Preface

This document specifies format standards to be used to facilitate data exchange for aircraft missions managed by the Earth System Science Division at NASA Ames Research Center. It is intended as a reference document for creating experimental datasets. The standards should be adhered to in all exchange files being contributed to the project archive, including instrumental measurements, theoretical calculations and operational data. It is important that the person responsible for actually generating a given dataset refer to this document when determining the format for that dataset. It is the responsibility of the principal investigator or team leader to make sure that the appropriate people have access to this document and that their data conforms to the format standards.

The specifications described in this document grew out of an effort beginning with the 1987 Stratosphere Troposphere Exchange Project (STEP) to put the experimental aircraft data on a medium and in a format that would be accessible to all experiment participants during the field experiment. Flight planning could then take into account the data from a previous flight, increasing the likelihood of meeting the overall goals and objectives of a given campaign. The standards developed for STEP were also used in the 1987 Airborne Antarctic Ozone Experiment (AAOE) and the 1989 Airborne Arctic Stratospheric Expedition (AASE). The basic premise in specifying format standards was to create self-descriptive datasets using a prescribed header structure to contain information about the data in a given file.

The STEP experiment used only the ER-2 aircraft and consisted primarily of *in situ*, time series data. It was this single dimensional data for which the original format specification was written. The use of remote sensing instruments on both the DC-8 and ER-2, and the generation of model output files creates multi-dimensional data which do not fit well into that original format. To better account for the variety of data, we have formalized some of the original concepts and have extended the requirements for header information to more adequately characterize the data. The file header now must include explicit specification of data dimensionality. We have also allowed for a more flexible specification of the data structure by the use of a file format index, described in the text.

In writing this new format specification, a conscious effort was made to retain much of the progress made in past experiments towards creating an environment of free data exchange. We have tried to build a logical extension to the original concepts rather than making a radical departure from them. Those with experience in the previous experiments (STEP, AAOE and AASE) should recognize that, in many cases, the format for their exchange files will remain much the same under these new specifications with only relatively minor, but important, changes to the header entries.

We want to acknowledge the fact that this document is the product of interactions between the authors and the experiment participants; indeed, the idea of data exchange standards was originally driven by a consensus of the participants, not by the "data managers". We appreciate the feedback that we have received, both written and oral, and have incorporated many of the suggestions into the final document. We also want to encourage everyone to feel free to contact us if they have any questions, problems or suggestions.

1 Introduction

This document describes a conceptual framework for specifying exchange data formats, and then gives a detailed description of the standard formats (although there are very important distinctions between measured quantities and those resulting from mathematical model calculations, for simplicity, the two will often be loosely termed 'data'). Those considering writing exchange files must review the format options presented in this document and determine the format most suitable for recording their data. If none are deemed suitable, consult with the project data manager to define a new format option. New file formats will be circulated to project participants as addenda to this document.

The primary goal of instituting format standards for data exchange is to promote accessibility and ease of use of a variety of datasets from different instruments, platforms and numerical models. The specific goals of the proposed system are:

- The exchange files must be readable on all computer systems commonly in use. These include PC's (MS DOS), Apple Mac's, DEC VMS and Unix systems.
- The exchange files must be self describing, such that the information needed to read the data is contained in an order dependent file header, and the minimum information required to analyze the particular dataset is contained within the file.
- Maintain as much compatibility as possible with existing formats from previous experiments, while allowing flexibility to handle new datasets and formats.
- Minimize the amount of software required to access diverse datasets by categorizing the datasets and allowing a minimal number of data formats.

The complexity of any system of standards increases with increasing generality. This standard represents a compromise between simplicity and generality. The generality of the proposed system stems from the incorporation of a file format index which, by referencing pre-defined format options, defines the format of both the file header and the data records. Thus, new file formats can be incorporated at a future time without changing those defined in this document. The complexity of this system increases with the number of file formats, so an attempt has been made to minimize the number of format options while at the same time accomodate the existing standard data formats from previous experiments; the file header formats are, however, different from the older standards.

An additional advantage of standardized file formats is that the data files can more easily be checked for format errors. Plans for future field experiments include computer programs to check the format of each data file as part of the procedure for submittal to the data archive.

The system described here assumes that all exchange files are in ASCII, because ASCII coded files are the most universally readable across computer systems from different vendors. It is anticipated that the same standards can be extended to include binary files as well. However, before that can be done, a convention for external data representation must be agreed upon due to the differences in internal representation on different machines. A special naming convention for binary files would also have to be adopted.

Guidelines for choosing file names are given in Section 2. Section 3 describes the basic structure of the data files, the types of variables, and how they regulate the format specifications. Section 4 describes some precautionary measures to ensure readable files. The array and implied loop notation used to specify the formats are defined in Section 5, and a collection of definitions of the variables and terminology is contained in Section 6. The file format specifications are given in Section 7, with a summary of data record formats at the end of the Section. An example of each standard format is given in Section 8. Since any particular format option can accommodate a variety of types of data, the concepts, terminology, and format specifications are first presented in an abstract manner so not to bias or narrow their definition. The examples in Section 8 are included to provide a tangible link between the abstract definitions and actual exchange data files. It will, therefore, be useful to refer to these examples while (or before) reading the rest of this document.

2 File Naming Convention

The main objective of a file naming convention is to convey as much information about the file within the limited number of characters allowed for the name. Because of MS DOS limitations, file names must be limited to a maximum of eight characters, followed by a period, and up to three characters in the file name extension. This is a severe restriction considering the many and varied types of measurements and modeling results that are being exchanged. For this reason, there can be no hard and fast rules for file names, but some guidelines for choosing file names are now offered, and the exceptions will be handled as they arise.

In general, the file name consists of a two character prefix which uniquely identifies the measurement or instrument, followed by a six digit number which specifies the UT year, month, and day on which the data within the file begins. The file name extension is separated from the file name by a period, and consists of three characters which identify the measuring platform, the flight number for the indicated day (for aircraft data files), and a volume number to allow for continuation of large files requiring more than one volume of the exchange medium (diskettes, tapes, etc.).

To eliminate conflicting file names, the two character file name prefixes must be decided upon prior to an experiment. Also, the extensions may have to be modified to accomodate new data collection platforms. It is suggested, however, that no matter what the platform, if the data within the file pertains to a particular date or time period then the file name should indicate the UT date for which the data begins. Alternatively, if the data within the file covers an entire mission, including some representation of the mission name in the file name would be more indicative of the contents of the file. Several suggestions and examples will now be given.

2.1 Aircraft data files

Aircraft data files have the standard file name of a two character prefix followed by a six digit number indicating the UT year, month, and day of takeoff. The three character extension starts with either an E (for ER-2) or a D (for DC-8), followed by an A or B to indicate the first or second flight on the indicated day, and ends with a number 1, 2, 3, etc., to indicate the volume number. Some examples are (the prefix MM stands for some arbitrary measurement):

MM910116.EA1.....1st ER-2 flight on Jan 16, 1991, Vol No. 1

MM910116.DA2.....1st DC-8 flight on Jan 16, 1991, Vol No. 2

2.2 Ozonesonde data files

If the ozonesonde data file contains sounding data for a single flight at a particular launch site then use the two character prefix to denote the launch site, followed by a six digit number indicating the UT date of launch. The first character of the extension might be a B (balloon), followed by two digits indicating the UT hour of launch (rounded to the nearest hour). For example:

LW910116.B17.....17 Z launch at Lerwick on Jan 16, 1991

Alternatively, an exchange file might include all ozonesonde soundings from a particular launch site for the duration of a particular mission. For this case, the two character prefix could denote the instrument, and the remaining six characters could be used to represent the mission name. The extension could denote the launch site and volume number. For example:

OS89AASE.LW1.....Lerwick ozonesonde soundings for 1989 AASE, Vol No. 1

2.3 Radiosonde data files

Radiosonde data files present a problem because there are so many launch sites, and possible launch times, that it is difficult to uniquely describe the location and launch time with 11 characters. For this reason, all radiosonde soundings for a particular time period might be lumped together in one file with a name like:

