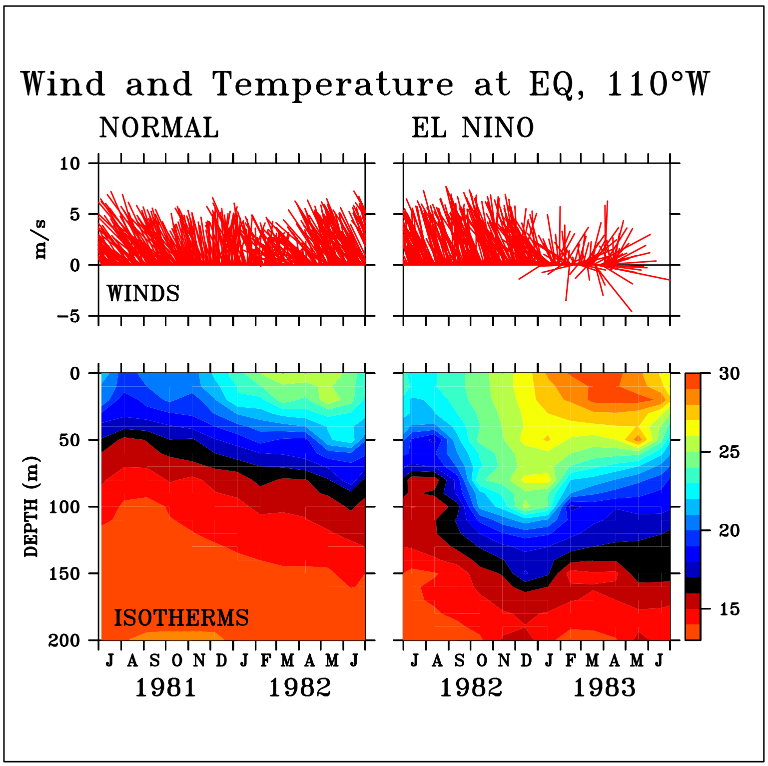
EPIC

A System for Management and Analysis of Oceanographic Time Series and Hydrographic Data



Nancy N. Soreide
Pacific Marine Environmental Laboratory
NOAA Building Number 3
7600 Sand Point Way NE
Seattle, WA 98115

Table of Contents

CHAPTER 1

System Overview

EPIC is a system for management, display and analysis of oceanographic hydrographic data and time series data. Supported data types include CTD, XBT, Bottle data, TOPS profiling current meter data, and time series of temperature, pressure, currents, winds, salinity relative humidity, and so on. The primary objectives in developing EPIC were to provide the user with interactive access to data sets, with graphical summarizations of data sets meeting the selection criteria, and with easy, interactive graphics display and data analysis utilities.

The EPIC system includes the following elements: the data sets, a data index (which includes descriptions of the data sets), graphics display and analysis routines, documentation, and maintenance utilities. The data sets and the data index are on-line. Data selection is made interactively by specifying data type, geographic location, depth and time in answer to questions. Plots and listings can be made to summarize the space-time characteristics of the selected data. The data selection process need not be repeated when plotting and analysis is performed, and unwanted data sets are easily eliminated from the selected data. Each data selection can be identified by a name of the users choice. This process is outlined in detail in the Beginners Guide Chapter (page 11).

The types of data in EPIC at present are listed below with their identifying symbols.

TABLE 1. **Pressure-indexed profiles.**

Name	Symbol	Description
CTD	C	Conductivity/Temperature/Depth measurements
XBT	X	Expendable BathyThermograph Measurements
BOT	B	Bottle data

TABLE 1. Pressure-indexed profiles.

Name	Symbol	Description
AMETEK	AMK	Acoustic Doppler Current Measurements
TOPS	TOPS	TOPS free-falling profiling current meter

TABLE 2. Time series.

Name	Symbol	Description
Temperature	T	Thermistors, etc. on moored buoys
Pressure	P	Bottom pressure recorders
Currents	U	Current meters on moored buoys
Air Temperature	A	Temperature recorders on moored buoys
Salinity	S	Salinity from moored buoys
Rel. Humidity	H	Relative humidity from moored buoys
Drifters	D	Drifting buoy data
Moored ADCP	MA	Moored Acoustic Doppler Current Profile data
Wind	W	Wind recorders on moored buoys

EPIC includes a suite of programs for display and analysis of both time series and hydrographic CTD-type data. Once the data selection has been made, any plotting or analysis program can be started by typing the program name in response to the operating system prompt. Data display and analysis can be performed interactively, although batch mode operation is available when large numbers of data sets have been selected.

The major EPIC graphics package is PPLUS. It is documented in the PPLUS manuals, and its relation to EPIC is described in the EPIC and PPLUS Chapter (page 29) of this manual. Some older EPIC programs use the DISSPLA graphics package, but use of this package is being phased out.

The EPIC system is described in this manual and in the on-line help. In general, the manual documents system elements which are not frequently updated, such as the data file formats and the relational data base structure. Since new programs and procedures are continuously being included, full and up to date documentation for these is always found in the on-line HELP.

The EPIC data sets are documented in two ways. Each data file includes descriptive header or attribute information about the data. The data index also contains the same header or attribute information for each data set. When the data index is searched during the data selection process, the user is left with a summary of information about each data file selected in the form of the EPIC file. The data index is presently based on RIM, which is a relational data base described in the Data Management With Rim Chapter (page 55). It is possible to extract information about the data sets from the data index by using RIM directly on the data base tables (see the Data Management With Rim Chapter).

To use EPIC, first establish definitions as indicated in the Getting Started chapter (page 9). Then see the Beginners Guide chapter (page 11) for an introduction to the data selection process and some commonly used programs.

CHAPTER 2

Operating Systems: VMS And Unix

Overview

The EPIC system was originally developed for the VAX/VMS operating system. In this environment, EPIC is a complete system including a data selection module and a suite of over 100 graphics display and analysis programs. System elements for data selection, data display, and data analysis function as independent modules. The system is well documented, with a user manual and extensive on-line help. The system is written in Fortran and C, and all elements are supported for VAX/VMS computers.

EPIC is rapidly migrating from the VAX/VMS environment to a system supported for both VMS and Unix. Many EPIC system elements are already operating system independent, and all EPIC software development is now being done on Unix and downloaded to VMS. Source code will be maintained on Unix, with pre-processor directives to allow compilation under either operating system. Programs are written in ANSI Standard Fortran or C. The major EPIC graphics package is PPLUS, which is supported for both Unix and VMS. In order to minimize the effect of data file format, support for the Unidata netCDF format has been incorporated into EPIC. This portable data format was developed by Unidata, at UCAR, under NSF funding, and data files in this format are machine readable by VMS, MAC, PC and Unix computers. Any netCDF file can be copied across a network and used on any of these hardware platforms.

Advances in ocean observing systems, such as large arrays of moored buoys, or acoustic doppler current meters, sample the environment more densely than was previously practical, and necessitate expanding EPIC beyond the original one-dimensional data set capability. Use of the Unidata netCDF format provides EPIC with a multi-dimension file format.

The Epic System Library (Eps Library)

The EPIC system has a new input/output library for reading and writing data files. This new EPIC System library (EPS library) has been designed to permit oceanographic data to be stored in either Classic EPIC format or netCDF format data files. Application programs performing input/output to data files, with the EPS library, call a “format independent layer” of subroutines, which are generic routines to open and close data files, to read and write attributes or header information, and to read and write data. The generic routines are independent of the physical format of the data on the disk. There is a “format dependent layer” for the original, one-dimensional, binary Classic EPIC file format and another “format dependent layer” for the more recent, multi-dimensional netCDF file format. These formats are described more fully in the chapter titled Data Files (page 35). The EPS library automatically determines the actual format of the data file when the file is opened for read. If the file is opened for write, the application program calls the EPS open routine with an argument to indicate which format is to be used when the data file is written. Classic EPIC files created on the VAX (as binary, sequential, machine-dependent VMS files) can be read from Unix with the EPS library. By providing this compatibility and by providing transparent support for both data formats, the EPS library facilitates a smooth transition from the VAX to Unix, and from Classic EPIC format to netCDF format.

With the new EPS Library support for netCDF format, EPIC system data files are machine independent. In addition, the EPS library extends the useful lifetime of the application programs by eliminating program dependence on data file format, which has been a major cause of difficulty in previous computer conversions. The EPS library is designed so that it could easily support additional data formats by the addition of a “format dependent layer” (consisting of eight C subroutines) for each additional data format.

The EPS library is supported for VMS and for Unix, and the EPS routines are callable from Fortran and from C. All new EPIC software utilizes the EPS data file input/output library, and older software is being converted to use it as well.

