Data Storage Standards for the Atmospheric Sciences

David Hooper
STFC Rutherford Appleton Laboratory,
Chilton, Didcot, OX11 0QX, UK

13th May 2009

Abstract

In order to ensure the long-term usefulness of scientific data, it is essential that they are recorded using a commonly-readable file format, which should ideally be self-describing. Even more importantly, the files should include appropriate items of metadata, i.e. information about the data. This lecture will focus on the use of the Climate and Forecast (CF) metadata conventions, which have been designed for use together with the netCDF file format. They are designed to capture details which are often common-knowledge within the research groups who operate instruments but which might not be documented elsewhere. Consequently they are equally as important for current data usage, particularly where files are exchanged between different research groups, as they are for ensuring the long-term usefulness.

Preamble

These notes were written to accompany a lecture given at the Radar School, held 12th - 16th May 2009, which preceded the 12th International Workshop On Technical and Scientific Aspects of MST Radar (MST12), held 17th - 23rd May 2009 in London, Ontario (Canada). The following abbreviations are used:

BADC	British Atmospheric Data Centre
CASPAR	Cultural, Artistic and Scientific knowledge for Preservation, Access and Retrieval
CEDA	Centre for Environmental Data Archival
CF	Climate and Forecast (metadata convention)
csv	comma separated variable (file format)
ISO	International Organization for Standardization
MST	Mesosphere-Stratosphere-Troposphere
NERC	Natural Environment Research Council
netCDF	network Common Data Form
NTP	Network Time Protocol

Coloured text is used to represent the following:

`File names`
`File contents`
`Computer commands`
`netCDF data types`
`netCDF attributes`
`netCDF dimensions`
`netCDF variables`
Internal and external hyper links

Appropriate web links are given in the Section 9.

1 Introduction

The best way to demonstrate the importance of adopting commonly-accepted data storage standards is to consider the contents of a file which follows none. Below are shown the first 10 lines from a file named `sw010203`. This is taken from the archive of the NERC (Natural Environment Research Council) MST Radar Facility at Aberystwyth (in the UK):

```
4.31 155.3   3.92 136.1   5.15 140.2   4.23 137.1   4.75 150.2   4.71 137.9
4.35 146.5   4.52 138.0   4.83 153.7   5.40 145.8   4.63 141.0   4.90 137.3
4.31 143.3   4.58 157.0   4.94 141.7   4.65 143.1   4.63 143.0   4.88 149.5
5.42 148.5   4.92 140.4   4.04 146.7   3.92 151.5   5.02 135.3   5.06 151.6
4.65 152.3   4.31 168.8   3.79 145.3   5.92 152.9   5.02 145.8   4.77 161.6
4.79 144.1   4.60 147.5   5.33 150.1   4.81 141.0   6.02 146.9   4.38 149.0
4.42 142.5   4.58 133.4   4.35 150.5   4.96 149.8   5.56 143.4   5.08 148.5
5.19 141.6   4.40 142.4   4.10 152.6   5.02 134.0   4.94 142.9   5.27 144.4
5.38 141.5   5.88 144.8   6.00 140.1   4.75 158.3   5.08 148.1   5.46 163.5
4.27 150.8   4.69 138.8   5.71 144.0   5.21 138.8   5.00 132.4   5.06 144.4
```

There is nothing in the file's name or contents to give any indication as to the format (should data be read first column by column and then row by row or *vice versa*?) or as to the type of data that it contains. By itself, this file is of no value. Nevertheless, simply from knowing that the file name prefix `sw` indicates surface wind data (i.e. speed and direction from a wind vane and anemometer) recorded at the Frongoch location, which is 3 km to the west of the MST radar site (at Capel Dewi, near Aberystwyth), it is possible to make an educated guess as to how the contents should be read and interpreted.

The ISO 8601 specification for representing a date, in a basic format, is through a character string of length 8 `YYYYMMDD`, where `YYYY` represents the year, `MM` represents the month [01 - 12] and `DD` represents the day [01 - 31]. File names which incorporate the date in this format will be arranged in increasing chronological order when listed alphabetically following the ASCII convention. The use of only two characters to represent the year, as in this case, should be avoided as the interpretation becomes ambiguous. Nevertheless, assuming that a `YYMMDD` format has been adopted, the string `010203` should be interpreted as representing 3rd February 2001, and not as 1st February 2003 (following the standard British convention) or as 2nd January 2001 (following the North American convention). It must also be assumed that this relates to the period 00:00:00 to 23:59:59 in UTC rather than in Local Time. Although it would be acceptable to use some form of local time (as long as the time zone is also recorded), a fixed frame of reference should be used for countries which adopt daylight saving during the summer months.