RS910116.121.....radiosonde soundings for 12 Z Jan 16, 1991, Vol No. 1

3 Concepts and Structures

The reason for writing an exchange file is to convey some measured, calculated, or otherwise derived quantity, which will be called the PRIMARY variable. There may be more than one PRIMARY variable in a given exchange file. In addition, there may be some ancillary information concerning the measurement, calculation, or interpretation of the PRIMARY variables or the data records containing their values. These secondary quantities will be referred to as AUXILIARY variables. Usually the inclusion of AUXILIARY variables is optional, but there are some format options in which they are required because they provide information about the ensuing data records. Both PRIMARY and AUXILIARY variables are considered as dependent variables, and are always recorded with reference to at least one INDEPENDENT variable. INDEPENDENT variables can be time, spatial coordinates, index values, or any other quantity that can be used to uniquely identify a particular PRIMARY variable value. Each INDEPENDENT variable represents a dimension on which the PRIMARY variables are dependent. PRIMARY variables are considered as discrete functions of the INDEPENDENT variables, whereas AUXILIARY variables are associated with just one explicitly recorded INDEPENDENT variable.

The information recorded within exchange files is of two types, either numeric or character string. Character strings may contain any printable ASCII character (ASCII decimal values between 32 and 126 inclusive), whereas numeric values are restricted to characters 0 through 9, the plus sign, the minus sign, the period, and the letter E used in exponential notation. Except for the purpose noted in Section 4, an exchange file must not contain non-printable ASCII characters. The non-printable characters have ASCII decimal values of 0 through 31, and values greater than 126.

Each exchange file has a file header which conveys information about the PRIMARY, AUXILIARY, and INDEPENDENT variables, and the order in which they are recorded in the file. Rather than attempt to pre-define a single file header format which accounts for all existing data formats, as well as any future formats, a File Format Index (FFI) is used to uniquely define the exchange file format. By reference to pre-defined format options, the value of the FFI determines the number of INDEPENDENT variables, whether the values of the INDEPENDENT and dependent variables are numeric or character string, the format of the file header, and the format of the data records.

Included in the file header are descriptions and/or units of measure for the INDEPENDENT, PRIMARY, and AUXILIARY variables. All variables must be defined in the records in which they are expected to appear, and cannot be omitted or have blank spaces substituted for their values. Associated with each PRIMARY and AUXILIARY variable is a 'missing' value to denote missing or erroneous data values. These missing values must be larger than any 'good' data values recorded within the file so that a simple test on the magnitude of a data value will determine if it represents missing or usable data. A scale factor is associated with each numeric PRIMARY and AUXILIARY variable. The scale factors are included to encourage recording of the data as scaled whole numbers, without a decimal point or exponential notation, and thus reduce the size of the file. There are no scale factors or missing values for the INDEPENDENT variables.

The order in which the PRIMARY and AUXILIARY variables are defined in the file header is the same order in which they are recorded in the data records. The order in which the INDEPENDENT variables are defined in the file header determines the dependence of the PRIMARY variables on the INDEPENDENT variables and, therefore, the manner in which the PRIMARY variables are recorded. The recorded dependence of the PRIMARY variables on the INDEPENDENT variables is such that, from the point of view of writing the data records, the most rapidly varying dimension is listed first in the file header, and the most slowly varying dimension is listed last.

If the number of values in the most slowly varying dimension is not pre-determined (as with the time dimension in many cases) then it is termed the *unbounded* dimension. Of necessity, only one dimension, or INDEPENDENT variable, can be *unbounded*, while the others, if any, must be *bounded*. The number of values in the *bounded* dimensions are defined in either the file header or the data records. Values of the *unbounded* INDEPENDENT variable are explicitly recorded at pre-determined locations within the data records and are termed INDEPENDENT VARIABLE MARKS. The AUXILIARY variables, if any, are specified immediately after the INDEPENDENT VARIABLE MARKS, either within the same record or the subsequent records. The *unbounded* INDEPENDENT variable must be a monotonic quantity.

As an illustration, consider airborne lidar measurements of ozone, recorded as a time sequence of vertical profiles of ozone. For this example, ozone number density is the PRIMARY variable, altitude above Earth's surface is the *bounded* INDEPENDENT variable, and a monotonic measure of time is the *unbounded* INDEPENDENT variable. The same records which contain time may also contain AUXILIARY variables. Since ozone values at all recorded altitudes are read for each time mark, the dependence on altitude is considered the more rapidly varying dependence, so altitude is the first INDEPENDENT variable defined in the file header; time is defined second in the file header. An examination of the standard formats in Section 7 reveals that there are several file formats which could accommodate the data in this example, the selection of the most appropriate format depends on the nature of the altitude measurements. If the values of altitude are constant then FFI 2010 could be used, with the constant altitudes defined in the file header. In this instance, AUXILIARY variables are optional, but one may wish to include additional information, say, aircraft longitude and latitude, in the AUXILIARY variable list. If the altitude values are variable, but the interval between the altitudes is constant, then FFI 2310 is more appropriate. In this instance the number of altitudes, base altitude value, and altitude increment are supplied in the AUXILIARY variable list. If the altitude values and the intervals between altitudes are variable then FFI 2110 is the most appropriate option, with the number of altitudes given in the AUXILIARY variable list, and the altitude values read from the records containing the ozone values. In these last two instances (FFI 2310, 2110), the indicated AUXILIARY variables are required, in the sense that they provide necessary information for reading subsequent data records, but one still has the option to include additional AUXILIARY variables.

The file header also contains information on the originators of the exchange file and their affiliations, the source of the PRIMARY variables, the mission which the data supports, and (by popular demand) the number of lines in the file header. The originators will often be the principal investigators for a particular instrument or model simulation, and the instrument and platform, or model, will be the source. At the beginning of each mission, a mission name will be decided upon and used in all exchange files.

Also included, are the date for which the data applies, the date the data was reduced or revised (not necessarily the date the file was written, although the two may be the same), the volume number for the exchange file, and the total number of volumes required to record the complete dataset. For large datasets requiring more than one volume of the medium on which they are written (diskette, tape, etc.), the data is continued in a new file, on a new volume, and after a file header with an incremented volume counter (see IVOL, NVOL in Section 6).

There are allowances for three types of comments in the file header. Two of these comment types have reserved locations within the file header, and are associated with counters defining the number of lines occupied by each type of comment. The first type is for more complete descriptions of the variables, instrument, or other comments that apply in general to all of a particular kind of dataset; these are called normal comments. The second type, called special comments, are reserved to note special problems or circumstances concerning the data within a specific exchange file. If the exchange file is a revised dataset then it is recommended that the special comments describe how it differs from the previous version of the dataset. The third type of comments are merely annotations which may follow numeric values; these comments must be contained on the same line as, and separated by at least one space from the last numeric value expected in the record.

The data records immediately follow the file header records and continue to the end of the file. One or more spaces (ASCII decimal value 32) delimit successive numeric values within a line in both the file header and the data records.

4 Implementation Considerations

Even though an ASCII file is the most universally readable type of file, there are differences in the way different operating systems define the end of a line for their I/O device drivers and compilers. Therefore, some consideration must be given to the way in which files are transferred between machines with different operating systems.

MS DOS uses the ASCII characters for carriage return `<CR>` and line feed `<LF>` to terminate each line, Macintosh uses just `<CR>`, Unix uses just `<LF>`, whereas VAX/VMS has control words at the beginning of each line which give the number of characters in the line. It is, therefore, impracticable to write an ASCII file which will appear as a native to every operating system. If inter-system file transfers are performed using some of the 'standard' file transfer software (Kermit, FTP, DECnet-DOS, etc.) then the conversion to the appropriate end-of-line designator is done automatically during the file transfer, assuming it is not a bit for bit (binary) transfer. But if the file is written to some storage medium (diskette, tape, compact disc, etc.) under one operating system, and read from the medium by another operating system, then it may be necessary to rewrite or edit the file to a form with the appropriate end-of-line designator. Analogous to the end-of-line designator, the end-of-file designator differs with different operating systems, but is appropriately converted using standard file transfer software.