With regard to the netCDF format, the EPS Library design simplifies production of a standardized implementation of netCDF for oceanographic data. This standardized implementation is registered with Unidata (the netCDF developers) as the PMEL-EPIC Conventions. We anticipate that Conventions documents will be registered in the near future for additional data types, such as drifting buoys and ADCP data. Adoption of the PMEL-EPIC Conventions and use of the EPS library by other oceanographic institutions would facilitate sharing of oceanographic analysis software between institutions. The PMEL-EPIC Conventions documents are available from the Unidata anonymous ftp account.

HOST: ftp.unidata.ucar.edu
DIRECTORY: /pub/netcdf/Conventions
PMEL-EPIC/
PMEL-EPIC/CTD
PMEL-EPIC/Time_Series

PMEL also provides EPIC, the EPS library, and related information and examples on anonymous ftp (described in the last section of this chapter).

The PPLUS Scientific Graphics Package

The primary graphics package for EPIC is PPLUS, which is supported for VMS and Unix workstations. PPLUS is an interactive, command driven, scientific graphics package which includes features such as Mercator projection, Polar Stereographic projection, color or gray scale area-fill contour plotting, and support for many devices: X-windows, PostScript, Tektronix, and others. This powerful and flexible package recognizes EPIC data formats, and it can extract axis labels and graph titles from the data files. Most EPIC programs make plots for the user automatically, and leave the user with a PPLUS command file or script, which the user can edit in order to customize a plot, or combine several plots into a composite. With PPLUS, plots of non-EPIC data in almost any disk file format can be included with EPIC data on a plot. Plots are of publication quality. Animations can be made easily for Unix by generating a PPLUS plot for each frame, transforming the PPLUS metacode files into HDF format with the PPLUS m2hdf filter, and then displaying the resulting HDF bit maps as an animation with the XDataSlice utility, which is freely available on Internet from the National Center for Supercomputing Applications. This process is performed for EPIC users with the EPIC TIMZSEC animation routine.

Algebraic Calculations With Epic Data

Algebraic manipulations of EPIC data fields is possible with the PPLUS netCDF calculator, which is available interactively within the PPLUS graphics package as the “e” command (using nccalc). It provides for interactive algebraic manipulation of data fields in EPIC data system files (Unidata netCDF format as well as the original EPIC format). With this, multi-dimensional data fields can be summed to calculate a mean, and then anomalies from this mean can also be calculated and displayed. The results of the calculations can be written out as a new EPIC system data file. There is a simple re-grid function to allow calculations with fields on different grids. Data sets, such as time series, can be extracted easily from the multi-dimensional data sets with the “e” command. The PPLUS “e” command also includes an interactive data editor for one-dimensional data fields, which allows data to be edited interactively, and the edited results written out to an EPIC system data file.

Epic Programs For Vax And Unix

All EPIC programs described in the Beginners Guide chapter (page 11) are available for both VMS and Unix. Almost all the general purpose routines and graphics display programs (approximately one fourth of the entire body of EPIC programs) are available for Unix at this writing, and an increasing number of utility and analysis programs is becoming available for both operating systems.

The EPIC system elements whose migration to Unix is not complete include the on-line HELP, the data selection module, and the remainder of the analysis programs, and this work is presently underway.

Software Available On Anonymous Ftp

Both EPIC and the EPS library, well documented with manuals and a suite of sample Fortran and C programs, 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 and EPIC, as well as information about the commercially available PPLUS graphics package, are all included in PMEL's 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

CHAPTER 3 Getting Started

To Get A Copy Of This Manual

To get a copy of this manual, type the following lines on your terminal in response to the VAX operating system prompt:

```
$ @DISK1:[OC.SYMBOLS]EPIC
$ EPIC_MANUALS
```

The manual will be printed on the printer. It is also available from PMEL's anonymous ftp account, as described in the previous chapter.

Required Definitions

VAX/VMS Symbol Definitions

EPIC requires several assignments and definitions to execute under VMS. The following should be included in your LOGIN.COM file prior to running EPIC:

```
$ @DISK1:[OC.SYMBOLS]EPIC
$ @DISK1:[OC.SYMBOLS]PLOT5.COM
$ GRAPHTERM :== xxxx ! for non X-windowing terminals
```

```
$ PPLX node_name ! for X-windowing terminals
```

where xxxx describes your graphics terminal and has the following allowed values:

```
VT240
GVT+
ZENITH
TEK4010
MAC
TEK41XX
TEK4105
TAB
```

Executing EPIC.COM defines the necessary EPIC symbols. Executing PLOT5.COM defines symbols needed by PPLUS, which is the major EPIC graphics package. In order to provide automatic entry and exit into and out of graphics mode you should use the GRAPHTERM that corresponds to your terminal. If your terminal is a TEK4010 or TEK4014 compatible, but not one of the above, then place your terminal into graphics mode before plotting and use GRAPHTERM :== TEK4010. For more help with PPLUS, see

HELP PPLUS and the PPLUS manual.

Unix Environment Variables And Paths

Use of EPIC on Unix requires establishing the proper environment variables and making the appropriate inclusions in your PATH.

If you are on a non X-windowing terminal, you must include the following in your .login file:

```
setenv GRAPHTERM string      non X-windowing terminals
setenv DISPLAY node_name:0   X-windowing terminals
```

where “string” describes your graphics terminal and can have the following values:

```
VT240
GVT+
ZENITH
TEK4010
MAC
TEK41XX
TEK4105
TAB
```

The PPLUS and EPICHOME directories directories must be included in your PATH. These path designations depend on the directory locations selected when PPLUS and EPIC were installed on your computer system. See your EPIC system manager for details.

CHAPTER 4

Beginners Guide

To use EPIC, you must establish appropriate definitions as described in the Getting Started Chapter (page 9). The following sections describe the process of selecting data sets and running some frequently used EPIC programs.

Data Selection

Using EPIC starts with the data selection process. Once you've made the necessary additions to your LOGIN.COM and run your LOGIN.COM, you can type EPIC after the operating system prompt. The program will prompt you for data selection criteria such as data type, latitude, longitude, etc. EPIC accepts input in either upper or lower case. The format of responses to EPIC questions is flexible. Here are some examples:

Latitudes 10 30.0N or 10.5N or +10.5 or 10.5 NOTE - South latitudes are negative

Longitudes 95 00W or 95 W or 95W or -95 or -95.0 NOTE - West longitudes are negative

Date/times 01 DEC 82 0000 or 1 12 82 or 1 dec 82

For equatorial data, specifying a starting latitude/longitude without specifying an ending one will yield data taken within 0.24 degrees of the one specified for CTD-type data and within 0.75 degrees for time series data.

A starting time specified without an ending one will result in a search for data from that time to that time + 24 hours. For time series data, any data sets with data during the specified time range will be selected, i.e., selected data sets may have data during only part of the time range specified.

For CTD data, the “required depth range” assures that all selected data sets have data within the specified range. For example, a required range of 0 to 1000 meters for CTD data will exclude CTD casts which only go as deep as 500 meters.

When the EPIC data selection program prompts for a “filename for output”, enter a VAX filename without an extension (e.g., enter “95W” or “CTD110W”). EPIC will force the extension to be .DAT. The default name for this file is EPIC.DAT. It is called “the EPIC file”.

The EPIC file is a summary listing of data files which met your selection criteria. It doesn’t include any data; it’s just a list of data file names with some summary information about the data files. All the plotting or analysis programs use this file (or one like it) to locate the data sets you’ve selected. You can look at the EPIC file with any text editor, list it on the terminal, or print it on a printer. If you want to delete some of the selected data sets, you can use a text editor to delete those lines of the EPIC file. The programs will use only the data sets listed on the EPIC file.

You do not always need to go through the EPIC program to select the data sets you want. Another EPIC user may have given you an EPIC file, or just a list of data file names, or you may be familiar with the EPIC data directory structure and be able to obtain a list of the data files you want without the EPIC program. The EPIC programs will use a list of data file names just as easily as an EPIC file to locate your selected data sets. Just be sure that the list has one data file name per line with the file name starting in column 1. A list like this is called a “pointer file” in all the EPIC documentation. When an EPIC program prompts you for the name of the EPIC file, give it the name of your list of data files. In fact, any EPIC program which generates output data files will also generate an output pointer file. This output pointer file is simply a list of the output data file names.

At this time, there is no EPIC data selection program for Unix, and users must create lists of data files (pointer files), in place of the EPIC file which is generated by the data selection program on the VAX.

