APPENDIX IV

Specification for Shipboard Data Logging

UNOLS-RVTEC
Data Interchange Subcommittee
Marc Willis, OSU-COAS, Chair

The opinions and recommendations expressed here are those of the Chair, and do not necessarily reflect those of anyone else.

Background

- 1993 - Rich Findley initiates discussion of common data formats, DIsC is formed (Willis, Polous (UHI), Nelson (URI), Mulhern (NOAA). DIsC charge is to evaluate various data formats and recommend one.
- 1994 - DIsC recommends use of netCDF as standard logging format, RVTEC adopts netCDF. DisC is charged with developing a specification for logged data.
- 1995 - DISC submits long overdue draft specification for shipboard data logging.

The Specification is intended to provide a common base which can be adapted or added to depending on the needs and desires of an individual institution.

The Specification deals with:

- Shipboard logged data files, their format and types of information they contain.
- Information which is associated with the data files, and information associated with the data items within logged files.
- A recommended sensor suite common to all UNOLS vessels.
- Recommendations for standard identifiers for data items in the "standard" shipboard sensor suite.
- A recommended data distribution specification for onboard, (near) real-time use.

The Specification does NOT deal with:

- Format of data actually delivered to the user, or the medium used for delivery.
- Any aspect of hardware or software other than those pertaining to netCDF.
- Any other product (data or display) delivered to the user.

Recommendations for Logged Data Files:

- netCDF has been adopted as the standard logging format for shipboard data.
- Logged data files should contain raw, untransformed data.
- Data files should have headers which identify the file as to ship or system which generated the file, and some indication of file or software version.
- Data files should always be accompanied by descriptor files (at least one is mandatory). Descriptor files can include sensor identification references, calibration information, etc.

Recommendations for Data Items:

- Sensors should be identified within the netCDF file, or cross-referenced to the descriptor file(s) so that the serial number and other relevant information can be determined.
- Sensor ID should include information on the location of the sensor on the ship.
- Sensor data should include calibration equations, constants and factors, and other relevant calibration information.

Recommendations for "Standard" Parameters:

- Time Information
  - Date
  - Time
  - Consecutive Sample Number
- Navigation Information
  - GPS
  - SavNav (?)
  - Loran-C (?)
- Sea Surface Information
  - Temperature (multiple?)
  - Conductivity (at least one with T)
- Meteorological
  - Wind Speed
  - Wind Direction [Wind Vector]
  - Barometric Pressure
  - Humidity
  - Air Temperature

Identification of Standard Parameters

- Standard Parameters should carry standard identifiers

Onboard Data Distribution

- Data delivered in real-time should conform to the NMEA-0183 standard. NMEA-0183 includes specifications for electrical interface, data format, etc.
- Data delivered in real-time should be transformed, "calibrated" values rather than raw values.

Unresolved Issues:

- Decomposition of combination data items?
- Derived variables?
- Multiple real-time streams?
- Is the specification detailed enough?
- Implementation?
- Others

A "Schematic of Typical Shipboard Data Acquisition and Logging System" that

appears here in this appendix. Copies of this schematic can be obtained from the UNOLS Office.

********DRAFT*******

A Specification For Shipboard Data Logging

UNOLS-RVTEC Data Interchange Subcommittee Marc Willis, OSU-COAS (chair)

Draft 24 March 1995, rev 15 June 95, 21 Sep 95

Disclaimer: 1) The recommendations and opinions expressed in this document reflect those of the subcommittee chair, and do not necessarily reflect those of anyone else. 2) Throughout the document, examples from the Acquisition and Logging system in use on WECOMA are used to illustrate certain points and recommendations. These have been included only as examples, and are not meant to imply that they should be adopted. They are far from perfect. In some cases, the WECOMA examples given do not meet the specification.