There is currently no convenient solution to the above stated dilemma. It is mentioned mainly to alert originators and users of exchange files to the potential problems with transferring ASCII files between different operating systems. In the past, the MS DOS designators have been the convention for transferring files via diskette and compact disc, but this may change as technology and industry standards change. In any case, prior to each mission, the mode of file transfer, and the acceptable end-of-line and end-of-file designators, will be decided upon and communicated to the project participants.

Except for the purpose of preparing a file for use on a different operating system, there must not be any extraneous non-printable ASCII characters within an exchange file. The non-printable characters have ASCII decimal values of 0 through 31, and values greater than 126. For similar reasons, exchange files must not be Fortran output files with Fortran carriage control characters embedded within the file.

Operating systems and programming languages impose limitations on record length, magnitudes of integer and real numbers, and precision of real numbers. To comply with limitations in the most commonly used environments, the maximum record length in exchange files is 32766 characters. It is suggested that values of real numbers be limited to seven significant digits within the range 1.0E–38 to 1.0E+38. There is no need to place a limit on the magnitude of integer values, since the limit on record length and the context within which integers are used, automatically impose limits which are compatible with most environments.

There is, however, a fine point of distinction between integer values and real values recorded as scaled whole numbers. The values of the primary and auxiliary variables are often defined to be real numbers, even though they may be recorded as scaled whole numbers without decimal points (i.e., integers). But even though they may look like integers, the limitations on their range of values and number of significant digits is the same as for real numbers. VAX Fortran, for example, distinguishes between 16 bit integers and 32 bit integers. A number such as 99999 is too large to be read as a 16 bit integer, although it could be read as a real number or a 32 bit integer. Using a 32 bit integer to record a scaled whole number with more than seven significant digits (eg. 123456789) can be accomplished, but may force those reading the file to use some special contrivance to pick up all of the significant digits; it is, therefore, discouraged. Similar comments apply to real valued independent variables which are recorded without decimal points.

For numeric data, there should be an adequate number of digits to resolve the anticipated precision, but in the interest of minimizing the file size, the number of digits should not be larger than necessary. Also, unnecessary records of missing values should not be used to pad the beginning or end of the data section of an exchange file. If, for example, the data from an airborne instrument begins 10 minutes after takeoff, and terminates 10 minutes before landing, it is unnecessary to include 10 minutes of missing values before the data begins and after it terminates.

5 Notation

The array and implied loop notation, used to generalize the exchange file format definitions given in Section 7, will now be explained. The notation is merely a convenient means of specifying the file formats, and not intended to indicate useful or desirable array structures in computer programs.

Quantities enclosed in square brackets `[ ]` are read with one 'read' statement and, therefore, the quantities occupy one record which may exceed one line. One or more quantities appearing in a line, and not enclosed in square brackets, are read as one record and constitute one line in the exchange file. Similar comments apply to writing the records, but the descriptions which follow in this Section are from the perspective of reading the records.

The loop indicies act merely as counters, but several indicies are consistently used for special purposes. The index `m` is always used to count independent variable marks and, unless otherwise stated, the implied loop over `m` is unbounded. The index `s` is the counter for the independent variables (dimensions), `n` for the primary variables, and `a` for the auxiliary variables. The usage of other indicies (i, j, k, etc.) is less consistent, but usually they are counters for the actual values of a particular independent variable.

Consider the array `X`, which contains (not necessarily constant) values of the independent variables on which the primary variables are dependent. To reference a specific array element, we write `X(2,1)`. To reference a general array element, we write `X(i,s)`, where `i` and `s` can assume any allowable values. To indicate the allowable range of values for `i` and `s`, we write `X(i,s), i=1,NX(s), s=1,NIV`; which states `s` may take on integer values of 1 to `NIV`, and `i` may assume integer values of 1 to `NX(s)`, the value of `NX` depending on the value of `s`. `NIV` is the number of independent variables, and `NX(s)` is the number of values for the s-th independent variable.

For reading data records, the implied loop notation has a slightly different meaning, because then it implies that during the read operation the loop index *will* sequentially take on the values dictated by the loop limits. If the terminal value of a loop is smaller than the initial value, the implication is that the loop is not executed. The expression

```
[ X(i,s), i=1,NX(s) ]  s=1,NIV
```

is to be interpreted to mean that there are `NIV` records which will be read with `s` starting at a value of 1, incrementing by one for each record, and ending with a value of `NIV` for the last record. The notation within the square brackets indicates that, for each record, `NX(s)` values of `X` are read, with the value of `NX` depending on the current value of `s`.

As a further illustration, consider the example of reading primary variables which depend on two independent variables (`NIV=2`). The first independent variable (`X(i,1), i=1,NX(1)`) is bounded with its constant values defined in the file header. The second independent variable (`X(m,2)`) is unbounded with its values defined by the independent variable marks. Also assume that auxiliary variables are recorded in the same records containing the independent variable marks. Let the general expression for the data format be:

```
[ X(m,2)  ( A(m,a), a=1,NAUXV ) ]
[ V(i,m,n), i=1,NX(1) ]  n=1,NV
```

In the above expressions `X(m,2)` represents the m-th independent variable mark for the unbounded independent variable (the implied loop over `m` is unbounded); `A(m,a)` is the value of the a-th auxiliary variable at the m-th independent variable mark; and `V(i,m,n)` is the value of the n-th primary variable at the m-th independent variable mark and i-th bounded independent variable value. To be correct, the expression `V(i,m,n)` should really be written as `V(X(i,1),X(m,2),n)`. But to simplify the notation it will be expressed as `V(i,m,n)`.

To be more specific, assume there are three auxiliary variables (`NAUXV=3`), two primary variables (`NV=2`), and four values for the bounded independent variable (`NX(1)=4`). Given these values for the loop limits, the general expressions for the data format imply the following record structure (the intra-record spacing between values is merely for clarity):

```
[ X(m,2)    A(m,1)    A(m,2)    A(m,3) ]
[  V(1,m,1)    V(2,m,1)    V(3,m,1)    V(4,m,1) ]
[  V(1,m,2)    V(2,m,2)    V(3,m,2)    V(4,m,2) ]
[ X(m+1,2)  A(m+1,1)  A(m+1,2)  A(m+1,3) ]
[  V(1,m+1,1)  V(2,m+1,1)  V(3,m+1,1)  V(4,m+1,1) ]
[  V(1,m+1,2)  V(2,m+1,2)  V(3,m+1,2)  V(4,m+1,2) ]
[ X(m+2,2)  A(m+2,1)  A(m+2,2)  A(m+2,3) ]
etc., etc.
```

If, for the sake of illustration, `NAUXV=0` then according to the loop limits in the general expressions for the data format given above, the terminal value of the loop would be smaller than the initial value. The implication would then be that no auxiliary variables were present in the file and, therefore, none would be read from the file. The data records would remain the same as those in the above example, with the exception that there would be no auxiliary variables in the records containing the independent variable marks.

6 Definitions

A(m,a) value of the a-th auxiliary variable at the m-th independent variable mark (a=1,NAUXV). If A(m,a) is real, the use of scaled whole numbers is encouraged.

AMISS(a) a quantity indicating missing or erroneous data for the a-th auxiliary variable. The value of AMISS(a) must be larger than any 'good' value of A(m,a) recorded in the file.

ANAME(a) a character string specifying the name and/or description of the a-th auxiliary variable, on one line and not exceeding 132 characters. Include units of measure the data will have after multiplying by the a-th scale factor, ASCAL(a). The order in which the auxiliary variable names are listed in the file header is the same order in which the auxiliary variables are read from the data records, and the same order in which the auxiliary variable scale factors and missing values are read from the file header records.