Epic File Details

EPIC File For Hydrographic Data

The EPIC file generated by the data selection program for pressure-indexed profile data such as CTD data includes the selection criteria which were used, the data type selected, cruise identification, cast number, latitude, longitude, date, initial depth, final depth, and file specifications (disk, directory, and filename).

EPIC File For Time Series Data

The EPIC file generated by the data selection program for time series data includes the selection criteria which were used, the data type selected, latitude, longitude, depth, start time, end time, time increment in minutes, position consistency flag, depth consistency flag, and file specifications (disk, directory, and filename).

The position consistency flag is equal to "OK" for position- consistent data or "??" for data with some inconsistency in position (e.g., a composite series of equatorial data from 110W joined with equatorial data from 108W).

The depth consistency flag is equal to "OK" for depth-consistent data or "??" for data with some inconsistency in depth (e.g., a composite series of data at 15 m joined with data at 10 m depth).

Composite time series have "piece headers" containing specifics of each piece. Use program HEADER to look at the data file headers.

Programs

All EPIC programs are started by typing the program name after the operating system prompt. The programs described here can be used on data in the EPIC data base directories, on data produced by EPIC analysis programs, or any other data in an EPIC format. The following sections describe some of the most frequently used EPIC programs. The full suite of EPIC programs is summarized in the Program Synopsis chapter (page 17), and all programs are fully described in the on-line HELP. A complete list of all available programs is available on line with the EPIC_FIND utility, which also supports a key word search of a synopsis of all EPIC programs. The programs described below are available for VMS and for Unix.

SUMMARY

This program makes a summary listing of header information from any EPIC system data. There is are options to list data from the file in addition to header or attribute information. The listing can be made on your terminal or put into a disk file. After the operating system prompt, type SUMMARY and answer the questions. As with most programs, the first question will ask for the name of the EPIC file.

DATALST

This program makes a listing of any EPIC system data file in a format easily input to spreadsheet programs. The listing will be put into a disk file named after the input EPIC file. After the operating system prompt, type

DATALST and answer the questions. As with most programs, the first question will ask for the name of the EPIC file.

LOCATER

This program plots the locations of EPIC data files on a map. Program options include mercator projection and inclusion of coastlines. Data locations can be annotated with cast number for CTD data, or mooring and experiment information for time series data. This program uses the PPLUS graphics package. (This program is also known as EPICPLT.)

CTDPLT

This program makes a plot of CTD, XBT, TOPS or Bottle profile data. The plot can be made interactively on your terminal or can be routed directly to one of the plotters. To start the program, type CTDPLT after the operating system prompt. Then answer the questions. You'll be asked for the name of the EPIC file and then for plotting parameters. If you aren't sure what variables are included on the data sets you've selected, you can use DATALST or SUMMARY to look at the data.

Program CTDPLT uses the PPLUS graphics package. You need never learn to use PPLUS directly. However, CTDPLT leaves you with a PPLUS command file which you can use to modify the plot or to combine it with other plots, if you choose. See the EPIC And PPLUS Chapter (page 29) for more information.

CONCTD

CONCTD makes a contour plot of CTD, XBT, TOPS or Bottle profile data with depth on the y-axis and either latitude or longitude on the x-axis. You can also plot with the x-axis as distance along the ship track line. You can make the plot interactively or route it to one of the plotters. Start the program by typing CONCTD after the operating system prompt and answer the questions.

Program CONCTD uses the PPLUS graphics package. You need never learn to use PPLUS directly. However, CONTOUR leaves you with a PPLUS command file which you can use to modify the plot or to combine it with other plots if you choose. See the EPIC And PPLUS Chapter (page 29) for more information.

TIMPLT

This program makes a plot of time series data. The plot can be made interactively on your terminal or can be routed directly to one of the plotters. To start the program, type TIMPLT after the operating system prompt. Then answer the questions. You'll be asked for the name of the EPIC file and then for plotting parameters. If you aren't sure what variables are included on the data sets you've selected, you can use TIMLST to look at the data.

Program TIMPLT uses the PPLUS graphics package. You need never learn to use PPLUS directly. However, TIMPLT leaves you with a PPLUS command file which you can use to modify the plot or to combine it with other plots if you choose. See the EPIC And PPLUS Chapter (page 29) for more information. (Conversion of this program to Unix is underway, but may not be completed as you read this. All other programs in this section are available for Unix.)

RDSTATS

RDSTATS is a basic statistics package. Its output (file RDSTATS.INFO) lists the mean, minimum, maximum and variance of each variable in the file, as well as the vector mean speed and direction and principal axis if the file contains a vector. The user can also choose to rotate the vectors before calculating the statistics and whether (and for which variables) to compute RMS errors. The RMS calculation involves the integral time scale. The limits for the vector mean speed are simply calculated from the minimum and maximum values obtained with (UAVG+-URMS) and (VAVG+-VRMS). Start the program by typing RDSTATS after the operating system prompt and answer the questions.

CONTIME

CONTIME makes a contour plot of time series data with time on the x-axis and either latitude or longitude on the y-axis. You can make the plot interactively or route it to one of the plotters. Start the program by typing CONTIME after the operating system prompt and answer the questions.

Program CONTIME uses the PPLUS graphics package. You need never learn to use PPLUS directly. However, CONTIME leaves you with a PPLUS command file which you can use to modify the plot or to combine it with other plots if you choose. See the EPIC And PPLUS Chapter (page 29) for more information.

CHAPTER 5 Program Synopsis

This is intended as a brief overview of the EPIC programs. All programs are fully described in the on-line HELP. As new programs are developed, they will be included in the on-line HELP documentation, which is always the place to find the most recent information. To use a program, type the program name after the operating system prompt and then answer the questions. See the Beginners Guide Chapter (page 11) for examples. This list includes all the EPIC programs supported for VAX/VMS. Not all of these programs are available for Unix systems.

Root

EPIC Data Selection Program	{EPIC}

Ctd_avg

Mean CTD profile	{CTDAVG}
Mean TS profile	{TSAVG}
Subsamples scans, variables	{CTDEDIT}
Re-grids data in pressure	{CTDGRID}
Bins data by pressure (vertical average)	{CTDBIN}

Ctd_ocean

Brunt Vaisala frequency	{CTDBVF}
Isostherm Depths	{CTDISO}
Calculates desired variables	{CTDVAR}
Geostrophic velocity	{CTDVEL}
Gradients dy/dx OR dx/dy from CTD data	{CTDGRAD}
Surface dynamic height for XBT data	{DYNCTD}
Dynamic height ref to level of no motion	{DYNREL}

Ctd_plot

CTD x-y plots (optional overplotting)	{CTDPLT}
General contouring (y-axis is pressure)	{CONCTD}
Contours on pressure surface (x=lon,y=lat)	{CONSURF}
Original contour plots (pplus gridding routine)	{CONTOUR}
Nutrient/Bottle contour plots (one variable)	{CONNUT}
CTD locations with cast numbers (see also LOCATER)	{CTDLOC}

Ctd_util

Lists data	{STDLST}
Windows, replaces bad data	{CTDINTERP}
Re-computes selected variables	{CTDRECAL}
Integrates over selected variable	{CTDINT}
EBCDIC tape + documentation	{CTDOUT}
EBCDIC tape + documentation	{NODCOUT}
Applies polynomial	{CTDPOLY}
Move CTD data into MacIntosh OceanAtlas Browser	{CTDATLAS}
Makes time series of selected variable	{CTDTIM}
Velocity zeroed at selected depth	{ZEROVEL}
Cruise summary from CTDs on EPIC/pointer file	{CRUSUM}
CTD mixed layer depth	{CTDMLD}
Computes statistics over selected depth range	{CTDSTAT}

Time_arith

Adds/multiplies time series after linear xform	{ADDTIM}
Applies polynomial	{APOLY}
Square root of time series	{TIMSQRT}

Time_dsf

Must run this before using SPCTRA or COVSPC	{WORKSZ}
Coefficients for spectra, coherences, freq dom EOFs	{SPCTRA}
Spectra or coherences from spectral coefficients	{SPCPLT}
Frequency domain EOFs from spectral coefficients	{SPCEIG}
Time domain EOFs from spectral coefficients	{CVREIG}
Spectral coefficients for correlations	{COVSPC}
Correlations and lags from spectral coefficients	{COVAR}

Time_filt

General filter program - replaces FILTIME	{ALLFILTS}
Low pass, band pass filters - Replaced by ALLFILTS	{FILTIME}
Hanning filter - Replaced by ALLFILTS	{HANNING}
Running mean	{RMEAN}