Editorial Comment: Standards established in UNOLS, particularly those dealing with technical services, computing, or data acquisition, have largely been a failure. The establishment of standards tends to stifle innovation, and predetermine the direction of development. Developments in the technical services area should be unfettered, with wide latitude for experimentation. innovation and creativity. This document approaches the problem of shipboard data logging from the standpoint of performance rather than specifics. That is, to establish the outcome of the effort, rather than the route by which an individual institution might arrive there. This specification has been written in hopes that it can easily be implemented on shipboard systems from the simplest to the most complex. The specification leaves wide leeway for institutions to pursue a range of development without violating the basic specifications. The specification is intended to provide a background against which development can take place. The specification should not drive new developments, nor determine their direction. The rich variety of shipboard computing efforts now seen in the fleet should not be diminished by this specification. That is not its intention.

Summary of Recommendations:

- **At the 1994 RVTEC meeting held in Miami, netCDF was adopted. as the standard data format for shipboard data logging.**
- **Data files should always be accompanied by descriptor files.**
- **Logged data files in netCDF format require a *.cdl file. Each physical data storage unit which contains a data file (or files) should also contain a copy of any descriptor files.**
- **Data files should have headers which identify the file as to ship or system generating the file, and file version in use.**
- **Raw, untransformed data should be logged in the data files. Transformed data in engineering or scientific units may be added to the file, but should be in addition to the primary raw data.**
- **All data records should include information on time of collection, and averaging information if any. If averaging is used, at least the number of samples, maximum value, minimum value and mean should be recorded.**
- **Sensors should be identified within the data file or cross- referenced to the descriptor file so that the serial number and other relevant Information for each sensor can be easily determined.**
- **Sensor Identification should include information on the location of the sensor on the ship. This is important when a calculated parameter (e.g., sea surface salinity) requires two measurements from the same location.**
- **Sensor data should include calibration equation(s), constants and factors, date of last calibration, where calibrated, and any other calibration-related information which might assist with interpretation of the data.**
- **A Basic Shipboard Sensor Suite should consist of the following:**

- **Date, Time, Sample Number, GPS, SatNav(?), Loran-C(?), Ship Heading, Ship Speed, Sea Surface Temperature, Sea Surface Conductivity, Wind Speed and Direction, Barometric Pressure, Humidity, Air Temperature.**
- **Standard Shipboard Sensor data items In logged files should be identified in a standard way from ship to ship.**
- **Onboard real-time data products should conform to the NMEA-0183 standard. The NMEA standard specifies electrical interface, data format, standard sentence formats, and allows addition of user specific data sentences as "proprietary." The use of "proprietary" sentences should be kept to a minimum. [Where possible, transformed and "calibrated" numbers should be output in the onboard real-time stream, to maximize its usefulness.]**

1. Basic Assumptions and Parameters

- **1.1. There is a recognized need for a common data format for UNOLS vessels.**
- **1.2. The data format recognized as best for this purpose is netCDF (UCAR network Common Data Form.** (There is no discussion of netCDF itself in this document)
- **1.3. The convention is to be established as a minimum set of specifications and parameters. This will not preclude a particular Institution from implementing a superset of these specifications.**
- **1.4. The specification will contain not only the standard format (netCDF), but recommendations for information to be included in the data file.**
- **1.5. There is to be wide latitude for individual institutions to adapt these specifications to their particular needs and capabilities.**
- **1.6. No attempt was made to integrate data from other . sources (CTD, ADCP, Bathy systems, etc.) Into this specification**

2. Definitions

These definitions apply to the discussion which follows. They are not meant to imply any particular arrangement of components or any particular topology of shipboard data acquisition. ITEMS MARKED WITH AN ASTERISK (*) ARE NOT PART OF THE SPECIFICATION.

Note: A Schematic of Typical Shipboard Data Acquisition and Logging System inserted here. It is the same as the schematic discussed previously. Copies may be obtained from the UNOLS Office.

- **2.1. Data Format**
- The standard logical format for data storage. This refers only to the logical format (netCDF), not the medium or system on which the data is stored. It can be used onboard in user processing and display functions but is generally not considered available (or useful) as a real-time data transfer format. It is intended to function as an archiving and logging format for post-cruise use.
- **Data File:** File of data in **data format**.
- **2.2. *Delivery Format**
- The format (logical and physical) in which the data is delivered to the user. This may include flat ASCII files in various forms (comma-separated, tab-separated, flat tables), particular database formats, and may include real-time serial ASCII output from acquisition systems, network access to data in (near) real-time, etc. **Delivery formatted** data is that delivered to the user in the most useful format. Data in the **delivery format** may be in any form (i.e., processed, averaged, engineering units, raw data, etc.). This may also include physical media such as floppy disks, removable media, rewriteable optical disk, and CD-ROM. Individual institutions should decide what delivery format they wish to use; this specification does not address **delivery format**. The most logical **delivery format** may or may not correspond to the **data format**.