ASCAL(a) scale factor (real) by which one multiplies recorded values of the a-th auxiliary variable to convert them to the units specified in ANAME(a).

character string is a string of at most 132 printable ASCII characters occupying one line of an exchange file. The printable ASCII characters have ASCII decimal values between 32 and 126 inclusive.

DATE UT date at which the data within the exchange file begins. For aircraft data files DATE is the UT date of takeoff. DATE is in the form YYYY MM DD (year, month, day) with each integer value separated by at least one space. For example: `1989  1 16` or `1989 01 16` for January 16, 1989.

DX(s) interval (real) between values of independent variable X(i,s), i=1,NX(s); in the same units as specified in XNAME(s). DX(s) is zero for a nonuniform interval. DX(s) is nonzero for a constant interval. If DX(s) is nonzero then it is required that NX(s) = (X(NX(s),s)–X(1,s)) / DX(s) + 1.

FFI file format index (integer). The FFI uniquely defines the file header and data formats. It is the second value recorded on the first line of an exchange file. The first (left-most) digit in the FFI gives the number of independent variables listed in the file header, the second digit gives the number of required (in the sense that they are necessary for reading the subsequent data records) auxiliary variables. The remaining digits are used to loosely associate file formats with similar characteristics.

independent variable mark an explicitly recorded value of an independent variable. Always required to specify values of an unbounded independent variable.

integer a whole number written without a decimal point. Leading zeros are insignificant.

IVOL volume number (integer) of the total number of volumes required to store a complete dataset, assuming only one file per volume. To be used in conjunction with NVOL to allow data exchange of large datasets requiring more than one volume of the data exchange medium (floppy disk, tape, etc.).

LENA(a) integer number of characters used to record auxiliary variable A(m,a) when A(m,a) is represented as a character string. The value of LENA(a) must be less than 133.

LENX(s) integer number of characters used to record independent variable X(i,s) when X(i,s) is represented as a character string. The value of LENX(s) must be less than 133.

line refers to a string of printable ASCII characters within an exchange file, terminated by the appropriate end-of-line (or new line) designator for the operating system under which the file was written. A properly defined line appears as a horizontal string of printable characters when displayed on some output device (terminal screen, printer, etc.), assuming the output device can accomodate the number of characters within the line. The maximum number of characters per line is 132.

MNAME a character string specifying the mission which the data is supporting, on one line and not exceeding 132 characters. The appropriate value for MNAME will be decided upon prior to the start of the mission.

NAUXC number of auxiliary variables (integer) whose values are recorded as character strings. If NAUXC = 0 then no auxiliary variables are recorded as character strings.

NAUXV number of auxiliary variables (integer). If NAUXV = 0 then no auxiliary variables are recorded and no missing values, scale factors, or names for the auxiliary variables are present in the file header.

NCOM(k) a character string containing the k-th normal comment line (k=1,NNCOML).

NIV number of independent variables (integer) on which the primary variables are dependent.

NLHEAD number of lines (integer) composing the file header. NLHEAD is the first recorded value on the first line of an exchange file.

NNCOML number of normal comment lines (integer) within the file header, including blank lines and data column headers, etc. Normal comments are those which apply to all of a particular kind of dataset and can be used to more completely describe the contents of the file. If NNCOML=0 then there are no normal comment lines.

NSCOML number of special comment lines (integer) within the file header. Special comments are reserved to note special problems or circumstances concerning the data within a specific exchange file so they may easily be found and flagged by those reading the file. If NSCOML=0 then there are no special comment lines.

NV number of primary variables in the exchange file (integer).

NVOL total number of volumes (integer) required to store the complete dataset, assuming one file per volume.

NVPM(s) integer number of independent variable values between independent variable marks, for the s-th independent variable. NVPM(s) = (X(m+1,s)–X(m,s)) / DX(s).

NX(s) number of values (integer) for the s-th independent variable. If NX(s) is defined in the file header then it represents the constant number of values for the s-th independent variable. Otherwise, NX(s) is defined in the data records or, in the case of an unbounded independent variable, NX(NIV) is never specified but the values of X(m,NIV) are read from the data records (independent variable marks).

NXDEF(s) number of values (integer) of the s-th independent variable which are explicitly defined in the file header. If NXDEF(s)=NX(s) then all values of X(i,s), i=1,NX(s) are recorded in the file header. If NXDEF(s)=1 then only the first value, X(1,s), is recorded in the file header and the remaining values of X(i,s) are calculated as
X(i,s) = X(1,s) + (i–1) * DX(s) for i=2,NX(s).

ONAME a character string specifying the name(s) of the originator(s) of the exchange file, last name first. On one line and not exceeding 132 characters.

ORG character string specifying the organization or affiliation of the originator of the exchange file. Can include address, phone number, etc. On one line and not exceeding 132 characters.

RDATE date of data reduction or revision, in the same form as DATE.

real a real valued number that may include a decimal point or be written in exponential notation. It is preferred that the values of real numbers be limited to seven significant digits within the range 1.0E–38 to 1.0E+38.

record a logical record to be read by one 'read' statement. The maximum record length is 32766 characters with a maximum of 132 characters per line. The first character of a record is also the first character of a line.

SCOM(k) a character string containing the k-th special comment line (k=1,NSCOML).

SNAME a character string specifying the source of the measurements or model results which compose the primary variables, on one line and not exceeding 132 characters. Can include instrument name, measurement platform, etc.

V(X,n) value of n-th primary variable at specified values of independent variables X (n=1,NV). If V is real then the use of scaled whole numbers, without decimal points, is encouraged.

VMISS(n) a quantity indicating missing or erroneous data values for the n-th primary variable. VMISS(n) must be larger than any 'good' data value, of the n-th primary variable, recorded in the file.

VNAME(n) a character string giving the name and/or description of the n-th primary variable, on one line and not exceeding 132 characters. Include units of measure the data will have after multiplying by the n-th scale factor, VSCAL(n). The order in which the primary variable names are listed in the file header is the same order in which the primary variables are read from the data records, and the same order in which scale factors and missing values for the primary variables are read from the file header records.

VSCAL(n) scale factor (real) by which one multiplies recorded values of the n-th primary variable to convert them to the units specified in VNAME(n).

X(i,s) i-th value of the s-th independent variable (X(i,s), i=1,NX(s), s=1,NIV).

XNAME(s) a character string giving the name and/or description of the s-th independent variable, on one line and not exceeding 132 characters. Include units of measure and order the independent variable names such that, when reading primary variables from the data records, the most rapidly varying independent variable is listed first and the most slowly varying independent variable is listed last.

7 ASCII File Format Specifications

The file format specifications, ordered by increasing file format index, FFI, are given in this Section. Refer to the definitions in Section 6 for explanations of the variables. A brief description of the variables and file format is followed by the format of the file header, and the general expression for the data records. Occasionally, lower case characters, preceded by several periods, are used to annotate certain variables.

```
FFI = 1001:
One real, unbounded independent variable (NIV=1).
Primary variables are real.
No auxiliary variables.
Independent and primary variables are recorded in the same record.

NLHEAD   1001
ONAME
ORG
SNAME
MNAME
IVOL   NVOL
DATE   RDATE
DX(1)
XNAME(1)
NV
[ VSCAL(n), n=1,NV ]
[ VMISS(n), n=1,NV ]
[ VNAME(n) ]  n=1,NV
NSCOML
[ SCOM(k) ]  k=1,NSCOML
NNCOML
[ NCOM(k) ]  k=1,NNCOML
[ X(m,1)  ( V(m,n), n=1,NV ) ]
```