Time_int

Integrates desired variable over depth	{DEPINT}
Integrates desired variable over latitude	{POSINT}

Time_join

Combine variables from 2 data files	{COMBINE}
Extracts a single variable from a data file	{TIMEDIT}
Joins series at matching depths	{JOIN}
Joins series at matching depths, 0-fills gaps	{JOINZ}
Joins series at matching depths,fills gaps w/ 1.E35	{JOIN_FILL}
Joins retaining desired segment	{INSERT}
Splits files into equal length segments	{SPLIT}

Time_manip

Averages (daily, weekly, block, etc)	{TIMAV}
Removes mean, detrends or first-differences data	{DMEAN}
Calculates mean over depth	{TMEAN}
Zeros velocity at desired depth	{TIMZERO}
Bridges gaps in time series	{BRIDGE}
Puts time series on desired time grid	{TIMGRID}
Adjusts date of input time series data	{TIMSLIDE}
Zeros all values less than specified value	{KEEP_ABOVE}
Zeros all values greater than specified value	{KEEP_BELOW}
Extracts easterly winds	{EASTERLY}
Extracts westerly winds	{WESTERLY}
Interpolates to new depths	{NEWDEP}
Calculates monthly climatology	{CLIMON}
Principal axis calculation or vector rotation	{PRINAX}

Time_ocean

Dynamic Ht from temperature time series	{DYNTIME}
Surface geostrophic current from dynamic ht	{GVEL}
Heat content from mooring temperatures	{HEAT}
Isotherm depths from temperature time series	{ISOTHERM}
Vertical displacement of temperature	{TIMDISP}
Mixed layer depth from mooring temperatures	{MIXDEP}
Zonal or meridional transport from currents	{TRANSPORT}

Wind speed from zonal and meridional velocity	{WNDSPD}
Wind u,v from speed, direction	{WNDUV}
True or pseudo wind stress	{WNDSTR}

Time_plot

Time series x-y plots	{TIMPLT}
Contour plots	{CONTIM}
Yearly climatology from ATLAS files	{CLIMPLT}
Time line summary plots from an EPIC file	{TIMSUM}
Time line summary plots (used by TIMSUM)	{DATSUM}
Stacks x-y time series plots, x=time	{TIMSTACK}
Progressive vector diagram of u/v time series	{TIMPVD}
Histogram plot of time series data, with statistics	{TIMHIST}
Animations: contour plots of z-sections	{TIMZSEC}

Time_spect

Calculate and plot coherences	{COHERE}
Time domain EOF	{EOF}
Calculate and plot spectra	{SPECTRA}
Correlations and auto-correlations	{CORREL}
Fits to Y = C0 + C * COS (W*T - PHI).	{COSPHA}
Fits to Y = C0 + C1*T + C*COS (W*T - PHI)	{COSLIN}

Time_stats

Daily, monthly, etc mean and std dev	{STATS}
For plotting output of STATS program	{AVGSIG}
Mean, std dev over all times, first variable	{VAR}
Mean, std dev over all times, all variables	{VARALL}
Mean, min, max, variance, RMS errors	{RDSTATS}
RMS, mean, etc for pairs of time series	{RMSTIM}

Time_tides

HARMONIC - 29-day harmonic tidal analysis	{HARMONIC}
RESPONSE - Response Analysis of tidal signals	{RESPONSE}

Time_util

ASCII data into EPIC format	{EPICIN}
Finds gaps, calculates some summary statistics	{TIMGAP}
EPIC/pointer file into mooring pointer files	{MOORING}
Remake pointer file with unique data file names	{UNIQUE}
Mooring line drag	{DRAG}
EPIC data into DSF format	{EPIC_DSF}

CHAPTER 6

Help Utilities

Several help utilities are available for EPIC on VMS. Some of these have been implemented for Unix, and more will become available in future.

Epic_manuals

On VMS, you can get copies of any of the EPIC manuals by typing “EPIC_MANUALS” after the VAX prompt. There are manuals for the system itself, for the data manipulation routines, for the EPSLIB I/O library, and for the PPLUS graphics package. The EPIC manual contains an introduction to EPIC system use, an overview of the system, synopses of programs and subroutines, detailed descriptions of data file formats, descriptions of the relationship between EPIC and PPLUS (the major graphics package), and a description of data management with RIM (a relational data base).

EPIC is documented in the manual, in the on-line HELP and in TEPIC. In general, the paper manual documents system elements which are not frequently updated. Since new programs and procedures are continuously being added, full and up to date documentation for these will be found in the on-line HELP and in TEPIC (where TEPIC is available).

On Unix systems, contact your system manager to obtain copies of the EPIC system manuals.

VMS Help Utilities

To use EPIC on VMS, you need to establish some VAX/VMS symbols (see the VAX on-line help with $ HELP EPIC VAX_SYMBOLS). Once these symbols have been established, you can get a copy of any of the EPIC manuals by typing “EPIC_MANUALS” after the VAX prompt. There are manuals for the system itself, for the data manipulation routines, for the EPSLIB I/O library, and for the PPLUS graphics package. The EPIC manual contains an introduction to the system, and a synopsis of programs. Detailed and up-to-date descriptions of programs are contained in the on-line VAX HELP. At PMEL, the TEPIC utility provides access to the EPIC programs and help from a menu driven system. TEPIC depends on NASA’s TAE utility, and may not be available outside PMEL. You can locate programs by (1) reading the on-line help, (2) using interactive utility EPIC_FIND to locate programs by a string-search or a key-word search, or (3) using TEPIC (TEPIC may not be available outside PMEL), which lets you locate EPIC programs and help by category. You can run EPIC programs in VAX command mode by entering the program name after the VAX prompt, as a VAX comand, and, if TEPIC is available, from within the TEPIC menu-driven system.

To start the data selection process without TEPIC, type EPIC in response to the VAX’s prompt. You will end with an “EPIC file” which summarizes the data selected. It can be listed, PRINTed or EDITed, and locates data files for all programs. Plotting and analysis programs are started by typing the program name in response to the VAX’s prompt. If TEPIC is available, you can use all the EPIC programs from TEPIC, as described in the following sections.

VMS TEPIC Menu-driven Access To Programs And Help

TEPIC depends on NASA’s TAE utility, and will not be available unless TAE is installed.

EPIC programs are available through a menu-driven interface which lets users locate programs, look at help for the programs, and run the programs. This interface is based on the NASA product TAE (Transportable Application Executive). Type EPIC_INFO to get a listing of available EPIC programs and for information on EPIC via the menu-driven TAE. To use EPIC with the menu-driven TAE interface, type TEPIC and then enter M to go into Menu mode.

Programs are still being documented in the on-line help, but users will probably find it easier to locate programs and help by using TEPIC. Programs can be run through TEPIC (just type RUN to execute), or can be run by typing the program name after the VAX prompt.

Another new way to locate the program you want is to use EPIC_FIND. It does a key word search. For example:

```
$ EPIC_FIND plot
```

locates all programs which plot.

Introduction To TEPIC

TEPIC is EPIC with all it's programs available through a menu interface called TAE (Transportable Application Executive). All the same programs are available with the TEPIC system. You can run the programs in either of two ways: by typing the program name after the VAX prompt or by running it in TEPIC.

Type TEPIC to enter the VAX menu-driven system. This puts you into TAE with all the EPIC programs. You will be in TAE Command Mode. Type M to enter Menu mode. Then you can select any entry number (just type the number) or ask for help on that entry (type "HELP entrynumber").

Running A Program From TEPIC

To run an EPIC program, select the program from the menu by typing the number of the selection. Type "RUN" to execute the program.

TEPIC Modes

From TEPIC menu mode:

C Gets you into command mode
B Gets you previous menu
T Gets you the top menu
[menu] Selection is another menu
{prog} Selection is an "Classic EPIC" program - answer the questions
(prog) Selection is a "TAE EPIC" program - use TAE tutor mode to set parmeters

From TEPIC command mode:

M Gets you menu mode
HELP prog Help on that program
TUTOR program Tutor mode (parameter setting mode) on program Classic EPIC program - type RUN and answer questions TAE EPIC program - set parameters or RESTORE old parameters
prog Puts you into execution on a program
prog |restore=xxx.par, run=batch |parm1=val1, parm2=val2... Submits batch run, restoring parameter set "xxx.par"
SHOW-BATCH Look at TAE batch jobs - note JOBID number and QUEUE name
ABORT-BATCH JOBID=jobidnumber QUEUE=queuename Abort a TAE batch job

From TEPIC tutor mode:

To abort within TEPIC while a program is running:

Abort a program :

<CTRL> <C> Stops program execution, then type:
ABORT to abort the program
CONTINUE to continue execution
EXIT to exit

Abort a batch job:

1. Get into command mode (Type C if you are in Menu Mode)
2. SHOW-BATCH - get job id and queue name
3. ABORT-BATCH JOBID=jobid QUEUE=queuename
4. Type M if you want menu mode, type EXIT or LOGOFF to exit

To exit from TEPIC:

EXIT (or just E) to exit
LOGOFF (or just L) to log out of TAE and get back to the VAX prompt
DCL to return to the VAX temporarily (on the VAX, <CTRL><Z> to return to TAE)

On-line within TEPIC:

HELP on selections on the left side of the screen (menu mode)

Creating An EPIC Pointer File

Every program will prompt you for the name of the EPIC file. You should answer this with the name of an ASCII disk file containing a list of the names of the data files. This will provided for you by either the EPIC data selection program or by an EPIC program which computes something (like a time series average).