- **Delivery File:** Files of data in **delivery format**.
- **2.3. Logged Data Form**
- The form (raw, processed, engineering units) in which the data
- is stored in the logged **data format**.
- **2.4. *Sensor**
- Any point source of data. Includes analog and digital sensors,
- direct ascii and binary sources of data.
- **2.5. Sensor Information**
- Information associated with each sensor. Can include type, calibration constants and equations, identifiers, and any other information significant to use of the **sensor. Sensor information** may be included in the logged data file, or may be contained in one or more **descriptor files**.
    - **Descriptor file**
    - File or files containing information describing data in the **data** or **delivery files**.
    - A mandatory **descriptor file** for netCDF files is the *.cdl file for each **data file**.
- **2.6. *Acquisition Interface**
- Interface point between **sensor** and **logging system**. Most common types include serial interfaces and buffers and analog-to-digital converters.
- **2.7. *Acquisition Subsystem**
- Computer or other system which concentrates data from suites of sensors for transmission to the **logging system** An example is a meteorological subsystem. May be integrated with the **logging system**.
- **2.8. *Logging System**
- Computer system integrating and processing incoming **sensor** data from **acquisition interfaces** and logging it in the **data format**. This system may also produce data in the **delivery format**.
- **2.9. *Processing system**
- Computer system or part of **logging system** which converts sensor data into meaningful units for use onboard ship in (near) real-time.
- **2.10. *Display System**
- Separate computer system or part of **logging system** which displays data for ship users in (near) real-time.
- **2.11. Onboard Data Distribution System**
- Hardware/software system which allows users real-time access to data streams.

3. RVTEC Specificationf or Shipboard Data Formats

- **3.1. Data Format**
    - **RECOMMENDATION 1: At the 1994 RVTEC meeting hold in Miami, netCDF was adopted as the standard data format for shipboard data logging.**
    - **3.1.1. Data Files**
    - Data files should be of a size appropriate to the system on which they are written, media available, etc.
    - Each disk or other storage unit should contain not only the appropriate data file, but a copy of any other **descriptor file** necessary to interpreting the **data file** on that disk or storage unit.
    - **RECOMMENDATION 2: Data files should always be accompanied by descriptor files. Logged data files in netCDF format require a *.cdl file. Each physical data storage unit which contains a data file (or files) should also contain a copy of any descriptor files.**
    - Attachment 1 gives an example of a data and descriptor file system as used on WECOMA. This is only an example, and does not imply adoption of this system by anyone else.
    - **3.1.2. Header Information**
    - Each netCDF file should include information identifying the file as to which ship or system generated the file, and what file version is in use. This will be necessary because while the