FFI = 1010:
One real, unbounded independent variable (NIV=1).
Primary variables are real.
Auxiliary variables are real.
The independent and auxiliary variables are in the same record.
All primary variables for a given independent variable mark are recorded in the same record.

```
NLHEAD   1010
ONAME
ORG
SNAME
MNAME
IVOL   NVOL
DATE   RDATE
DX(1)
XNAME(1)
NV
[ VSCAL(n), n=1,NV ]
[ VMISS(n), n=1,NV ]
[ VNAME(n) ]  n=1,NV
NAUXV
[ ASCAL(a), a=1,NAUXV ]
[ AMISS(a), a=1,NAUXV ]
[ ANAME(a) ]  a=1,NAUXV
NSCOML
[ SCOM(k) ]  k=1,NSCOML
NNCOML
[ NCOM(k) ]  k=1,NNCOML
[ X(m,1)  ( A(m,a), a=1,NAUXV ) ]
[ V(m,n), n=1,NV ]
```

FFI = 1020:
One real, constant increment, unbounded independent variable with implied values between independent variable marks (NIV=1).
Primary variables are real.
Auxiliary variables are real.
The independent and auxiliary variables are in the same record.
A record of primary variable values at implied independent variable values is recorded for each primary variable.

```
NLHEAD   1020
ONAME
ORG
SNAME
MNAME
IVOL   NVOL
DATE   RDATE
DX(1)  ..............DX(1) not equal to zero
NVPM(1)
XNAME(1)
NV
[ VSCAL(n), n=1,NV ]
[ VMISS(n), n=1,NV ]
[ VNAME(n) ]  n=1,NV
NAUXV
[ ASCAL(a), a=1,NAUXV ]
[ AMISS(a), a=1,NAUXV ]
[ ANAME(a) ]  a=1,NAUXV
NSCOML
[ SCOM(k) ]  k=1,NSCOML
NNCOML
[ NCOM(k) ]  k=1,NNCOML
[ X(m,1)  ( A(m,a), a=1,NAUXV ) ]
[ V(i,n), i=1+(m-1)*NVPM(1),m*NVPM(1) ]  n=1,NV
```

```
FFI = 2010:
Two real independent variables, one unbounded and one bounded with constant
values (NIV=2).
Primary variables are real.
Auxiliary variables are real.
Independent variable mark and auxiliary variables are in the same record.
For each primary variable is a record of its values at the bounded
independent variable values.

NLHEAD   2010
ONAME
ORG
SNAME
MNAME
IVOL    NVOL
DATE    RDATE
DX(1)   DX(2)
NX(1)
NXDEF(1)
[ X(i,1), i=1,NXDEF(1) ]
[ XNAME(s) ]  s=1,2
NV
[ VSCAL(n), n=1,NV ]
[ VMISS(n), n=1,NV ]
[ VNAME(n) ]  n=1,NV
NAUXV
[ ASCAL(a), a=1,NAUXV ]
[ AMISS(a), a=1,NAUXV ]
[ ANAME(a) ]  a=1,NAUXV
NSCOML
[ SCOM(k) ]  k=1,NSCOML
NNCOML
[ NCOM(k) ]  k=1,NNCOML
[ X(m,2)  ( A(m,a), a=1,NAUXV ) ]
[ V(i,m,n), i=1,NX(1) ]  n=1,NV
```

```
FFI = 2110:
Two real independent variables, one unbounded and one bounded with its
values recorded in the data records (NIV=2).
Primary variables are real.
The first auxiliary variable is NX(1), other auxiliary variables are real.
The values of X(i,1) are included in the records with the primary variables.
If DX(2) is nonzero then X(m,2) must be recorded at a constant interval
of DX(2).  For this case, if NX(1)=AMISS(1) or NX(1)=0 then the implication
is that the records containing values of the bounded independent variable
and primary variables are omitted, and the next record contains the
succeeding independent variable mark and auxiliary variables.

NLHEAD   2110
ONAME
ORG
SNAME
MNAME
IVOL   NVOL
DATE   RDATE
DX(1)  DX(2)
[ XNAME(s) ]  s=1,2
NV
[ VSCAL(n), n=1,NV ]
[ VMISS(n), n=1,NV ]
[ VNAME(n) ]  n=1,NV
NAUXV ................The first auxiliary variable is NX(1)
[ ASCAL(a), a=1,NAUXV ]
[ AMISS(a), a=1,NAUXV ]
[ ANAME(a) ]  a=1,NAUXV
NSCOML
[ SCOM(k) ]  k=1,NSCOML
NNCOML
[ NCOM(k) ]  k=1,NNCOML
[ X(m,2)  NX(1)  ( A(m,a), a=2,NAUXV ) ]
[ X(i,1)  ( V(i,m,n), n=1,NV ) ]  i=1,NX(1)
```

FFI = 2160:
Two independent variables; the unbounded independent variable is a character string of length LENX(2); the bounded independent variable is real with its values recorded in the data records (NIV=2).
Primary variables are real.
The independent variable mark is in a separate record from the auxiliary variables.
The first auxiliary variable is NX(1).
NAUXC is the number of auxiliary variables recorded as character strings, which follow the real-valued auxiliary variables, and have lengths LENA(a), a=NAUXV-NAUXC+1,NAUXV. Therefore, AMISS(a), a=NAUXV-NAUXC+1,NAUXV are also character strings of length LENA(a).
The values of X(i,1) are included in the records with the primary variables.

```
NLHEAD   2160
ONAME
ORG
SNAME
MNAME
IVOL   NVOL
DATE   RDATE
DX(1)
LENX(2)
[ XNAME(s) ]  s=1,2
NV
[ VSCAL(n), n=1,NV ]
[ VMISS(n), n=1,NV ]
[ VNAME(n) ]  n=1,NV
NAUXV  .....................first auxiliary variable is NX(1)
NAUXC
[ ASCAL(a), a=1,NAUXV-NAUXC ]
[ AMISS(a), a=1,NAUXV-NAUXC ]
[  LENA(a), a=NAUXV-NAUXC+1,NAUXV ]
[ AMISS(a) ]  a=NAUXV-NAUXC+1,NAUXV
[ ANAME(a) ]  a=1,NAUXV
NSCOML
[ SCOM(k) ]  k=1,NSCOML
NNCOML
[ NCOM(k) ]  k=1,NNCOML
[ X(m,2) ]
[ NX(1)  ( A(m,a), a=2,NAUXV-NAUXC ) ]
[ A(m,a) ]  a=NAUXV-NAUXC+1,NAUXV
[ X(i,1)  ( V(i,m,n), n=1,NV ) ]  i=1,NX(1)
```

```
FFI = 2310:
Two real independent variables, one unbounded and one bounded with its
number of constant increment values, base value, and increment defined in
the auxiliary variable list (NIV=2).
Primary variables are real.
The first three auxiliary variables are NX(1), X(1,1), and DX(1); the other
auxiliary variables are real.
If DX(2) is nonzero then X(m,2) must be recorded at a constant interval
of DX(2).  For this case, if NX(1)=AMISS(1) or NX(1)=0 then the implication
is that the records containing values of the primary variables are omitted,
and the next record contains the succeeding independent variable mark and
auxiliary variables.
For each primary variable is a record of its values at the bounded
independent variable values.
The bounded independent variable values are X(i,1) = X(1,1) + (i-1) * DX(1)
for i=1,NX(1).

NLHEAD   2310
ONAME
ORG
SNAME
MNAME
IVOL   NVOL
DATE   RDATE
DX(2)
[ XNAME(s) ]  s=1,2
NV
[ VSCAL(n), n=1,NV ]
[ VMISS(n), n=1,NV ]
[ VNAME(n) ]  n=1,NV
NAUXV  ...........first three auxiliary variables are NX(1), X(1,1), DX(1)
[ ASCAL(a), a=1,NAUXV ]
[ AMISS(a), a=1,NAUXV ]
[ ANAME(a) ]  a=1,NAUXV
NSCOML
[ SCOM(k) ]  k=1,NSCOML
NNCOML
[ NCOM(k) ]  k=1,NNCOML
[ X(m,2)  NX(1)  X(1,1)  DX(1)  ( A(m,a), a=4,NAUXV ) ]
[ V(i,m,n), i=1,NX(1) ]  n=1,NV
```