If you've run the EPIC data selection program, that will leave you with an EPIC file which includes a list of selected data files (along with some information about the data files). If you have run a program, it will leave you with an EPIC pointer file, which is just a disk file containing the names of the data files the program created.

If the data selection program is not available on your system (the data selection program is still under development for Unix), you will have to create an ASCII disk file containing the names of the data files for input to the program as a pointer file. You can type a list using an editor, or you can get a list of data files from a directory with operating system commands. In the following examples, the pointer file name is myfile.dat and it will contain a list of CTD data files.

For VMS:

```
$ directory/nohead/notrailer/col=1/output=myfile.dat dh:[tw284.ctd]*.ctd
```

For Unix:

```
% ls -1 $cwd/home/tw284/ctd/*.ctd > ~jsmith/myfile.dat
```

CHAPTER 7 EPIC and PPLUS

PPLUS recognizes all EPIC data formats supported by the EPS library. It can locate selected variables in the data files and extract graph titles and axis labels from the data file headers. This section describes several aspects of the PPLUS/EPIC relationship.

Ctd-type Data

PPLUS recognizes EPIC CTD-type data and it will automatically extract the axis labels and the graph title from the CTD data file headers.

EPIC programs, such as CTDPLT, will plot CTD-type data using PPLUS and leave you with the PPLUS command file which made the plot. Program CTDPLT will overplot data from several CTD casts on a single page, and will also make overplots of several variables from a single CTD cast with a separate axis for each variable. The PPLUS script or command file which the program creates will serve as an example of this type of plotting. Some simple examples of PPLUS calculations and graphical displays with CTD data follow here:

Examples

Example 1:

Here is a simple PPLUS command file to make a T-S plot of CTD data where the EPIC file is named “ctd.dat”.

```
e openr “ctd.dat”
e nextr
e t1=temp[z=*]        ! extract temperature into variable
e s1=sal[z=*]         ! extract salinity
e line (s1,t1)        ! load salt, temp into PPLUS plot buffer
plot
```

Example 2:

Here is a simple example showing how to over plot Pressure versus Temperature from two EPIC CTD files:

```
e openr “ctd.dat”
e nextr
e t1=temp[z=*]        ! extract temperature into variable
e t1                  ! load into PPLUS plot buffer
e nextr
e t1=temp[z=*]        ! extract next temperature into variable
e t1                  ! load into PPLUS plot buffer
plot                  ! plot both temperature profiles
```

Example 3:

Here is a simple calculation of a mean CTD temperature profile from three input CTD data files. The three profiles are then overplotted with the mean profile.

```
e openr “ctd.dat”
e nextr
e t1=temp[z=*]        ! extract temperature into variable
e nextr
e t2=temp[z=*]        ! extract next temperature into variable
e nextr
e t3=temp[z=*]        ! extract next temperature into variable
e t_avg=(t1+t2+t3)/3  ! calculate mean value
e t_avg               ! load avgerage into PPLUS plot buffer
e t1                  ! load temperatures into plot buffer
e t2
e t3
plot                  ! Plot avg and individual temperatures
```

Time Series Data

PPLUS recognizes EPIC time series data. PPLUS can extract axis labels and a plot title from the data file headers.

Program TIMPLT generates a PPLUS (.PPC) command file, which can be edited to modify the plot, or you can run PPLUS directly.

Examples

The following show the minimum PPLUS commands needed to plot EPIC time series data as individual plots, overplotted, and as stick plots.

Individual Plots (overplot u and v from a wind data file:

```
e openr “time.dat”
e nextr
e u1=u[t=*]            ! extract u from data file
e v1=v[t=*]            ! extract v from data file
e u1                   ! load u,v into PPLUS plot buffers
e v1
plot                   ! overplot u,v time series
```

Stick Plots:

```
e openr “time.dat”
e nextr
e u1=u[t=*]            ! extract u from data file
e v1=v[t=*]            ! extract v from data file
e u1                   ! load u,v into PPLUS plot buffers
e v1
plotuv                 ! make stick plot of vector data
```

Plotting with a time axis on the graph:

```
e openr “../data/t5n165e.cdf”
e nextr
e d1=dynhgt[t=*]       ! extract dynamic height from data file
e status d1            ! extract status information about data
time,w’eps$fld_tmin’,w’eps$fld_tmax’,w’eps$fld_tmin’! time axis
taxis,on
e d1                   ! load data into PPLUS plot buffer
plot
```

Plot Legends

Plot legends are placed on overplotted data by programs TIMPLT and CTDPLT for Time series and CTD data.

PPLUS EPIC Global Symbols

PPLUS puts information about EPIC data sets into global PPLUS symbols. These symbols can be used in a PPLUS command file to place labels on graphs. Examples of this can be seen in the PPLUS command files in the proceeding sections.

TABLE 3. PPLUS Symbols Set By E When Status Function Is Called

Symbol	Description
eps$fld_name	Long name of variable
eps$fld_min	Minimum value of field
eps$fld_max	Maximum value of field
eps$fld_xmin	Minimum x axis value
eps$fld_xmax	Maximum x axis value
eps$fld_ymin	Minimum y axis value
eps$fld_ymax	Maximum y axis value
eps$fld_zmin	Minimum z axis value
eps$fld_zmax	Maximum z axis value
eps$fld_tmin	Minimum t axis value
eps$fld_tmax	Maximum t axis value
eps$fld_nx	Number of points in x axis
eps$fld_ny	Number of points in y axis
eps$fld_nz	Number of points in z axis
eps$fld_nt	Number of points in t axis

TABLE 4. PPLUS Symbols Set By E When Function Nextr Is Called

Symbol	Description
eps$pointerfile	Pointer file name
eps$filetype	File format
eps$filename	Data file name
eps$datatype	Type of data (CTD, TIME)
eps$varlist	List of variable in data file, separated with “;”
eps$namelist	List of generic names in data file, separated with “;”
ppl$eof	End of file read. YES is true, NO if false

TABLE 5. PPLUS Symbols Set When A Field Is Loaded

Symbol	Description
eps$field	Field name (short variable name)
eps$varname	Field name (long variable name)
eps$varunits	Field units
eps$latitude	Field latitude
eps$longitude	Field longitude
eps$depth	Field depth
eps$date	Date of field
eps$xtype	type of x-axis, (LONE, LAT, DEPTH,TIME, TEMP, FIELD)
eps$xlabel	x-axis label
eps$xunits	x-axis units
eps$ytype	type of y-axis, (LAT, DEPTH, SALINITY, FIELD)
eps$ylabel	y-axis label
eps$yunits	y-axis units
eps$label	field label
eps$lowercrnr	string consisting of the minimum x, y, and z
eps$uppercrnr	string consisting of the maximum x, y, and z
eps$xmin	minimum value of the x-axis
eps$xmax	maximum value of the x-axis

TABLE 5. PPLUS Symbols Set When A Field Is Loaded

Symbol	Description
eps$ymin	minimum value of the y-axis
eps$ymax	maximum value of the y-axis
eps$tmin	minimum time (only valid if axis is time)
eps$tmax	maximum time (only valid if axis is time)

CHAPTER 8 Data Files

All data file reading and writing for the EPIC system is done with the EPIC System library (the EPS library), which has been discussed in a previous chapter titled Operating Systems: VMS And UNIX (page 29). The EPIC System library supports multiple data file formats. At present, supported formats include the Unidata netCDF format and the Classic EPIC format. Classic EPIC data files are binary sequential 1-dimensional data files designed for observational data with three of the four space-time axes represented by a single point, that is, each CTD profile is stored in a one-dimensional disk file with data values indexed by depth or pressure, and time series data from an instrument is stored in a one-dimensional disk file with the data values indexed by time. NetCDF data files are multi-dimensional, and are transportable between many different types of hardware platforms, including VAX/VMS, Unix workstations, Cray supercomputers, PCs and MACs. One-dimensional data has historically been stored in Classic EPIC data files, but will be stored in netCDF files as EPIC moves into the Unix environment. As described below, with use of the EPS library, the two data formats can be used interchangeably. Additional data file formats, such as the HDF format (developed by the National Center for Supercomputing Applications) could be supported by the EPS library in the future.