The data are clearly grouped into pairs, e.g. the first pair (top-left) being `4.31 155.3`, which can only be interpreted as representing first speed (presumably in units of m s^{-1}) and then direction (presumably in units of degrees from North - although it is not yet clear whether this follows the meteorological or the vector convention, i.e. whether the values represent the direction from which or towards which the wind is blowing). Moreover, these pairs are spread across 6 columns (left to right) and 240 rows (top to bottom), giving a total of 1440 values each. Consequently it can be inferred that speed and direction data are recorded at 1 minute intervals, since there are 1440 minutes in a day. By plotting time series of these values, and looking for unrealistic discontinuities, it is possible to infer that the data should be read first column by column and then row by row.

This was exactly the procedure that I had to follow when I inherited responsibility for these files. There was no documentation relating to them at that time. In order to determine which convention had been used for wind direction, I was obliged to compare the data with those from the lowest range gates observed by the nearby MST radar. However, irrespective of whether the meteorological

or vector convention was assumed, there were often pronounced inconsistencies between the two sets of data. The only way in which the surface wind data made sense was if it was assumed that the actual meteorological convention wind direction was equal to minus the recorded value. This had presumably arisen from an incorrect attempt to convert from meteorological convention to vector convention (which is actually given by adding 180^o to the original values). It turns out that one of the data users had been aware of this problem, although it was not recorded anywhere.

2 Requirements for ensuring that data remain useful

Clearly it can be possible to decode the contents of an ASCII data file - even one with no header information - so long as the potential user has some idea of the expected contents (this is less true of binary files). However, no-one will be prepared to make this level of effort unless they have a strong need to access the data, e.g. to re-examine a case study which has previously been described in the published literature. Even then, the potential user will be obliged to make a number of important assumptions, including that the name of the file has not been altered (since the date to which the data correspond is not recorded anywhere else). Perhaps less obviously, the potential user may still be prevented from utilising the data for their intended purpose unless they know something about the type of instrument used, where it was located and how it was operated. This issue is highlighted by present-day climate research which makes use of data from a variety of historical sources in order to characterise earlier conditions. The data were never collected with such a purpose in mind. Nevertheless, without knowing something about the characteristics of each dataset, it would not be possible to combine them to give a global and (time) continuous picture. For example, changes in something as simple as the type of buckets used to sample sea water have affected the characteristics of sea surface temperature measurements.

It is impossible to guess the uses for which today's data might be required in the future. Consequently data creators should aim to record as much information as would be required by a contemporary scientist who knows nothing about the data in advance. For example, in the context of the data shown in the previous section, what does "surface" wind mean? That the measurements were made within a few cm of the ground? That they were made at an elevation of 10 m above the ground (a common meteorological standard and the one used for the Frongoch instrument since late 2001)? That they were made from an elevation of 5 m above the ground (not a common standard but this was true of the Frongoch instrument between 1995 and late 2001)? The height of the instrument above the ground determines the relative importance of surface friction, which affects both the speed and direction of the wind. Moreover, what sort of wind vane and anemometer were used? What are their accuracy characteristics? How often is the instrument recalibrated? Were the data recorded by a network time protocol (NTP) enabled computer or from an instrument with a free-running clock (which can drift by a few minutes over several months). What is the representativeness of the data?

A new surface met instrument (a "Vaisala Weather Transmitter WXT510", which happens to measure wind speed and direction in addition to the required temperature, pressure and humidity) was installed at the MST Radar site in late 2007. Great care was taken to align the instrument with true north so as to maximise the accuracy of the wind direction measurements. However, the agreement between the winds measured at the Capel Dewi and Frongoch locations (just 3 km apart) was found to be very poor. A histogram of the Capel Dewi wind directions revealed a strongly bimodal distribution with peaks in the ENE and WSW directions. The Capel Dewi site is located within an ENE-WSW-aligned valley which rises up by almost 100 m on either side. The wind data recorded at this location are consequently of limited use for representing the broader-scale low-level flow. It it is clearly important to know something about the the nature of the landscape surrounding the instrument, in this case, in addition to knowing its latitude, longitude and altitude above mean sea level. In general, it might also be important to know something about the reason for making the

measurements in the first place. This might be particularly true when the data were recorded as part of a field campaigns. The objective of the campaign may have a strong influence on the type of instruments used and the way in which they are operated.

The above are all examples of metadata, i.e. information about the data. Typically such information is common knowledge within the research group responsible for the instrument. It may even be documented on a website or in a published article. However, it is also often the case that such information is not documented anywhere. Under such circumstances, the information is vulnerable to being lost when the people responsible for collecting the data retire or move to other jobs. Therefore the data may become scientifically-useless even if they are easy to read.