FFI = 3010:
Three real independent variables, one unbounded and two bounded with constant values defined in the file header (NIV=3).
Primary variables are real.
Auxiliary variables are real.
The independent variable marks and auxiliary variables are in the same record.
For each primary variable and value of the second independent variable, is a record of primary variable values at values of the first independent variable.

```
NLHEAD   3010
ONAME
ORG
SNAME
MNAME
IVOL      NVOL
DATE      RDATE
DX(1)     DX(2)    DX(3)
NX(1)     NX(2)
NXDEF(1)  NXDEF(2)
[ X(i,1), i=1,NXDEF(1) ]
[ X(j,2), j=1,NXDEF(2) ]
[ XNAME(s) ]  s=1,3
NV
[ VSCAL(n), n=1,NV ]
[ VMISS(n), n=1,NV ]
[ VNAME(n) ]  n=1,NV
NAUXV
[ ASCAL(a), a=1,NAUXV ]
[ AMISS(a), a=1,NAUXV ]
[ ANAME(a) ]  a=1,NAUXV
NSCOML
[ SCOM(k) ]  k=1,NSCOML
NNCOML
[ NCOM(k) ]  k=1,NNCOML
[ X(m,3)  ( A(m,a), a=1,NAUXV ) ]
[ V(i,j,m,n), i=1,NX(1) ]  j=1,NX(2)  n=1,NV
```

FFI = 4010:
Four real independent variables, one unbounded and three bounded with constant values defined in the file header (NIV=4).
Primary variables are real.
Auxiliary variables are real.
The independent variable marks and auxiliary variables are in the same record.
For each primary variable and value of the third and second independent variables, is a record of primary variable values, at values of the first independent variable.

```
NLHEAD   4010
ONAME
ORG
SNAME
MNAME
IVOL      NVOL
DATE      RDATE
DX(1)     DX(2)     DX(3)    DX(4)
NX(1)     NX(2)     NX(3)
NXDEF(1)  NXDEF(2)  NXDEF(3)
[ X(i,1), i=1,NXDEF(1) ]
[ X(j,2), j=1,NXDEF(2) ]
[ X(k,3), k=1,NXDEF(3) ]
[ XNAME(s) ]  s=1,4
NV
[ VSCAL(n), n=1,NV ]
[ VMISS(n), n=1,NV ]
[ VNAME(n) ]  n=1,NV
NAUXV
[ ASCAL(a), a=1,NAUXV ]
[ AMISS(a), a=1,NAUXV ]
[ ANAME(a) ]  a=1,NAUXV
NSCOML
[ SCOM(k) ]  k=1,NSCOML
NNCOML
[ NCOM(k) ]  k=1,NNCOML
[ X(m,4)  ( A(m,a), a=1,NAUXV ) ]
[ V(i,j,k,m,n), i=1,NX(1) ]  j=1,NX(2)  k=1,NX(3)  n=1,NV
```

7.1 Summary of data record formats

```
FFI=1001:  [ X(m,1)  ( V(m,n), n=1,NV ) ]


FFI=1010:  [ X(m,1)  ( A(m,a), a=1,NAUXV ) ]
           [ V(m,n), n=1,NV ]


FFI=1020:  [ X(m,1)  ( A(m,a), a=1,NAUXV ) ]
           [ V(i,n), i=1+(m-1)*NVPM(1),m*NVPM(1) ]  n=1,NV


FFI=2010:  [ X(m,2)  ( A(m,a), a=1,NAUXV ) ]
           [ V(i,m,n), i=1,NX(1) ]  n=1,NV


FFI=2110:  [ X(m,2)  NX(1)  ( A(m,a), a=2,NAUXV ) ]
           [ X(i,1)  ( V(i,m,n), n=1,NV ) ]  i=1,NX(1)


FFI=2160:  [ X(m,2) ] ............................ X(m,2)=character string
           [ NX(1)  ( A(m,a), a=2,NAUXV-NAUXC ) ]
           [ A(m,a) ]  a=NAUXV-NAUXC+1,NAUXV
           [ X(i,1)  ( V(i,m,n), n=1,NV ) ]  i=1,NX(1)


FFI=2310:  [ X(m,2)  NX(1)  X(1,1)  DX(1)  ( A(m,a), a=4,NAUXV ) ]
           [ V(i,m,n), i=1,NX(1) ]  n=1,NV


FFI=3010:  [ X(m,3)  ( A(m,a), a=1,NAUXV ) ]
           [ V(i,j,m,n), i=1,NX(1) ]  j=1,NX(2)  n=1,NV


FFI=4010:  [ X(m,4)  ( A(m,a), a=1,NAUXV ) ]
           [ V(i,j,k,m,n), i=1,NX(1) ]  j=1,NX(2)  k=1,NX(3)  n=1,NV
```