The EPIC system library consists of a “format independent layer”, which is a set of generic input/output routines which are called by application programs to open data files, read or write “attributes”, axes and data, and then close the files. These routines do not directly manipulate the data files on the disk. Rather, there is a lower level “format dependent layer” of routines for each of the supported data formats. These format dependent routines perform the actual data file manipulations for each of the supported formats. With the the EPS library, all application programs are independent of data file format, and additional formats (other than the Unidata netCDF format and the Classic EPIC format) could be supported by the addition of a set of eight “format dependent” routines to the “format dependent” layer of the library. Application programs will transparently read and write data files in any of the supported formats without modification.

EPIC system library routines are callable from Fortran and from C, and are supported for VAX/VMS and for Unix machines. There is an excellent Users Guide for the EPS library, and also a suite of very simple examples, including C and Fortran programs to read and write EPIC system files, and PPLUS scripts to read, plot, calculate and make animations with EPIC system files. On PMEL's VAX, examples are in the directory EP_EXAMPLES:, and on Unix, they are in the directories ~epic/examples and /usr/local/eps. The EPS library, the EPS Library Users' Guide, and all examples, are all available via PMEL's anonymous ftp account (see Chapter 2).

Details about the supported EPIC data formats (Classic EPIC format, and the EPIC implementation of the netCDF format) are provided in the following sections, but the user, in general, does not need to know these details. In order to read and write EPS data files, the user should use the EPS Library Users' Guide, and also the example programs provided. Often, users are able to write programs to read and write EPS data files by using the example programs as a starting point, often with little reference to the EPS Library Users' Guide. The example programs are provided with this intention.

Unidata NetCDF Data Format

The Unidata netCDF format and data access library was developed to support the creation, access and sharing of data in a form that is self-describing and network-transparent. This software was developed at UCAR under NSF sponsorship, and is in wide use in the oceanographic and scientific communities. It is freely available on internet for many hardware platforms, including VAX/VMS, DEC/Ultrix, Sun/Unix, Cray/Unicos and MACs. NetCDF data files are hardware independent, which means that they can be image copied from machine to machine, and then read with the netCDF subroutines. NetCDF data files are read and written with the netCDF library subroutines, which are part of the Internet distribution. There are also netCDF utilities to inspect the data files and to dump the data from the files into ascii disk files. The netCDF library, and related utilities, are all supported by Unidata.

PMEL has developed the EPIC system library, which is layered on top of the netCDF input/output library, to write netCDF files with the PMEL-EPIC conventions for oceanographic data. The EPS library is not intended to provide the complete funtionality which is available with the netCDF library, but rather to simplify the production of a standardized implementation of netCDF for oceanographic data. It transparently provides support for some commonly used variations on the recommended EPIC standard implementation, and could be modified to provide support for others.

The EPIC netCDF file Conventions, which describe the EPIC system implementation of the netCDF format, have been registered with Unidata, by placing the following documents in the Unidata /pub/netcdf/Conventions/PMEL-EPIC directory on the host ftp.unidata.ucar.edu:

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

Classic EPIC Data Format

The Classic EPIC data file is a binary sequential file with at least one header consisting of 8 80-character lines followed sequentially by data scans, one per logical record (no packing). The data scans can include any number of variables (up to 20). The variables are real numbers. For pressure-indexed data (like CTD data), the first variable is usually pressure or depth. For time series data (like current meter data), the first two variables are usually date and time in EPIC format (date is a real number yymmdd, time is a real number hhmm). Each data variable in the data scan is described in the header by a numeric variable code. The numeric codes are defined in the epic.key file, (on VMS, it is EP_KEY:EPIC.KEY, on Unix, it is /usr/local/lib/epic.key. See the EPS manual and the on-line HELP for EPICKEY for a description of this key file and of how to use it.

There are two basic formats for the headers, one for pressure-indexed data (like CTD data), and one for time series data. The headers contain a great deal of descriptive information about the data. They can be read and updated and written by utility subroutines (see the on-line HELP).

Classic EPIC format files which are created on VAX/VMS are readable on Unix with the EPS library, and will continue to be supported, but the EPIC system is moving towards increased use of the widely used Unidata netCDF file format.

Classic EPIC Hydrographic Header Format

Line 1

col 1-5 “CAST “
col 6-16 Cruise number; example - “JOINT-1-OC-”
col 17-19 Consecutive STD cast number.
col 20-27 “ Date “
col 28-36 Date of STD cast; “30 JUN 75”
col 37-44 “ TIME “
col 45-48 GMT time of STD; example- “0132”
col 49-52 “ GMT”
col 53-54 Blank
col 55-80 Instrument Type

Line 2

col 1-4 “LAT “
col 5-12 Latitude of STD cast; example - “45 15.2N”
or “45.2533N”
col 13-19 “ LONG “
col 20-28 Longitude of STD cast; example - “124 15.8W”
or “124.2635W”

col 29-38 " WEATHER "
col 39 Weather code (see table 3)
col 40-51 "SEA STATE "
col 52 Sea state code (see table 2)
col 53-54 Blank
col 55-69 File Creation date hh:mm dd-mmm-yy
col 70-76 Blank
col 77 Bottle-data-flag: B if this is bottle data,
blank otherwise.
col 78* Header-type-flag: C for standard CTD header,
Blank if non-standard header.
col 79* Next-Header-Flag: N if another header follows
this on, blank if not.
col 80 Water mass flag: B Bering, G Gulf of Alaska,
S Shelikof, V Vents, P Puget Sound, E or blank
Equatorial, C Chukchi or Arctic N. of Bering Strait

* All headers, whatever type, must have these flags.

Line 3

col 1-10 "BAROMETER "
col 11-12 Atmospheric pressure in millibars (mbs) over
1000 mbs; example - "19" means 1019 mbs.
col 13-23 " WIND DIR "
col 24-26 Wind direction in degrees from which the wind is
blowing.
col 27-33 " T SPD "
col 34-35 Wind speed in knots
col 36-51 " KT VISIBILITY "
col 52 Visibility code (see table 6).
col 53-54 Blank
col 55-60 Number of data scans
col 61-66 Depth of first scan
col 67-72 Depth of last scan
col 73-77 Depth increment between scans (Blank for
unevenly spaced data)
col 78 Blank
col 79-80 Number of variables per scan

Line 4

col 1-6 "CLOUD "
col 7 Cloud type code (see table 4)
col 8-16 " AMOUNT "
col 17 Cloud amount code (see table 5)
col 18-23 " DRY "
col 24-27 Dry air temperature in degrees Celsius to the nearest tenth.
col 28-33 " WET "
col 34-37 Wet bulb temperature In degrees Celsius to the nearest tenth.
col 38-46 " DEPTH "
col 47-50 Water depth to the nearest meter
col 51-52 " M"
col 53-54 Blank
col 55-80 Data Origin Information

Line 5

col 1-4 EPICKEY Variable code for 1st variable in scan
col 5-8 EPICKEY Variable code for 2nd variable in scan
col 9-12 EPICKEY Variable code for 3rd variable in scan
.
.
.
col 77-80 EPICKEY Variable code for 20th variable in scan

Line 6 (WOCE information in fixed format, optional)

col 1-6 Start date (yymmdd); "880427"
col 7-10 Start time (hhmm); "0900"
col 11 blank
col 12-18 Start latitude (ddmm.mm); "1501.08" (15 deg, 01.08 min)
col 19 Hemisphere for start latitude; "N"
col 20 blank
col 21-28 Start longitude (dddmm.mm); "12959.83" (129 deg, 59.83 min)
col 29 Hemisphere for start longitude; "E"
col 30 blank
col 31-34 Water depth in meters; "5969"
col 35 blank
col 36-37 CTD number (2 digits only)

col 38 blank
col 39-43 Sample rate (Hz); “31.00”
col 44 blank
col 45-48 WOCE line number; “P16C”
col 49 blank
col 50-55 End date (yymmdd); “880427”
col 56-59 End time (hhmm); “1009”
col 60 blank
col 61-67 End latitude (ddmm.mm); “1502.11”
(15 deg, 02.11 min)
col 68 Hemisphere for end latitude; “N”
col 69 blank
col 70-77 End longitude (dddmm.mm); “12956.73”
(129 deg, 56.73 min)
col 78 Hemisphere for end longitude; “E”
col 79-80 blank