The requirements for ensuring that the data are potentially useful to future scientists can be summarised as follows:

- the data should be written using a commonly-readable format
- the data should have items of relevant metadata associated with them

Metadata are often recorded in the form of free text, which is ideal for a human reader. These will be referred to as verbose metadata. They can alternatively be in the form of words or phrases which are designed to be machine-readable and so must be used exactly as they are defined. These will be referred to as "controlled vocabulary" metadata. Metadata can even be in the form of numerical values, e.g. relating to observation or processing parameters. The ideal solution is to bind the metadata to the data in the same file. In this case, a self-describing file format should be used.

In order to fully describe a particular dataset, the potential scope of the metadata is effectively infinite. The problem is made worse if commonly-understood technical terms have to be defined explicitly. Consequently a set of metadata conventions should be adopted which is specific to the intended user community. Scientists who make use of data from the NERC MST Radar Facility are generally:

- English-speaking
- numerate and are capable of writing computer programs (they are typically either studying for or have reached BSc level in a mathematical or physical science)
- have some familiarity with atmospheric science

The Climate and Forecast (CF) metadata conventions, which will be discussed in detail in Section 5, are appropriate for this purpose. CF conventions have been designed specifically for use together with the netCDF file format, which will be discussed in Section 3. Nevertheless, as will be described in the following section, the same principles can be applied to other types of files.

The British Atmospheric Data Centre (BADC) is responsible for archiving data from a large number of sources - including from the NERC MST Radar Facility. A variety of file formats have been used and the amount of metadata included is highly variable. Consequently many of these datasets will already be of limited scientific value and others will cease to be as soon as there is a gap in the continuity of people making use of them. The BADC recommends that the following file naming convention is adopted in addition to the use of netCDF files and CF conventions. This naming convention allows potential data users to get an idea of the contents of a file without having to read it. Consequently data creators are advised to use something similar even if they do not intend to archive their files with the BADC. (As an aside, it is useful to adopt a regular naming convention for files of any sort, including those containing documents and digital images).

`instrument-name_location-name_YYYYMMDD[hh][mm][ss][_extra].ext`

`instrument-name` is the name of the instrument taken from a (locally-maintained) controlled vocabulary. This is `wind-sensors` for newer versions of surface wind data files from the NERC MST Radar Facility. Other instrument names include `met-sensors` for surface met sensors, `radar-mst` for the MST radar, and `lidar-ld40` for a Vaisala LD40 laser ceilometer. These names were chosen to be (broadly) consistent with those already used for other datasets stored by the BADC. The BADC have recently updated their recommendations so that the instrument name should be prefixed with an institute name (`nerc-mstrf-` in this case). This was in response to the situation of several groups recording the same type of data at the same location during large-scale field campaigns.

`location-name` is the name of the location at which the instrument was operated. This is also taken from a (locally-maintained) controlled vocabulary. The NERC MST Radar Facility uses the location names `frongoch` and `capel-dewi`. The more-general location name `aberystwyth` would have been ambiguous as the `frongoch` and `capel-dewi` sites are 3 km apart.

`YYYYMMDD[hh][mm][ss]` represents the date (and possibly time - the fields shown in square brackets are optional) in UTC for which the data were recorded. Note that the encoding of time information does not strictly follow the ISO 8601 specification, for which the character `T` should be used to separate the date and time fields.

`extra` represents any additional information, e.g. to differentiate between different levels of data from the same instrument (`spectra`, `radial` and `cartesian` in the case of the MST radar), different range resolutions, or even different versions of the data.

`ext` represents the file type, i.e. `nc` for a netCDF file.

3 An introduction to netCDF

The name netCDF (network Common Data Form) covers a set of software libraries and machine-independent file formats which support the creation, access, and sharing of array-oriented scientific data (e.g. time series, altitude profiles, or time-altitude data). In order to read and write netCDF files, you will need to download and install the following freely-available software from the Unidata website (see Section 9):

- the netcdf software package. This allows direct access to the data through a C, C++ or Fortran program. It also provides the facility for viewing file contents by simply typing the following at the command line:

  ```
  ncdump -h netcdf_file_path
  ```

  and substituting the appropriate file path for `netcdf_file_path`. Omitting the -h option causes the data values, as well as the data structure, to be shown. Examples of output from this command will be considered shortly.

- a suitable interface program if you require access to the data through a language other than C, C++ or Fortran. Interfaces are available for a large number of languages including Matlab, IDL, CDAT, Python and Perl.

Users of netCDF files only need to be aware of the the simple conceptual structure (described in Section 4). They do not need to know anything about the exact file structure (at the byte level), which is handled entirely by the netcdf software. This makes adapting to unfamiliar files (e.g. from different data producers, created on different hardware platforms, or created under different operating systems) extremely easy. Nevertheless, it is netCDF's ability to store unlimited amounts of

metadata which makes it particularly important from the perspective of this lecture.