8 Examples

Examples of the file formats, ordered by increasing file format index, are given in this Section. These are fictitious examples and any similarity with existing exchange files is purely a manifestation of the author's lack of imagination. For each example is a file header, followed by a sample of the data records. Numeric constants in the file headers have been annotated with comments enclosed by `{ }`. The annotations are included here as references and need not appear in exchange files.

```
22  1001              {NLHEAD  FFI}
MERTZ, FRED
PACIFIC UNIV.
WIND DATA FROM ER-2 METEOROLOGICAL MEASUREMENT SYSTEM (MMS)
1991 TAHITI OZONE PROJECT
 1  3                     {IVOL  NVOL}
1991  1 16   1991  1 16   {DATE  RDATE}
0                         {DX(1)=0 for nonuniform time intervals}
ELAPSED TIME IN UT SECONDS FROM 0 HOURS ON THE DAY GIVEN IN DATE ON LINE 7
3                         {NV=number of primary variables}
0.1  0.1   0.1            {primary variable scale factors}
999  9999  999            {primary variable missing values}
HORIZONTAL WIND SPEED (m/s)
HORIZONTAL WIND DIRECTION (deg); TRUE DIRECTION FROM WHICH IT BLOWS.
VERTICAL WIND SPEED + up (m/s)
1                         {NSCOML=number of special comment lines}
Pilot experienced CAT between the times 50300-50400.
4                         {NNCOML=number of normal comment lines}
Preliminary wind data
1Hz desampled from 5Hz
OMEGA used for calc = 0.06280  RAD/SEC
  UTs       Spd  Direc Vert Wind
  30446.9  305  2592   22
  30447.9  304  2596   22
  30448.9  305  2601  999
  30449.9  306  2603  999
  30450.9  307  2606   25
  30451.8  307  2607   27
  30452.8  309  2610   29
  30453.8  310  2610   29
  30454.8  312  2621   32
```

```
40  1010               {NLHEAD  FFI}
Mertz, Fred
Pacific University
DC-8 Mark IV Interferometer
1991 TAHITI OZONE PROJECT
1  1                       {IVOL  NVOL}
1991  1 16   1991  2 15    {DATE  RDATE}
0                          {DX(1)=0 for nonuniform time intervals}
UT fractional day number of year given in DATE
8                          {NV=number of primary variables}
1.0E+17 1.0E+14 1.0E+13 1.0E+14 1.0E+14 1.0E+13 1.0E+13 1.0E+18   {VSCAL}
999 999 9999 9999 999 9999 9999 9999                              {VMISS}
O3 vertical column density; molecules / cm**2
NO vertical column density; molecules / cm**2
NO2 vertical column density; molecules / cm**2
HNO3 vertical column density; molecules / cm**2
ClNO3 vertical column density; molecules /cm**2
HCl vertical column density; molecules / cm**2
HF vertical column density; molecules / cm**2
H2O vertical column density; molecules /cm**2
10                         {NAUXV=number of auxiliary variables}
1.0 1.0 1.0 1.0 0.1 0.1 0.1 1.0 1.0 1.0      {ASCAL}
99 99 99 99 9999 99999 9999 999 999 999      {AMISS}
Month number of observation (UT)
Day of month (UT)
Hour of day (UT)
Minutes of hour (UT)
Latitude of observation; decimal degrees
Longitude of observation; decimal degrees
Solar zenith angle, reckoned from DC-8, at time of observation; degrees
Air temperature at time of observation; degrees Celsius
Static pressure at time of observation; millibars
Potential temperature at time of observation; degrees Kelvin
0                          {NSCOML}
5                          {NNCOML}
NOTE 1: This is a single file for the entire mission, which will be
updated after each flight during the mission.  See line 7 of header for
date of last update.
NOTE 2: All these column values will change when analyses are repeated.

 16.521  1 16 12 30  -59 -1250  884 -56 237 328
  80  24   75  142  12  240   72   47
 16.538  1 16 12 55  -60 -1211  885 -57 237 328
  70  19   82  121  12  243   72   56
 16.558  1 16 13 24  -64 -1277  889 -57 237 327
  71  16   78  118  10  237   56   49
 19.530  1 19 12 43  -60 -1250  882 -56 315 330
 105  24   85  241  26  390  106   61
 19.633  1 19 15 11  -58 -1266  882 -50 329 304
```

```
29  1020                   {NLHEAD  FFI}
MERTZ, FRED
PACIFIC UNIVERSITY
ER-2 LYMAN-ALPHA HYGROMETER
1991 TAHITI OZONE PROJECT
1  1                       {IVOL  NVOL}
1991 01 16  1991 01 16     {DATE  RDATE}
1.0                        {DX(1), constant time interval}
30                     {NVPM(1)=number of implied times per marked time}
ELAPSED TIME IN UT SECONDS FROM 0 HOURS ON DAY GIVEN BY 'DATE'
1                      {NV=number of primary variables}
0.01                   {scale factor for primary variable}
999999                 {missing value for primary variable}
WATER VAPOR VOLUME MIXING RATIO IN PARTS PER MILLION
4                      {NAUXV=number of auxiliary variables}
1.0 1.0 1.0 1.0        {scale factors for auxiliary variables}
99 99 99 99999         {missing values for auxiliary variables}
HOUR OF THE DAY (UT)
MINUTES OF THE HOUR (UT)
SECONDS OF THE MINUTE (UT)
OBSERVATION COUNT STARTING FROM TIME COMPUTER IS TURNED ON.
0                      {NSCOML}
6                      {NNCOML}
This is PRELIMINARY data
08:05:01 COMPUTER ON
CALB 8.525E+13    4.829E+7     T =  3.000E+2    DltP =  0.000E+0
NOB  2.799E+2    -1.471E-3    -7.641E+22
OHB  1.279E+2     5.113E-4    -7.641E+22

 29301.0   08 08 21     200
 999999 999999 999999 999999 999999 999999 999999 999999 999999 999999
 999999 999999 999999 999999 999999 999999 999999 999999  87166  84175
  76721  80130  81401  79359  79887  84339  89955  97811  95614  91508
 29331.0   08 08 51     230
  88126  86236  79440  81826  82911  90481  92042  91391  94605  93040
  87099  85103  87131  87423  82418  75260  64485  59903  63633  68262
  72430  75216  78814  77879  72445  69610  66126  60302  55169  48993
 29361.0   08 09 21     260
  39742  39137  38357  38002  36171  35267  36094  38442  40786  41725
  42796  43492  44009  43589  42926  43308  46048  49405  51660  47088
```

```
31  2010               {NLHEAD  FFI}
Mertz, Fred
Pacific University
NMC analyzed grid data interpolated to DC-8 flight path
1991 TAHITI OZONE PROJECT
1  1                       {IVOL  NVOL}
1991 01 16   1991 01 16    {DATE  RDATE}
0.0  30.0                  {DX(1), DX(2)}
8                          {NX(1)}
8                          {NXDEF(1)}
250 200 150 100 70 50 30 10     {X(i,1)}
Pressure levels (mb)
Elapsed time in UT seconds from 0 hours on day given by DATE
3                      {NV=number of primary variables}
1.0   0.1  1.0E-09     {primary variable scale factors}
99999 9999 9999999     {primary variable missing values}
Geopotential height (gpm)
Temperature (K)
Potential vorticity (K m**2/(kg s))
2                      {NAUXV=number of auxiliary variables}
1.0 0.1                {auxiliary variable scale factors}
99999 9999             {auxiliary variable missing values}
Geopotential height of the DC-8 at the indicated time (gpm)
Temperature at DC-8's position at the indicated time (K)
0                      {NSCOML}
5                      {NNCOML}
The geopotential height, temperature, and potential vorticity values were
interpolated from NMC analyses to a vertical cross-section along the
DC-8 flight path.
NOTE: PRELIMINARY data.
     250mb   200mb   150mb   100mb    70mb    50mb    30mb    10mb
  3350  1127 2682
      9994   11395   13219   15762   17970   20000   23016   29411
      2150    2154    2156    2115    2082    2042    1991    2021
      4119    7050    8030   11300   16200   23500   50300  386000
  3380  1289 2671
      9992   11393   13217   15760   17968   19998   23013   29408
      2151    2154    2156    2115    2081    2041    1990    2020
      4128    7050    8040   11300   16200   23500   50400  386000
  3410  1479 2653
      9990   11392   13215   15759   17966   19996   23010   29404
      2151    2154    2156    2115    2081    2041    1990    2020
      4138    7060    8050   11400   16200   23500   50500  386000
```

```
38  2110               {NLHEAD  FFI}
Mertz, Fred
Pacific University
ER-2 Microwave Temperature Profiler (MTP)
1991 TAHITI OZONE PROJECT
1  1                       {IVOL  NVOL}
1991  1 16  1991  1 16     {DATE  RDATE}
0.0  0.0                   {DX(1), DX(2)}
Remote sensing "applicable altitude" (meters)
Elapsed UT seconds from 0 hours on day given in DATE
2                      {NV=number of primary variables}
0.1 0.1                {scale factors for primary variables}
9999 9999              {missing values for primary variables}
Brightness temperature (C)
Potential temperature (K)
15                     {NAUXV=number of auxiliary variables}
1.0 1.0 1.0 1.0 1.0 0.1 1.0 0.1 0.1 0.01 0.001 0.1 0.1 1.0 1.0      {ASCAL}