Line 7 Two analysis program comment fields

col 1-40 Analysis Program Comments (most recent)
col 41-80 Analysis Program Comments (second-most recent)

Line 8 Two analysis program comment fields

col 1-40 Analysis Program Comments (third-most recent)
col 41-80 Analysis Program Comments (fourth-most recent)

Classic EPIC Time Series Header Format

Primary Header

Line 1

col 1-5 “LAT “
col 6-13 Latitude, e.g., “45 15.2N”
or “45.2533N”
col 14-21 “ LONG “
col 22-30 Longitude, e.g., “124 15.2N”
or “124.2635W”
col 31-38 “ DEPTH “
col 39-44 Depth in meters, e.g., “2000.0”

col 45 "M"
col 46-47 Blank
col 48-52 Mooring identification, e.g. "ET-06"
col 53-54 Blank
col 55-60 Starting date (uummdd) e.g., "841101"
col 61 Blank
col 62-65 Starting time (hhmm)
col 66-67 Blank
col 68-73 Ending date (yymmdd)
col 74 Blank
col 75-78 Ending time (hhmm)
col 79-80 Blank

Line 2

col 1-5 "DT = "
col 6-15 text field describing series delta-t,
e.g.,"20 MINUTES"
col 16-35 text field describing variables,
e.g., " TEMPERATURE "
col 36-45 project name, e.g., "EPOCS "
col 46-47 Blank
col 48-52 Experiment name, e.g., "ATLAS"
col 53-54 Blank
col 55-69 File Creation date, e.g., "hh:mm dd-mm-yy"
col 70-71 Blank
col 72-73 number of pieces in composite series
col 74 Flag to indicate position consistency
(1 for inconsistency, 0 or blank otherwise)
If Drifter data, this is a position quality
flag, 1 if position is low quality.
col 75 Flag to indicate depth consistency
(1 for inconsistency, 0 or blank otherwise)
col 76 Filler flag (fill value is 1.E35)
0 or blank if no missing values; 1 for missing values
col 77 D if this is Drifing buoy data, blank otherwise
col 78* Header-type-flag: T for standard Time Series
header, Blank if non-standard header.
col 79* Next-Header-Flag: N if another header follows this
on, blank if not.
col 80 Water mass flag: B Bering, G Gulf of Alaska,
S Shelikof, V Vents, P Puget Sound, E or blank

Equatorial, C Chukchi or Arctic N. of Bering Strait
* All headers, whatever type, must have these flags.

Line 3

col 1-14 Starting GMT time e.g., "0000 12 NOV 85"
col 15-18 " TO "
col 19-32 Ending GMT time, e.g.,"0000 03 APR 86"
col 33-47 Length of Series in days, e.g., " (140.0 DAYS)"
col 48-52 Water depth in meters, e.g., " 2000"
col 43-54 Blank
col 55-60 Number of data points in series
col 61-78 Series delta-t in minutes
col 79-80 Number of variables per scan of data (including date and time as variables 1 and 2)

Line 4

col 1-26 Text Description field
col 27-52 Instrument Type Information
col 53-54 Blank
col 55-80 Data Origin Information

Line 5

col 1-4 EPICKEY Variable code for 1st variable in scan
col 5-8 EPICKEY Variable code for 2nd variable in scan
col 9-12 EPICKEY Variable code for 3rd variable in scan
.
.
.

col 77-80 EPICKEY Variable code for 20th variable in scan

Line 6

col 1-60 Data Comment Field
col 61-80 Missing data flags for variables 1-20
0 or blank if no data is missing
1 if data is missing (fill value is 1.E35)

Line 7 Two analysis program comment fields

col 1-40 Analysis Program Comments (most recent)
col 41-80 Analysis Program Comments (second-most recent)

Line 8 Two analysis program comment fields

col 1-40 Analysis Program Comments (third-most recent)
col 41-80 Analysis Program Comments (fourth-most recent)

Piece Header

The piece headers contain descriptive information about the individual pieces of composite time series. The composite time series are created by concatenating the data from multiple deployments of an instrument at a specific buoy location.There is one non-blank line of header for each piece in the composite, formatted as follows:

Line 1

col 1-2 Piece number, e.g., “ 3”
col 3 Blank
col 4-11 Latitude, e.g., “45 15.2N”
col 12 Blank
col 13-21 Longitude, e.g., “124 15.2N”
col 22 Blank
col 23-36 Starting GMT time e.g., “12 NOV 85 0000”
col 37 Blank
col 38-51 Ending GMT time, e.g.,”03 APR 86 0000”
col 52-56 Depth in meters, e.g., “ 300M”
col 57 Blank
col 58-77 Instrument and/or mooring identification
col 78-80 Blank

Line 2 (Blank except for flag in column 78 when not needed to describe a “piece” in the composite.)

col 1-2 Piece number, e.g., “ 2”
col 3 Blank
col 4-11 Latitude, e.g., “45 15.2N”
col 12 Blank
col 13-21 Longitude, e.g., “124 15.2N”
col 22 Blank
col 23-36 Starting GMT time e.g., “01 JAN 86 0000”
col 37 Blank
col 38-51 Ending GMT time, e.g.,”05 FEB 86 0000”
col 52-56 Depth in meters, e.g., “ 300M”
col 57 Blank
col 58-77 Instrument and/or mooring identification
col 78* Header-type-flag: P for Piece header flag
Blank if non-standard header.

col 79* Next-Header-Flag: N if another header follows this
on, blank if not.
col 80 Blank
* All headers, whatever type, must have these flags.

Lines 3 - 8 (Blank if not needed to describe a “piece” in the composite series.)

col 1-2 Piece number, e.g., “ 3”
col 3 Blank
col 4-11 Latitude, e.g., “45 15.2N”
col 12 Blank
col 13-21 Longitude, e.g., “124 15.2N”
col 22 Blank
col 23-36 Starting GMT time e.g., “05 FEB 86 0000”
col 37 Blank
col 38-51 Ending GMT time, e.g.,”03 APR 86 0000”
col 52-56 Depth in meters, e.g., “ 300M”
col 57 Blank
col 58-77 Instrument and/or mooring identification
col 78-80 Blank

Data File Naming Conventions

The final, master version of the data files are placed on a “data disk” by the responsible project or data manager. Conventions for the naming of the data files, and the directory structure, can be decided by the project which collects or manages the data. The EPIC system has no dependence on these conventions, and, they should be chosen in a manner which is intuitive or logical for those who will be working with the data. This section outlines the conventions chosen by PMEL for data collected in the equatorial Pacific, and are provided only as an illustration. Data files are located on the disk with VAX/VMS name DH: (the same disk is cross-mounted on Unix with the name /home/hayes). CTD type files are in the directory DH:[DATA] (/home/hayes/data on Unix). Time series files are in the directories DH:[COMP] (composite series) and DH:[TIME] (individual time series). When this disk is cross-mounted on Unix, these names are /home/hayes/comp and /home/hayes/time.

Hydrographic Data File Names

CTD-type data file names are a combination of: the cruise identification (e.g. EP484 for EPOCS cruise 4 in 1984), the data type (C for CTD, B for bottle data, X for XBT data), and the cast number (001 or 1001). The extension indicates the data type (CTD, BOT, XBT). Here is a sample CTD data file name: [DATA.EP484.CTD]EP484C001.CTD (on Unix, it is data/ep484/ctd/ep484c001.ctd). Any pertinant information regarding calibration or the cruise on which the data was collected will be in a documentation file in the directory with the data files, with a name like [DATA.EP286.CTD]EP286CTD.DOC.