basic format (netCDF) will not change in the short term, the information in the files will change as sensors are added or changed.
    - **RECOMMENDATION 3: Data files should have headers which identify the file as to ship or system generating the file, and file version in use.**
- **3.2. Logged Data Form**
- It is important that the data logged on shipboard be in a form which is most useful for post-cruise processing by scientific parties. Basically, there are two choices: To log raw data, or to log fully transformed and calibrated data. The best choice from our perspective is to log raw data, and include in the data file all relevant sensor information, so that the end-user can easily locate information for calibration and transformation of the data. By logging raw values, any post-cruise calibrations, sensor difficulties, etc., can be incorporated into post-cruise processing with a minimum of difficulty. This does not preclude logging of calibrated data in the logged file, but such logging should be in addition to the raw data.
- The **logged data form** should take into consideration the suite of sensors to be logged, and how they are identified. There are advantages to establishing a list of basic sensors which all ships should have: commonality of data sets between ships, and a basic data set which investigators can count on from ship to ship. This does not preclude a particular institution adding additional sensors to this basic suite. A suggested basic suite is listed below (Section 3.4).
    - **RECOMMENDATION 4: Raw, untransformed data should be logged in the data files. Transformed data In engineering or scientific units may be added to the file, but should be in addition to the primary raw data.**
    - **RECOMMENDATION 5: All data records should include information on time of collection, and averaging information if any. If averaging is used, at least the number of samples, maximum value, minimum value and mean should be recorded.**
- **3.3. Sensor Information**
- Following from recommendation 4, it is necessary to include information in the **data file**, or in the **descriptor file(s)** which will allow the user to 1) identify the sensor associated with a particular value, 2) identify the location of the sensor on the ship (where important), 3) identify the calibration constants and other calibration information for a particular sensor, 4) identify the calibration type or equation for conversion of the raw data, and 5) produce calibrated values from the raw data logged in the data file. The information in the *.cdl file will only allow the user to extract the actual raw values from the *.cdl data file. .cdl files do not include specific information about data values.
- Attachment 2 shows an example of the sensor and location identification system used on WECOMA. This is only an example, and does not imply adoption of this system by anyone else.
    - **RECOMMENDATION 6: Sensors should be identified within the data f lie or cross-referenced to the descriptor file so that the serial number and other relevant information for each sensor can be easily determined.**
    - **RECOMMENDATION 7: Sensor identification should include information on the location of the sensor on the ship. This is important when a calculated parameter (e.g., sea surface salinity) requires two measurements from the same location.**
    - **RECOMMENDATION 8: Sensor data should include calibration equation(s), constants and factors, date of last calibration, where calibrated, and any other calibration-related information which might assist with interpretation of the data.**
- **3.4 RECOMMENDATION 9: A Basic Shipboard Sensor Suite**
  The sensor suite listed below is intended as an initial starting point for discussion. It is based on limited knowledge of the sensors routinely available on UNOLS ships. There may be more than one source for each data item.
    - **3.4.1. Time**
      Date
      Time
      Sample Number

- **3.4.2. Navigation** GPS (possible multiple sources) SatNav (not universal)
  Loran-C (not universal)
  Ship Heading
  Ship Speed
- **3.4.3. Sea Surface (direct, flow-through and/or towed)** Temperature (may be more than 1)
  Conductivity (at least one associated with a temperature for calculation of salinity)
- **3.4.4. Meteorological**
  Wind Speed
  Wind Direction

  The following sensors can be basic, or part of an extended suite. Each additional meteorological sensor added also adds to the work load of the technicians, and to the complexity of the system.
  *Barometric Pressure
  *Humidity
  *Air Temperature

- **3.5. RECOMMENDATION 10: Identification of Standard Parameters**
  To aid in the interpretation of data in netCDF files, and to make the vessel-to-vessel transition as painless as possible for scientific users, it may be necessary to adopt a standard list of ascii identifiers for variables in the netCDF files. These identifiers would appear only in the netCDF file, as a variable associated with a data value.

  An example of such a variable identification scheme can be found in Attachment 4. This attachment describes the EPIC convention as registered with UNIDATA. This may serve as an example of a more rigorous convention, such as that required for standard parameters in this specification.
- **3.6. Onboard Data Distribution**
  One of the most frequent comments heard on WECOMA is: "Why can't I get [some data item] in real time like they have on the [some other ship] ?", and its converse 'I sure wish I'd had access to [some WECOMA data item] when I was on the [some other ship]." This experience is not unique to WECOMA, and points to the need for a common onboard data distribution specification for the fleet. As above, this is not meant to preclude any development beyond the basic specification, but as a basic service which can be improved on. The goal is to provide science parties with a "standard" service which can be counted on from cruise to cruise and ship to ship; that they win be able to go to any ship and find a plug which says "Real-time data," and have confidence that the data they got last time will be there this time.