99 99 99 99 99999 999 999 9999 9999 9999 99999 999 999 999 999      {AMISS}
Number of "applicable altitudes" recorded in subsequent data records
Hours (UT)
Minutes (UT)
Seconds (UT)
Pressure altitude of ER-2 (ft)
Aircraft pitch (deg)
Aircraft roll (deg)
Horizon brightness temperature (C), average of Chan 1 & 2 brightness temp.
Potential temperature (K) from above horizon temp. and aircraft press.alt.
dT/dz (K/km), from Chan 1 & 2 blended Temperature profile
dTHETA/dp (K/mb); THETA is potential temperature
dT/dz (K/km) from Chan 1
dT/dz (K/km) from Chan 2
Peak downward acceleration (centi-G's)
Peak upward acceleration (centi-G's)
0                      {NSCOML}
3                      {NNCOML}
The brightness temperatures are approximately equal to air temperatures at
ER-2 altitudes.

  29589  5  8 13  9  44890  24   1 -728 3459  440   996  49  34  53   9
 14060 -729 3516
 13940 -728 3499
 13810 -731 3474
 13680 -728 3459
 13560 -740 3421
  29603 15  8 13 23  45170  24   2 -712 3500  -17  -679 -11  -4  56  10
 15030 -721 3688
 14780 -719 3650
```

```
37  2160               {NLHEAD  FFI}
Mertz, Fred
Pacific University
WMO coded teletype transmission of upper air soundings
1989 AASE
1  1                       {IVOL  NVOL}
1989  1 16   1989  1 16    {DATE  RDATE}
0                          {DX(1)=0 for nonuniform pressure interval}
5                    {LENX(2)=length of character string defining station
Pressure level (hPa)
Radiosonde station identifier (BBSSS), BB=block #, SSS=station code.
5                    {NV=number of primary variables}
1.0 0.1 0.1 1.0 0.1          {scale factors for primary variables}
99999 9999 999 999 9999      {missing values for primary variables}
Geopotential of the isobaric surfaces in geopotential meters (gpm).
Air temperature at the isobaric surfaces in degrees Celsius (C).
Dew-point depression at isobaric surfaces (C).
Wind direction at the isobaric surfaces (degrees).
Wind speed at the isobaric surfaces in knots.
9                      {NAUXV=number of auxiliary variables}
1                      {NAUXC= # of char. string auxil. variables}
1.0 1.0 1.0 1.0 1.0 0.01 0.01 1.0    {scale factors for auxil. variables}
999 99 99 99 99 99999 9999 9999      {missing values for auxil. variables}
30                                   {LENA=length of char.auxil.variable}
zzzzzzzzzzzzzzzzzzzzzzzzzzzzzz
Number of pressure levels in the sounding.
Year of sounding, last two digits (UT).
Month of year (UT).
Day of month (UT).
Hour of day (UT).
East longitude of station (deg).
Latitude of station (deg).
Elevation of station above MSL (m)
Station name
0                      {NSCOML}
1                      {NNCOML}
Ship stations are in block 99.
71082
  4  89  1 16 12  -6233  8250   66
Alert/Ellesmere Island
 850.0   1136  -331   48  235   330
 700.0   3498  -363   36  999  9999
 500.0   4770  -467   50  235   420
 400.0   6230  -541   60  235   490
99C7C
 14  89  1 16 12  -3550  5270    0
zzzzzzzzzzzzzzzzzzzzzzzzzzzzzz
1014.0      0    16   39  270   290
1000.0    118     6   27  280   210
```

```
33  2310               {NLHEAD  FFI}
Mertz, Fred
Pacific Univ.
DC-8 DIAL ozone number densities
1991 TAHITI OZONE PROJECT
2  7                       {IVOL  NVOL}
1991  1 16   1991  2 20    {DATE  RDATE}
0.0                        {DX(2)}
Geometric altitude of observation (m)
Elapsed time in UT seconds from 0 hours on day given by DATE
1                      {NV=number of primary variables}
1.0E+09                {scale factor for primary variable}
99999                  {missing value for primary variable}
Ozone number density (#/cc)
9                      {NAUXV=number of auxiliary variables}
1.0 1.0 1.0 1.0 1.0 1.0 1.0 0.01 0.01    {scale factors for auxil. var.}
999 99999 999 99999 99 99 99 99999 9999  {missing values for auxil. var.}
Number of altitudes for current time mark
Geometric altitude at which data begins (m)
Altitude increment (m)
Geometric altitude of aircraft (m)
Hour of day of observation (UT)
Minutes of hour (UT)
Seconds of minute (UT)
East longitude of aircraft (deg)
Latitude of aircraft (deg)
0                      {NSCOML}
5                      {NNCOML}
VERTICAL AVERAGING INTERVAL: 975 METERS AT 1-7 KM ABOVE AIRCRAFT
                             2025 METERS > 7 KM ABOVE AIRCRAFT
                  (TRANSITION RANGE VARIES WITH SIGNAL STRENGTH)
HORIZONTAL AVERAGING INTERVAL:   60 KM

 30335   26 12819  75 10389  8 25 35 -13324  -945
  1340  1519  1660  1779  1868  1939  1973  1992  1989  1955
  1934  1897  1817  1721  1619  1514  1434  1343  1258  1203
  1140  1088  1037   956   892   878
 30360   22 12819  75 10383  8 26  0 -13322  -993
  1351  1523  1658  1774  1860  1930  1962  1974  1966  1932
  1909  1877  1803  1706  1600  1493  1407  1310 99999 99999
  1094  1045
 30384   93 12744  75 10378  8 26 24 -13312 -1031
   934  1378  1541  1673  1782  1862  1925  1950  1956  1946
  1912  1884  1843  1765  1667  1565  1457  1375  1279  1194
```

```
23  3010               {NLHEAD  FFI}
Mertz, Fred
Pacific University
NOAA/NMC grid point analyses
1989 AASE
1  1                       {IVOL  NVOL}
1989  1 16   1989  1 16    {DATE  RDATE}
5.0  2.5  12.0             {DX(1), DX(2), DX(3)}
8    3                     {NX(1), NX(2)}
1    1                     {NXDEF(1), NXDEF(2)}
-25                        {X(1,1); X(i,1) = -25 -20 -15 -10 -5 0 5 10}
60.0                       {X(1,2); X(j,2) = 60.0 62.5 65.0}
East longitude (deg)
Latitude (deg)
Elapsed time in UT hours from 0 hours on day given by DATE
2                      {NV=number of primary variables}
1.0E-08  0.1           {scale factors for primary variables}
99999 9999             {missing values for primary variables}
Potential vorticity on 400 K isentropic surface (K m**2/(kg s))
Temperature on 400 K isentropic surface (K)
0                      {NAUXV=number of auxiliary variables}
0                      {NSCOML}
0                      {NNCOML}
   0
   1604   1597   1589   1578   1570   1578   1584   1589
   1598   1583   1561   1534   1506   1478   1447   1446
   1440   1439   1442   1469   1493   1512   1527   1537  {last PV rec}
 2234 2251 2259 2250 2247 2200 2194 2187
 2194 2151 2159 2150 2147 2166 2175 2165
 2121 2136 2140 2140 2138 2127 2111 2104                  {last T rec}
  12
   1532   1522   1509   1492   1472   1467   1459   1450
   1419   1433   1448   1465   1483   1503   1525   1567
   1670   1691   1711   1724   1737   1744   1745   1743  {last PV rec}
 2224 2241 2249 2240 2237 2200 2184 2177
 2184 2141 2149 2140 2137 2156 2165 2155
 2111 2126 2130 2130 2128 2117 2101 2101                  {last T rec}
  24
   1587   1578   1569   1558   1546   1533   1641   1626
```

```
24  4010               {NLHEAD  FFI}
Mertz, Fred
Pacific University
NOAA/NMC grid point analyses
1989 AASE
1  1                       {IVOL  NVOL}
1989  1 16   1989  1 16    {DATE  RDATE}
5.0  2.5  40.0  0.0        {DX(1), DX(2), DX(3), DX(4)}
8    3     2               {NX(1), NX(2), NX(3)}
1    1     2               {NXDEF(1), NXDEF(2), NXDEF(3)}
-25                        {X(1,1); X(i,1)= -25 -20 -15 -10 -5 0 5 10}
60.0                       {X(1,2); X(j,2)= 60.0 62.5 65.0}
400 440                    {X(k,3)}
East longitude (deg)
Latitude (deg)
Potential temperature (K)
Elapsed time in UT hours from 0 hours on day given by DATE
1                      {NV=number of primary variables}
1.0E-08                {scale factor for primary variable}
99999                  {missing value for primary variable}
Potential vorticity (K m**2/(kg s))
0                      {NAUXV=number of auxiliary variables}
0                      {NSCOML}
0                      {NNCOML}
   0
   1604   1597   1589   1578   1570   1578   1584   1589
   1598   1583   1561   1534   1506   1478   1447   1446
   1440   1439   1442   1469   1493   1512   1527   1537 {last 400K rec}
   3135   3151   3175   3198   3220   3240   3260   3278
   3326   3348   3369   3389   3409   3428   3446   3465
   3498   3492   3485   3476   3468   3459   3464   3446 {last 440K rec}
  12
   1532   1522   1509   1492   1472   1467   1459   1450
   1419   1433   1448   1465   1483   1503   1525   1567
   1670   1691   1711   1724   1737   1744   1745   1743 {last 400K rec}
   3424   3419   3409   3396   3379   3354   3327   3297
   3193   3158   3125   3095   3065   3037   3011   2998
   2956   2938   2920   2914   2909   2906   2905   2906 {last 440K rec}
  36
   1587   1578   1569   1558   1546   1533   1641   1626
```