All of the entries within a "classic" netCDF format file (this is the default format although 3 newer file formats are available) are of one of the following data types. All should be assumed to be arrays, although some can simply be of length 1. The characters in the left-hand column correspond to the abbreviations used by ncdump program.

b	byte	8-bit signed integer
""	char	8-bit unsigned integer (strings are technically arrays of type char)
s	short	16-bit signed integer
i	int	32-bit signed integer
f	float	32-bit floating point
d	double	64-bit floating point

The newer "netCDF-4" format allows the following additional data types:

ushort	16-bit unsigned integer
unint	32-bit unsigned integer
int64	64-bit signed integer
unint64	64-bit unsigned integer
string	variable length character string
bool	(8-bit) Boolean

One of the limitations of netCDF is that it can only be read by a limited number of applications. Many of the BADC's data users do not have significant programming abilities and rely on Microsoft Excel for all data analysis and visualisation (typically of one-dimensional datasets). This package cannot (at present) be used to read netCDF files. Although the ASCII-based NASA-Ames format (which is also recommended by the BADC) can be read by Excel, it has a limited capacity for storing metadata. Consequently the BADC has recently developed a specific implementation of a comma separated variable (csv) format, for use with one-dimensional data, which makes use of CF conventions. Links for both the NASA-Ames and BADC-CSV formats can be found in Section 9.

4 Conceptual structure of a netCDF file

A netCDF file is composed of just three basic entry types: global attributes, dimensions and variables (which can themselves have attributes).

Global Attributes Each one has a data type, a name and a value. It stores an item of metadata which is relevant to the file contents as a whole. These are typically arrays of type char, i.e. strings, and they can be spread over several lines (in which case the data producer should insert \n characters in order to indicate the location of the line breaks). A global attribute can alternatively be in the form of a numerical value, e.g. recording the value of an observation or a processing parameter. The global attributes required by the CF convention will be described in the next section.

Dimensions Each one has a name and a length. All of the other details relating to the dimension, e.g. the coordinate values, are stored by a variable (described below) of the same name as the dimension which is known as a coordinate variable. A netCDF dimension can be used to represent a real physical dimension such as time, latitude, longitude, or altitude. It can alternatively be used to index other quantities such as a signal component number or station number. The file creator can add as many dimensions as they wish to a file. Different variables can use different combinations of the available dimensions. As will be seen shortly, some variables may even have no dimensions associated with them.

Variables Each one has a data type, a name (the file creator is free to choose any name they wish for a variable), a shape (which is indicated by an ordered list of the dimensions that it uses), and an array of values (of the shape and size implied by the dimensions). A variable may also have any number of attributes attached to it. These are identical in nature to the global attributes described above except that they apply only to the variable to which they are attached.

The structure of a netCDF file is demonstrated below using the (partial) contents of an actual file named nerc-mstrf-wind-sensors_capel-dewi_20080114_wxt510.nc which contains 1 minute interval surface wind data from the Capel Dewi site for 14th January 2008. The layout closely follows the format produced by the ncdump program.

```
global attributes:
        :verbose_metadata = "Free text description" ;
        :file_version_number = 1s ;
        :data_year = 2008s ;
        :data_month = 1s ;
        :data_day = 14s ;
dimensions:
        time = 1440 ;
variables:
        float longitude() ;
                longitude:units = "degrees_east" ;
                longitude:axis = "X"
        float latitude() ;
                latitude:units = "degrees_north" ;
                latitude:axis = "Y" ;
        float altitude() ;
                altitude:units = "m" ;
                altitude:axis = "Z" ;
        int time(time) ;
                time:units = "seconds since 2008-01-14 00:00:00 +00:00" ;
                time:axis = "T" ;
        float mean_wind_speed(time) ;
                mean_wind_speed:units = "m s-1" ;
                mean_wind_speed:coordinates = "latitude longitude altitude" ;
                mean_wind_speed:cell_methods =
                                        "time: minimum (interval: 3 s)" ;
                mean_wind_speed:missing_value = 99.9f ;
        short mean_wind_direction(time) ;
                mean_wind_direction:units = "degree" ;
                mean_wind_direction:coordinates =
                                          "latitude longitude altitude" ;
                mean_wind_direction:cell_methods =
                                        "time: minimum (interval: 3 s)" ;
                mean_wind_direction:missing_value = 999s ;
```

The verbose_metadata global attribute is entirely fictitious and has been added to demonstrate the principle of including verbose metadata in a global attribute. Actual global variables of this type will be described further in Section 5. The other global variables, which actually exist, simply contain numerical values which indicate the year, month and day on which the data were recorded and the file version number. The variable time is the coordinate variable for the dimension time. Note that this adopts the standard method of representing time, i.e. as seconds past 00:00:00 for the date

in question. The final part of the units attribute, i.e. +00:00, indicates that UTC is used. It is permissible to adopt any other time zone. The significance of the variables longitude, latitude and altitude, and of the attributes for variables mean_wind_speed and mean_wind_direction, will be described in the following section.