  An example of the real-time, NMEA-formatted stream used on WECOMA is shown in attachment 3. This is only an example, and does not imply adoption of this system by anyone else.
  - **RECOMMENDATION 11: Onboard real-time data products should conform to the NMEA-0183 standard. The NMEA standard specifies electrical interface, data format, standard sentence formats, and allows addition of user-specific data sentences as "proprietary." The use of "proprietary", sentences should be kept to a minimum. [Where possible, transformed and "calibrated" numbers should be output in the onboard real-time stream, to maximize its usefulness.]**

4.0. Some Unresolved Issues (large and small):

- **4.1., Decomposition of combination data items?**
  Some data items found commonly on UNOLS vessels are actually combinations of several discrete data items. The prime example of this is GPS data acquired via serial interface from a receiver or

navigator. Each GPS "record' may contain multiple data items (for example, position, time, satellites tracked, DOPS, etc.). Should this be treated as a single data item, or a number of data items which are logged separately? If they are treated separately, which items should be included in the standard suite, and which optional?

- **4.2. Variables derived from other variables?**
  Certain data items are best dealt with as a value derived from two or more other variables. In the case of measured winds, it is better to average the wind vectors, rather than to average wind speed and direction separately. Which data items are to be accepted as combinations, and which separately?
- **4.3. Multiple Real-Time Streams allowed?**
  Many devices transmitting data to shipboard acquisition systems send data already formatted in NMEA-0183 format, notably navigation receivers. Should the specification allow multiple real-time streams that, in combination, conform to the specification? Must each vessel supply all data in a single stream, or is a combination of several navigation streams and one analog data stream acceptable?
- **4.4. How much farther do we want to go?**
  This specification is very general. Is there merit in establishing a more rigorous convention, such as that described in Attachment 4? Does this specification go too far?
- **4.5. How should the specification be implemented?**

Attachment I

R/V WECOMA Iogged Data File System *(SLOGGER)*

File Type	Format	Description	Name
Raw Data file	netCDF	Main data file	*MMDDHHmm.cdf*
cdf "list" file	ascii'C'	cdf descriptor file	*midas.cdl*
instrument ascii configuration		instrument ID and calibration info.	*inst.cfg*

SLOGGER data disks are 3.5",'1.44 MB PC-formatted floppy disks. Each disk has a

copy of *midas.cdl, inst.cfg,* and a single data file.

- Data file naming:
  - digits - month
  - digits - day
  - digits - hour
  - digits - minute

Time indicated is that for the first data record in the file. First data record after program is started begins on the minute.

Attachment 2

R/V WECOMA Instrument Type Codes	R/V WECOMA Instrument Location Codes
[rev 12/94]	[rev. 12/94]
AD Air Temp, Dew Point	A= met 1
AO Air Temp, OSU	B = met 2
AR Air Temp, Rotronics	C = met 3

AV Air Temp, Vaisala
AZ Air Temp, Other

BA Barometer, A.I.R.
BZ Barometer, Other

CS Cond, SeaBird
CZ Cond, Other

DE Down-welling, Shortwave (Eppley 8-48)
DL Down-welling, Longwave (PIR)
DP Down-welling, PAR
DS Down-welling, Shortwave (PSP)

FL Flow Rate
FR Fluorometer Range
FV Fluorometer Value

HR Humidity, Rotronics
HV Humidity, Vaisala
HZ Humidity, Other

PH Heave
PP Pitch somewhere/etc.
PR Roll
Example Code
RO Rain Gauge, ORG 100
RB Rain Gauge, ORG 700
RS Rain Gauge, Siphoning
RZ Rain Gauge, Other
SA Salinity, Computed
'M Water Temp, Seabird
'M Water Temp, Other
UL Upwelling, Longwave
US Upwelling, Shortwave
UZ Upwelling, Other
WH Wind Heading
WS Wind Speed
WV Wind Vector

D = Dog house (non met)
E = Engine room
F = Flow thru
G = Bow mast (non met)
H = Bow, in water
I = Trailed to port
J = Trailed to stbd
K = Trailed aft
L = Transducer well
M = Main Mast (platform)
N = future
O = future
P = Port (main mast)
Q = future
R = CTD package
S = Starboard (main mast)
T = Towed vehicle
U = future
V = future
W = future
X = future
Y = future
Z = No where/off line/in a box

TSL001 = SeaBird Temperature, Transducer well, Sensor ID#001.

