World Meteorological Organization, Geneva.

TDL ON 00-1

January 1, 2000

APPENDIX

THE TDL MOS DEVELOPMENT SYSTEM
IBM 360/195 VERSION

by

Harry 'R. Glahn

I. INTRODUCTION

Model Output Statistics (MOS) is a technique in which a predicatand is related statistically to predictors which are themselves forecasts (or output) from numerical models. This technique was conceived and planning began for its use in connection with the development of the Subsynoptic Advection Model (SAM) (Glahn and Lowry, 1972) in 1965. Since that time, and up until 1972, two sets of MOS processing programs have been used, one in connection with SAM and PE (the NMC 6-level Primitive Equation Model, Shuman and Hovermale, 1968) data and the other with PE and Trajectory (the TDL Trajectory model, Reap, 1972) data--the so-called PEATMOS system.

As other models were developed, it became clear that we needed a more general system which would accept data, and allow the merging of data, from several models. It would have to be flexible enough so that output from new models, as they are developed, could be accommodated. Planning for this new system started in March 1972. It went into use in 1973 and was described by Glahn (1973). The so-called PEATMOS data collection ended September 30, 1972 and the new MOS collection started on October 1, 1972. All MOS-related programs up to that time were for the CDC 6600.

In late 1973, NOAA began phasing out its CDC 6600 computers and installing its IBM 360/195 system. This made necessary the conversion of all programs and data tapes. These converted programs and tapes are now ready for use and are described in this Note. Conversion was a joint effort of many people in TDL and several have actively contributed to the programming. In this latter group I want especially to mention Frank Globokar, George Hollenbaugh, Al Forst, Don Foster, and Fred Marshall.

This Office Note is updated and expanded as needed; revisions and additions are issued with the pages dated. Gary Carter has assumed joint responsibility for its upkeep. Significant contributions have been made by several persons other than those mentioned above, including John Jensenius and Paul Dallavalle. MOS related development programs are maintained in a program library and user documentation is provided in TDL Office Note 75-2, edited by Glahn, et al. (1975).