5 The Climate and Forecast (CF) metadata conventions

The CF metadata conventions can be used for data relating to the atmosphere, the Earth's surface and the oceans (including sea ice). They were designed principally with model-generated data in mind, although they can easily be used for observational data. They require that the global attribute Conventions is included in each file and recommend that the global attributes title, institution, source, history and references also be included. These are all arrays of type char and their names are taken from a controlled vocabulary. Note that only the attribute name Conventions begins with an upper-case character.

Conventions Describes the version of CF metadata conventions followed in creating the file. The value is taken from a controlled vocabulary (all of the other required global attributes contain free text). The most recent version (at the time of writing) is indicated by the value CF-1.4. Versions are backwards compatible so that a file created using CF-1.0, for example, will still be compliant under CF1.4.

title A short description of the contents of the file, e.g. giving the nature of the data. For the example surface wind data file this has the value Surface meteorological data (wind) from the NERC MST Radar Facility at Aberystwyth.

institution Details of the institution(s) responsible for acquiring, processing and storing the data. For the example file this has the value

Data recorded by the Natural Environment Research Council (NERC) Mesosphere-Stratosphere-Troposphere (MST) Radar Facility at Aberystwyth - http://mst.nerc.ac.uk
Data processed by the Rutherford Appleton Laboratory, Space Science and Technology Department - http://www.sstd.rl.ac.uk
Data held at the British Atmospheric Data Centre - http://badc.nerc.ac.uk/data/mst

source The name of the instrument or the model from which the data were produced. For the example file this has the value Vaisala Weather Transmitter WXT510.

history The details of where, when and how any operations were applied to the data, e.g. describing the original acquisition, any processing steps which have been applied, and the creation of the files in question. These details provide a data audit trail which can be invaluable, for example, in distinguishing between different versions of data. For the example file this simply has the value File created 2009-04-21 01:00:03 +00:00 on machine claudius.

references The details of any relevant documentation describing e.g. the instrument, the measurement technique, the method of data processing, or the quality of the data. In principle, the metadata for a given dataset encompass all of the documents that relate to it, including the contents of its website, articles published in the refereed literature, and the proceedings from conferences. Clearly it would be impractical to fit all of these into each file. Nevertheless, the details of a few key publications should be included. See Section 7 on how to link publications back to a dataset. Although facility websites are useful resources during the active lifetime of an instrument, they cannot be expected to persist in the longer term. There are (as yet) no non-web-based references for the instrument referred to by the example file. The value of

this global attribute is consequently,
`Basic information about the data is available at http://badc.nerc.ac.uk/data/mst More detailed information about the data is available at http://mst.nerc.ac.uk The manufacturer's details about the instrument can be found at http://www.vaisala.com`.

`comment` Any relevant details which are not covered by the other global attributes. Even if suitable references exist, it is useful to provide some description of the instrument, the measurement technique, the method of data processing, and the quality of the data. This is particularly important for one-of-a-kind instruments, which are familiar to only a limited group of scientists, although it is still useful for more-widely-available manufactured instruments. The value of this global attribute for the example file is rather long and so it is shown separately in Section 6.

Any other global attributes that the file creator chooses can be added to the file. It is a good idea to add some sort of file version number so that changes can be tracked if improvements are made.

The CF convention also recommends that the following attributes are attached to each variable (where possible). The `units` attribute must be included for coordinate variables.

`standard_name` An array of type `char` taken from a controlled vocabulary. Standard names only exist for parameters which have an agreed and unambiguous definition. They are only added to the CF "standard name table", together with their definition and "canonical units" (see below), after a period of open discussion and review within the CF-users' community. Standard names exist for most of the commonly-used atmospheric parameters. For example, the variable `mean_wind_speed` can be assigned the standard name `wind_speed`, which has canonical units `m s-1` and is defined as: `Speed is the magnitude of velocity. Wind is defined as a two-dimensional (horizontal) air velocity vector, with no vertical component. (Vertical motion in the atmosphere has the standard name upward_air_velocity.) The wind speed is the magnitude of the wind velocity.` The variable `mean_wind_direction` is given the standard name `wind_from_direction`, with canonical units `degree`, to indicate that it follows the meteorological convention. The standard name `wind_to_direction` would have been used if the vector direction convention had been used. The wind speed and direction information could alternatively have been given in terms of `northward_wind` and `eastward_wind`, both of which have canonical units of `m s-1`.