Attachment 3

Slogger Raw Data NMEA 0183 Format

This page defines the format of the serial data stream sent by the WECOMA Slogger system. The data is in the National Marine Electronics Association's NMEA 0183 Standard for Interfacing Marine Electronic Navigational Devices. $P type proprietary sentences are used to transfer data which does not fall within the scope of approved NMEA 0183 sentences.

Transmission Specs

- Source: Slogger
- Format: ASCII NMEA 0183 data
- Serial Configuration:
    - 4800 Baud
    - No Parity
    - 8 Data Bits
    - 1 Stop Bit
- Data Interval: 1 minute

Record Format

The Slogger system acquires data from serial, analog, and frequency devices, computing a mean, minimum, and maximum sample value for each input over the period of a minute. It then broadcasts this data out a serial port in the herein defined format. Below is a sample slogger data stream. As can be seen, each data record is bound with the proprietary coded items $PSTA and $PEND. All data in these bounds was acquired during the same time interval.

$PSTA, <'record count'>	/* Start of Slogger record*/
$ZCZDA, <'data'>	/* sample time and date*/
$TRGLL, <'data'>	/* SatNav Data*/
$TRVTG, <'data'>	
$TRTRF, <'data'>	
$LCGLL, <'data'>	/* LORAN-C Data
$LCGTD, <'data'>	
$LCSBK, <'data'>	
$LCSCY, <'data'>	
$LCSNU, <'data'>	
$LCVTG, <'data'>	
$LCZLZ, <'data'>	
$GPGGA, <'data'>	/* GPS Data
$GPVTG, <'data'>	
$GPGLL, <'data'>	/* GPS 10 scan average
$GPGFF, <'data'>	/* GPS crystal clock
$WISTN, 01	/* Start of MET A */
<'data'>	/* Raw data from MET A Computer
$WISTN, 99	/* End of MET A */
$WISTN, 02	/* Start of MET B */
<'data'>	/* Raw data from MET B Computer /*
$WISTN, 99	/*End of MET B */

```
$WISTN, 03                      /* Start of MET C */
<'data'>                        /* Raw data from MET C Computer /*
$WISTN, 99                      /*End of MET C*/
$HEHDT, <'data'>                /* Gyro Data */
$VDVBW, <'data'>                /* Speed Log Data
$PRAWA2D, W011,<'data'>         /*Port wind speed data*/
$PRAWA2D, W012, <'data'>        /* Port wind heading data */
$PRAWA2D, W021, <'data'>        /* Stbd wind speed data */
$PRAWA2D, W022, <'data'>        /* Stbd wind heading data*/
$PRAWA2D, F011,<'data'>         /*Fluorometer Signal Data*/
$PRAWA2D, F012, <'data'>        /* Fluorometer Range Data*/
$PRAWFREQ, L011, <'data'>       /* SST Data */
$PRAWFREQ, T011,<'data'>        /*FlowThru SST Data*/
$PRAWFREQ, C011, <'data'>       /* Flow Thru Conductivity Data*/
$PEND, <'record number'>        /* End of Slogger Data Set */
```

Note: The output from the NET systems is expected to already be in NMEA-0183 format.

Attachment 4

Unidata Registration of PMEL-EPIC netCDF Conventions
NOAA Pacific Marine Environmental Laboratory
Seattle, WA

24 November, 1993

This document describes the netCDF implementation utilized by NOAA's Pacific marine Environmental Laboratory (PMEL) for use with the EPIC software package for oceanographic data. PMEL has developed the EPIC system library (EPS library), which is layered on top of the netCDF input/output library, to write netCDF files with the conventions described here. This library is not intended to provide the complete functionality which is available with the netCDF library, but rather to simplify production of a standardized implementation of netCDF for oceanographic data. It transparently provides support for some commonly used variations on the recommended standard implementation, and could be modified to provide support for others. With the use of this library, All application programs are independent of data file format, and formats other than the Unidata netCDF format can be supported by the addition of a set of "format dependent" routines to the "format dependent" layer of the library. At present, one other format is supported in addition to the netCDF format. Both EPIC and the EPS library, are available via anonymous ftp. Data files with the PMEL-EPIC conventions are compatible with the netCDF calculator function which PMEL uses in conjunction with the PPLUS graphics package. Simple examples including C and Fortran programs to read and write EPIC netCDF files, PPLUS scripts to read, plot, calculate and make animations from netCDF files, information about the interactions of PPLUS, the EPS library, EPIC and the TOGA-TAO Display software, and information about the commercially available PPLUS graphics package, are all included in the anonymous ftp directories. In addition, PMEL-EPIC netCDF files are compatible with the MATLAB netCDF interface (mexcdf) developed by USGS/WHOI, and available from USGS or from PMEL.

HOST:	csg.pmel.noaa.gov (192.68.161.12)
DIRECTORY:	anonymous/epic
	anonymous/eps
	anonymous/eps/examples
	anonymous/eps/pplus
	anonymous/tao/matlab
HOST:	crusty.er.usgs.gov (128.128.19.19)
DIRECTORY:	/pub/mexcdf

This document (from the Unidata /pub/netcdf/Conventions/PMEL-EPIC directory on unidata.ucar.edu) describes the Conventions common to all PMEL-EPIC style netCDF data files. Please see the accompanying documents describing Conventions specific to specific data types:

/pub/netcdf/Conventions/PMEL-EPIC/CTD/Conventions. CTD

/pub/netcdf/Conventions/PMEL-EPIC/Time_Series/Conventions.Time_series

==============

General Convention

==============

The EPIC system library is intended for reading and writing geophysical data, and assumes this data is represented by three spatial axes and one time axis. Each data file has dimensions, or axes, of longitude, latitude, depth and time. One or more of these axes may be collapsed to a single point, but four axes are always present. For example, data files containing oceanographic observational data, such as time series data or CTD profile data, may have latitude and longitude axes each consisting of a single point. Although we provide the ability to read netCDF files with fewer than four axes, we do not recommend this practice. our underlying philosophy is that all geophysical data is located by longitude, latitude, depth and time, and if this information is not included as an axis, it must be included elsewhere within the data file, if the file is to be self-describing.

==============

AXES

==============

EPIC axis conventions are described here. These conventions are compatible with the netCDF calculator function which PMEL uses in conjunction with the PPLUS scientific graphics package.

Geographic axes in a PMEL-EPIC netCDF file are, at present, described by a numeric variable code included as an attribute in the data file. There is a disk file named "epic.key" which contains all EPIC variable codes with other related information. The numeric variable code is a unique identifier for the variable or axis, and is described more fully in the section below on "VARIABLES". The use of the numeric variable code to define the axis is necessary for EPS library V2.1 and earlier, and for PPLUS Vl.2c and earlier. In future releases of both the EPS library and PPLUS, the use of an axis variable code may be replaced by the use of units from UDUNITS, and may no longer be required.

Longitude axis

Many longitude representations are possible, however, for comparability with the netCDF calculator function of PPLUS Vl.2c (and earlier versions), use of the West longitude convention is required. The West longitude convention, described in detail below, defines the numeric representation of the longitude axis in the netCDF file, with positive values for west longitudes. In future releases of the netCDF calculator function of PPLUS and also of the EPS library, both East and West longitude conventions will be supported. The recommended west longitude convention is described in the following paragraphs.

We recommend that users represent the longitude axis with the West longitude convention. This means that western longitudes are positive numbers, for example, 17O W is +170.0. Eastern longitudes can have either of two representations, both of which are supported by the EPS library, EPIC system application programs, and the PPLUS graphics system:

1. If the data being represented would best be described as continuous across the dateline, then east longitudes are written as (360.-long). EXAMPLE: 170E is written as 190.0.

2. If the data being represented would beat be described as continuous across the Greenwich meridian, then east longitudes are written as (-long). EXAMPLE: 170E is written as -170.

The units of the longitude axis in the EPIC system library routine call should be selected either from the unidata netCDF's udunits.dat file, or the EPIC system key file (see the EPS manual). The default spatial axis data type is real number. The recommended units are degree-west (epic variable code 501).

Latitude axis

The recommended latitude convention is for north latitudes to be represented by positive numbers (e.g., 10 N is +10.0), and south latitudes by negative numbers (e.g., 10S is -10.0). The units of the axis should be compatible with UDUNITS. The default spatial axis data type is real number. The recommended units are degree_north (epic variable code 500).