`long_name` An array of type `char`. This is, in effect, a free text version of the standard name and might be used, for example, as a label on a plot. It does not need to be included if a standard name exists but it is essential otherwise. Its value is `mean wind speed over 3 s of sampling` for variable `mean_wind_speed` and `mean meteorological convention wind direction over 3 s of sampling` for variable `mean_wind_direction`

`units` An array of type `char` taken from a controlled vocabulary. For parameters which feature in the standard name table, the "canonical units" represent the default units. Nevertheless, it is permissible to use any other units which are physically equivalent (but not necessarily identical) to the canonical units. In order for a file to be CF-compliant, these alternative units must be taken from the Unidata "UDUNITS" library - Section 9. This also applies in the case of parameters for which there is no standard name. Dimensionless values, such as relative humidity, are represented by a `units` value of `1`, i.e. the data values will be in the range 0.0 to 1.0. However, it is more typical to think of relative humidity values being in the range 0.0 to 100.0%, in which case the `units` attribute should

be given the value %, which is taken from the UDUINTS library. Radar return powers are commonly described in units of dB, which does not feature in the original UDUNITS library (although it is now covered by the UDUNITS2 library). This has been dealt with, for NERC MST Radar data, by adding a an attribute comment to the signal power variable. This attribute has the value, Powers have dimensions of W. The values are uncalibrated and are stored in dB units, where P_dB = 10.0 * log10(P_linear_units) .

_FillValue An array of length 1 of the same data type as the variable. Its value represents the value assigned to those elements of the variable for which there is a missing datum. This attribute is not used for all variables, e.g. those representing reliability flags. A previous convention required that an attribute named missing_value was used to represent a missing datum value. Both attributes, which have identical values, are used for files generated for the NERC MST Radar Facility.

The CF conventions make a large number of additional recommendations as to how to represent various aspects of the data. However, these are not obligatory. One of these concerns the way in which to represent geographical coordinate information for single station data (CF was originally designed with global climate data in mind and so it was assumed that all variables would use dimensions of latitude, longitude, and possibly altitude). This is achieved by first creating variables longitude, latitude and altitude, which are all numerical arrays of length 1 (containing the appropriate values) and which are not associated with any dimensions - see Section 4. The mean_wind_speed and mean_wind_direction variables are then associated with these coordinates by assigning them each an attribute coordinates of data type char which contains the value latitude longitude altitude, i.e. a space-separated list of the names of the appropriate variables.

A separate recommendation concerns the order in which dimensions should be used by variables. If any or all of the dimensions have the interpretations of date or time (T), height or depth (Z), latitude (Y), or longitude (X), they should be used in the relative order T, Z, Y, and then X. Moreover, the corresponding coordinate variables should be given an attribute axis, an array of type char and length 1, whose value is T, Z, Y or X - see Section 4. All other dimensions should, wherever possible, be placed to the left of the spatio-temporal dimensions. This recommendation has been followed by the MST radar "Cartesian" files (i.e. those containing the eastward, northward and upward components of the wind), which use dimensions of time and altitude in that order. However, in the cases of the spectral files (which use dimensions of range and doppler_velocity - only a single dwell is stored in each file) and the radial files (which use dimensions of time, range and signal_component_number), the non-spatio-temporal dimensions have been placed to right. Note that range has an interpretation of height or depth and so is equivalent (in this context) to altitude.

CF recommends that any statistical operations which have been carried out on the data should be recorded in the value of a variable attribute cell_methods, which is an array of type char. For the data stored in the example file, measurements are made 4 times a second over 3 seconds. From these 12 samples, the minimum, mean and maximum values of speed and direction are recorded. Consequently, although they are not shown in Section 3, the example file contains the variables minimum_wind_speed, maximum_wind_speed, minimum_wind_direction, and maximum_wind_direction in addition to mean_wind_speed and mean_wind_direction. Each of these has an attribute cell_methods whose value is time: minimum (interval: 3 s), time: mean (interval: 3 s), or time: maximum (interval: 3 s). However, it should be noted that the method involved is probably made more clear in the free text description of the global attribute comment - see Section 6.

Data creators may add any other attributes that they wish. A CF-compliant application, e.g. a display program, will correctly process the attributes that it expects to find and will ignore any

which are non-standard. The following are used by various files from the NERC MST Radar Facility.

`valid_range` This is an an array of the same data type as the variable and of length 2 (its use is recommended by CF). It contains values of the minimum and maximum values within which the variable data are to be expected. For example, its values for the wind direction variables in the example file are `0s` and `360s` - as opposed to the equivalent range between `-180s` and `180s`. The same information can alternatively be given using two separate variable attributes `valid_min` and `valid_max`. The value of the attribute `missing_value` must be outside of this range.

`flag_values` If an integer-type variable is used as a flag, it should have an attribute named `flag_values` of the same data type as the variable and with a length sufficient to contain all of the permissible values of the variable. Its use is recommended by CF. An additional attribute named `flag_meanings`, an array of type `char`, should contain the interpretation of the flag values as a series of space-separated strings (equal in number to the number of permissible flag values). For example, the MST radar (spectral and radial) files make use of a `short` variable named `data_weighting_window_index`. The `flag_values` attribute is an array of type `short` and length 3 which contains the values `0s`, `1s`, and `2s`. The `flag_meanings` attribute is an array of type `char` which contains the value `rectangular Hanning other`.

`accuracy_details` Although the CF conventions do provide ways of describing model data accuracy, these are not currently well-adapted for applying to observational data. It is better to include an estimate of accuracy than to give no indication of accuracy at all. I have previously made use of an attribute named `accuracy`, which is of the same data type as the variable and of length 1. However, the accuracy often depends on the value of the variable. Consequently an array of type `char` containing free text is better suited for this purpose. Its value for variable `mean_wind_speed` is `+/-0.3 m s-1 or +/-3% (whichever is greater) for wind speed in the range 0 - 35 m s-1, +/-5% for wind speed in the range 36 - 60 m s-1. The wind direction is tightly constrained by the approximately ENE-WSW alignment of the valley within which the instrument is located. The winds cannot therefore be interpreted as being representative of the broader-scale low-level flow.` Its value for for variable `mean_wind_direction` is `+/-3 degree. The wind direction is tightly constrained by the approximately ENE-WSW alignment of the valley within which the instrument is located. The winds cannot therefore be interpreted as being representative of the broader-scale low-level flow.`

`resolution` Data generated by the Vaisala WXT510 instrument are encoded in a manufacturer-defined ASCII message which imposes a fixed resolution. I have used the non-standard attribute `resolution`, which is of the same data type as the variable and of length 1, to record this information. Its value for the variable `mean_wind_speed` is `0.1f` and its value for `mean_wind_direction` is `1s`.

6 An example use of the global attribute "comment"

The following is the value contained within the global attribute `comment` for the netCDF files containing surface wind data from Capel Dewi.

```
 The Vaisala Weather Transmitter WXT510 is a single unit which
measures a variety of surface meteorological parameters in three
independent measurement cycles. Pressure, temperature and relative
humidity (ptu) are measured in the first cycle, precipitation (precip)
in the second, and wind speed and direction (wind) in the third. The
```

data (for a given day) from each of these cycles are stored in separate files with respective names of the format:

nerc-mstrf-met-sensors_capel-dewi_YYYYMMDD_wxt510-ptu.nc
nerc-mstrf-met-sensors_capel-dewi_YYYYMMDD_wxt510-precip.nc
nerc-mstrf-wind-sensors_capel-dewi_YYYYMMDD_wxt510.nc

where YYYYMMDD is an 8-character string representing the date. Missing data values are used to indicate gaps in data collection.

Pressure is measured by a silicon BAROCAP (R) sensor, temperature by a capacitive ceramic THERMOCAP (R) sensor, and relative humidity by a capacitive thin film polymer HUMICAP (R) 180 sensor. These are simultaneously sampled at 60 s intervals.

Precipitation is measured by a RAINCAP (R) piezoelectric sensor, which has an area of 60 cm2. The signal generated by the impact of a single raindrop is proportional to its volume. Internal signal processing is used to differentiate between the signals generated by rain, hail and undesired sources. Nevertheless, occasional small rain accumulations appear to be unrelated to precipitation. Data messages are only generated for the duration of a precipitation event, when they are available at 10 s intervals. The rainfall and hail accumulations represent the amounts since the last non-zero readings. The rates represents the mean values over the preceding 60 s. Within the first 60 s of a precipitation event, the rates are evaluated only over the period since the start time of the event.

Wind speed and direction are measured by a WINDCAP (R) sensor. This consists of an array of three equally-spaced ultrasonic transducers (approximately 11 cm apart) in a horizontal plane. The time it takes for sound to travel between each pair of transducers depends on the temperature and the humidity of the air and on the component of wind vector along the direction joining the transducers. For a given pair of transducers, the effects of temperature and humidity are the same for the sound travelling in both the forward and the reverse directions. However, the effects of the wind vector are opposite. Measurements are required between at least two pairs of non-collinear transducers in order to derive the wind speed and direction. The use of measurements from all three pairs of transducers provides redundancy, which allows the wind information to be derived from the two pairs of transducers which provide the best quality signals. No attempt is made to calculate the wind direction when the speed drops below 0.05 m s-1. Instead, the last calculated direction is used until the wind speed rises back above 0.05 m s-1. Samples of wind speed and direction are taken 4 times a second over 3 seconds. From these 12 samples, the minimum, mean, and maximum values of speed and direction are recorded.