Depth axis

The depth axis should be given with the oceanographic convention of depth as a positive number, increasing downwards from the surface of the water towards the bottom of the ocean. The units of the axis should be compatible with UDUNITS. The default spatial axis data type is real number. Recommended units include dbar (pressure axis with epic variable code 1) or meters (depth axis with epic variable code 3).

Although the EPS library and PPLUS will support depth as a negative number, decreasing from the surface towards the bottom of the ocean, EPIC system application programs do not support this convention.

At present, only surface atmospheric data is being written to EPIC style netCDF data files. Conventions could be expanded to include this axis type, if desired, e.g., if upper air data is written in future, this axis could have the name "Height', with a corresponding epic variable code created to identify it uniquely.

Time axis

The EPIC system library routines return the time axis from a data file to the calling routine in the form of a two-integer array, in which the first integer is the "True Julian Day Number" with units of days, and the second integer is the number of milliseconds since 0000 GMT of the True Julian Day. The True Julian Day (eg, May 23, 1968 is 2,400,000), used by astronomers,, should not be confused with the "year-day" (eg, Feb. 2 is year-day 33). The "year-day" is frequently called julian day (incorrectly) by oceanographers and meteorologists. Our double-dimensioned integer time word (wordl=True Julian Day, word2=milliseconds since 0000 GMT of the True Julian Day) allows millisecond accuracy for time periods extending over centuries. There is a complete set of EPS routines for manipulation, calculation, and character string representation of this standard representation of time. The representation of the time axis within the data file can be of several types, for both read and write, including the UDUNITS standard, the two-integer array which is used internally by the EPS routines, and some other time representations which are supported for compatibility with other in-house software packages. Time axes can be written or read in either real or integer format. Note that, regardless of the format of the time axis in the netCDF data file itself, the values used internally by the EPIC system library will be the two-integer array.

==============

VARIABLES

==============

Each variable in an EPIC data file is described by a numeric variable code in the data file. There is a disk file named "epic.key" which contains the variable code and other related information about the variable. The numeric variable code is a unique identifier for the variable. Although the idea of a numeric code identifying the variable may seem at first unnecessary for netCDF files, there are several advantages to having a dependable variable identifier associated with the variables in the file. One reason is that it is unlikely that all the information about the variable will actually be included in the netCDF file (for example, exactly what algorithm was used to calculate salinity). Another reason is that it is difficult to build tools that will do complex tasks based on generic netCDF files. For example, salinity calculated by two different methods could be identified with two different numeric variable codes, but application software can be written to recognize both of these codes as a 'salinity" value for use in the calculation of density. Therefore, we recommend that the variable codes be used to identify variables in the data file. Although the EPS library will write netCDF files which omit the numeric code entirely, this omission will result in a lack of information available to EPIC or other application programs.

==============

ATTRIBUTES

==============

There are standard attributes created by the EPS library for EPS files, as described in the following table, and additional attributes may be defined by the user as desired.

==

Standard Attributes Automatically Written for EPS Files

==

global	CREATION-DATE	c	File creation date and time
Conventions		c	** Name of Conventions used by the file,(eg. PMEL-EPIC/CTD)
variable	name	c	variable name
long_name		c	variable long name
generic - name		c	variable generic name
FORTRAN - format		c	data Fortran format
units		c	variable unit
epic-code		i	variable code; write only when
it is defined in epic.key file			
axis	FORTRAN format	c	axis data Fortran format
units		c	axis unit
type		c	axis data type
epic_code		i	axis variable code; write only when
it is defined in epic.key file			
NOTES:		i is integer	
		c is character	

** The global attribute "Conventions' is not automatically written by EPS library V2.1, or earlier, but will be written automatically by later releases of EPS.

Similarly, standard global attributes are created for EPS data files containing specific types of oceanographic data (e.g., PMEL-EPIC/CTD and PMEL-EPIC/Time - Series). Additional global and variable attributes may be defined by the user as desired. The standard global attributes for each specific data type are described in the Conventions documents in those subdirectories of pub/netcdf/Conventions (eg, pub/netcdf/Conventions/PMEL-EPIC/CTD).