The instrument at the NERC MST Radar site is mounted on a pole at approximately 1.9 m above ground level and at approximately 15 m to the north of the site bungalow. Messages are recorded by a Network Time Protocol enabled data logging computer together with the time (to the nearest second) at which they arrived. Measurements of pressure, temperature and relative humidity are generally in good agreement with those from the Campbell Scientific climate data logger, which is located approximately 50 m further north. Although the rain rates show similar patterns to those measured by a tipping bucket raingauge, the

```
values from the WXT510 tend to have larger peak values and to be more
variable as a function of time. This is consistent with the fact that
the rates are evaluated over 1 rather than 10 minutes. The radar site
is located within an approximately ENE-WSW aligned valley which leads
to a strongly bimodal distribution of wind directions. Consequently
the wind information cannot be interpreted to be representative of the
larger-scale low-level flow. The wind data from the 10 m tower at
Frongoch farm, which is located 3 km to the west, are recommended for
this purpose.
```

7 External documentation

Given that web pages cannot be relied upon to last beyond the lifetime of the instruments to which they relate, efforts should be made to publish key instrument details in other ways. Many large academic organisations now have document repositories which are intended to provide a long-term home for "grey" literature, i.e. non-official documents which are nevertheless useful. This is a good way of preserving information which would not be acceptable in a refereed article. The BADC uses the Centre for Environmental Data Archival (CEDA) this purpose.

It is also now possible to cite datasets in refereed journals in much the same way that articles are cited. This provides an effective way of linking datasets with their broader metadata. The BADC provide the following way to cite data from the NERC MST Radar Facility:

> "Natural Environment Research Council, Aberystwyth Radar Facility, [Hooper, D.]. The NERC Mesosphere-Stratosphere-Troposphere (MST) Radar Facility at Aberystwyth, [Internet]. British Atmospheric Data Centre, 2006-, *Date of citation*. Available from http://badc.nerc.ac.uk/data/mst/"

where an the date of citation should be substituted for *Date of citation*. This follows ISO standard 690.

8 Conclusions

The contents of the obligatory CF metadata fields are pieces of information which data creators should already know and which data users will need to know. Consequently the conventions do not cover anything which is inherently difficult to determine. Nevertheless, the process of defining your first CF-compliant dataset can take quite some time. In particular, it may not be obvious what sort of details to include in the `comment` global attribute. However, this becomes easier for each new dataset that you deal with. If you are generating files which are intended to be used by more than just a few people, the long-term benefits of making them CF-compliant will far outweigh the amount of effort required to do so. At the very least, you should make the effort to include CF-like information in your files. Even a small amount of metadata is better than no metadata at all. Moreover, a clear free text description will probably be of more benefit than simply ensuring that the correct controlled vocabulary terms have been included. Example files from the NERC MST Radar Facility dataset can be accessed through a link given in the following section.

9 References

- British Atmospheric Data Centre (BADC) CSV file format
  http://badc.nerc.ac.uk/help/formats/badc-csv/

- Centre for Environmental Data Archival (CEDA) Document Repository
  http://cedadocs.badc.rl.ac.uk/
- Climate and Forecast (CF) Metadata Convention
  http://cf-pcmdi.llnl.gov/
- International Organization for Standardization (ISO)
  http://www.iso.org/
- NASA-Ames file format (on the BADC website)
  http://badc.nerc.ac.uk/help/formats/NASA-Ames/
- NASA-Ames file format (on the NASA website)
  http://cloud1.arc.nasa.gov/solve/archiv/archive.tutorial.html
- MST12 Radar School and Workshop
  http://www.mst12.com/
- NERC MST Radar Facility resources for this lecture
  http://mst.nerc.ac.uk/mst12.html
- Unidata netCDF
  http://www.unidata.ucar.edu/software/netcdf/
- Unidata UDUNITS
  http://www.unidata.ucar.edu/software/udunits/

10 Acknowledgements

I am grateful to the following who have all provided expertise which has shaped the way in which NERC MST Radar Facility data are now stored: Sam Pepler and Graham Parton (from the BADC), Alison Pamment (who is in charge of the CF standard name table), and Esther Conway (who is working on the CASPAR project - "Cultural, Artistic and Scientific knowledge for Preservation, Access and Retrieval").

This version of the document was finalised on 8th May 2009.