Time Series Data File Names

Time series data file names are a combination of: the data type (T for Temperature, P for Pressure, U for currents..), the location (e.g., 110W2S), the experiment identification (one id for a time series from one deployment, two ids to indicate the first and last deployments of a composite series) (e.g. EP14 for EPOCS mooring 14, or ET2_ET12 for a composite series whose first piece is from buoy ET2 and whose last piece is from buoy ET12) and the delta-t of the time series (H for Hourly data, D for Daily data, X for some other delta-t or unevenly spaced data). The extension indicates the depth of the instrument. Here are some sample time series file names, followed by their Unix equivalents:

VAX/VMS:

```
DH:[TIME.TMP.110W.2S]T110W2S_EP14D.010
DH:[TIME.TMP.110W.2S]T110W2S_ET2D.010
DH:[COMP.PRS.91W.1S]P91W1S_S4_5D.PRS
```

Unix:

```
/home/hayes/time/tmp/110w/2s/t110w2s_ep14d.010
/home/hayes/time/tmp/110w/2s/t110w2s_et2d.010
/home/hayes/comp/prs/91w/1s/p91w1s_s4_5d.prs
```

Variable Definitions

EPIC Key File

Each variable in an EPIC data file is described by a numeric code in the data file. There is a disk file named “epic.key” which contains the variable code, a short label, a long label, the generic name, the units of the variable, the format for printing the data values (in Fortran format notation), and a comment field. The short, long and generic names are character strings. The short name is four characters or less, and is suitable for use as a title for a column on a data listing. The long name is intended to be used in a label, such as an axis label on a graph. The generic name is used to access data from within the nccalc netCDF calculater, which is linked with the PPLUS graphics package. The EPIC variable key code is an integer with 4 or fewer digits, and is a unique identifier for the variable. The comment field is optional, and includes information about the variable, such as “Temperature in degrees Centigrade on the 1968 temperature scale”. For example, Air Temperature in a data file has numeric variable code of 21, short name of “AT”, long name of “Air Temperature, C”, generic name of “atemp”, units of “C”, format of “f10.2”, and a comment field of “1968 standard”.

The file epic.key is used by EPIC system routines to identify or locate the variables in a data file. It is also used by PPLUS to label plots automatically. If the data file is in Classic EPIC format, then the data file contains only numeric variable codes to identify each variable in the file, and all other information about the variable is found in the epic.key file by the EPIC programs and by the PPLUS graphics package. For Classic EPIC data formatted files, the epic.key file is required. For netCDF formatted data files, the information in the epic.key file is replicated in the data file attributes. At the time when the data file is written, this information can be included in the netCDF file by simply specifying the appropriate numeric variable key code. The EPIC system library will extract the other information from the epic.key file for inclusion in the data file as attributes. Alternatively, an application program writing the netCDF file can set some or all of these attributes in the data file directly, without specifying a numeric variable key code. It is possible to write a netCDF file which omits the numeric code and these attributes entirely, but this omission will result in a lack of information available to EPIC programs and also to PPLUS.

The following example illustrates the use of the epic.key file by EPIC programs. Plotting programs prompt the user to enter the variable to be plotted. If the user chooses Temperature and Oxygen, then the program will search through the variable codes for each of the variables to locate one which it can identify as a temperature and another which it can identify as an oxygen, and can then make a plot of those variables. Alternatively, the user can select the variables to be plotted by specifying the numeric variable codes, for explicit, exact identification of the data in the file. If the numeric variable codes are absent from the data file, then the program will be unable to locate a temperature or oxygen variable for plotting.

The epic.key file is also used by PPLUS to obtain variable information for Classic EPIC formatted data files. For netCDF files, PPLUS obtains this information from the netCDF attributes. PPLUS can then label plots automatically. The following table summarizes the relationship between the variable information in the epic.key file, the netCDF data file attributes, and the use of this information by PPLUS. EPIC variable information in epic.key file, in netCDF data files, and associated use by PPLUS, is as follows:

TABLE 6.

epic.key key file element	**netCDF variable attribute name**	**PPLUS variable attribute name**	**PPLUS global symbol name**	**PPLUS use for data labels**
variable code	epic_code	varid	in eps$varlist	
short label	name	name		title
long label	long_name	lname	eps$fld_name	axis
generic name	generic_name	gname	in eps$namelist	

TABLE 6.

epic.key key file element	netCDF variable attribute name	PPLUS variable attribute name	PPLUS global symbol name	PPLUS use for data labels
units	units	units		
format	FORTRAN_format	frmt		
comment field	(unused)	(unused)	(unused)	

The epic.key file changes as new variable codes are added to it. On VAX/VMS, the file is EP_KEY:EPIC.KEY. On unix, at PMEL, the file is /opt/src/epic/lib/epic.key. The newest version of the file can been seen with the EPIC utilities “key” and “epickey”. The listing presented here may no longer be current:

```
  :   :UNDEFINED               : :counts: :Undefined Variable
 0:   :Undefined               :generic:units:format:Undefined Variable
 1:P  :PRESSURE (DB)           :depth:dbar:f10.1:
 2:D  :DEPTH (CM)              :depth:cm: :
 3:D  :DEPTH (M)               :depth:m:f10.1:
 4:P  :PRESSURE (PASCALS)      :depth:Pa: :
 5:PD :PRESSURE DIFFERENCE (DB) :depth:dbar: :
 6:ISO:ISOTHERM DEPTH (M)      :depth:m: :
 7:SL :SEA LEVEL ANOMALY (M)   :depth:m: :
 8:MLD:MIXED LAYER DEPTH (M)   :depth:m: :Program MIXDEP
 9:P  :SEA LEVEL PRESSURE (MB) :depth:mbar: :For Ferret
10:DYN:DYNAMIC METERS          :dynhgt:dyn-m:f10.4:
11:DYN:DYM HT (SUBSAMPLED DATA) :dynhgt: : :
12:dDN:DYNn-DYNs               :dynhgt: :f10.5:Diff of North and South Dyn Ht.
13:DYN:DYNAMIC HEIGHT (DYN CM) :dynhgt:dyn-cm: :
14: dD:dDYN1-dDYN2 (DYN M)     :dynhgt:dyn-m: :Delta-Dyn (ctd1-ctd2) for geost vel calc
15:d1 :DYN1 (DYN M)            :dynhgt:dyn-m: :Dyn1 (ctd1) for geost vel calc
16:d2 :DYN2 (DYN M)            :dynhgt:dyn-m: :Dyn2 (ctd2) for geost vel calc
17:ZPG:ZONAL PRESSURE GRAD     : : : :Calculated for specific areas
18:hght:height (m)             :height:m:f10.2:
20:T  :TEMPERATURE (C)         :temp:C:f10.2:IPTS-1968 standard
21:AT :AIR TEMPERATURE (C)     :atemp:C: :
22:T0 :THERM TEMPERATURE (C)   :temp:C: :Reversing therm. 1E35 indicates no therm.
23:AT :AIR TEMPERATURE MIN (C) :atemp:C: :
24:AT :AIR TEMPERATURE MAX (C) :atemp:C: :
25:T  :SST (C)                 :temp:C: :
26:T  :SST MIN (C)             :temp:C: :
27:T  :SST MAX (C)             :temp:C: :
28:T  :TEMPERATURE (C)         :temp:C:f10.2 :ITS-1990 Standard
29:IT :TEMP INTEGRAL           :temp:C: :Depth integrated temperature
30:PT :POTENTIAL TEMP (C)      :potemp: : :PT = Potential Temperature
31:P4 :POTENTIAL TEMP 4000 (C) :potemp: : :Potential temp with 4000m reference level
32:T  :TEMPERATURE (C)         :temp:C:f10.4:unknown standard
33:T  :CORRECTED TEMPERATURE (C):temp:C:f10.4:corrected for mooring motion
34:T2 :Secondary Temperature   :temp:C:f10.2:IPTS-1968 standard
35:T2 :Secondary Temperature   :temp:C:f10.2:ITS-1990 Standard
38:PT :POTENTIAL TEMP (C)      :potemp: : :Potential Temperature (ITS-90)
40:S  :SALINITY (PPT)          :sal:PPT:f10.2:From SALIN1
41:S  :SALINITY (PSU)          :sal:PSU: :Practical Salinity Units
42:S  :SALINITY (PSU)          :sal:PSU: :Practical Salinity Units
43:S0 :BOTTLE SALINITY (PSU)   :sal:PSU: :From Salinometer. 1E35 indicates no sample.
44:S3 :Alternate Salinity      :sal: : :Practical Salinity Units
45:S4 :Alternate Salinity      :sal: : :Practical Salinity Units
49:S  :SALINITY                :sal: :f10.2:calculated for XBT’s from mean T-S curve, M McCarty
50:C  :CONDUCTIVITY            :con:mmho/cm: :
51:C  :CONDUCTIVITY            :con:S/m:f10.3:
52:rcon:RAW CONDUCTIVITY (SEABIRD):cond:counts:i10:Seabird cell raw conductivity
```

```
 53:rcon:RAW CONDUCTIVITY (VACM) :cond:counts:i10:vacm raw conductivity
 54:C2  :Secondary Conductivity  :con:mmho/cm: :
 55:ATTN:ATTENUATION             :attn:m-1:f7.5:added for r2d2 ctd data
 56:NEP:BACKSCATTER INTENSITY    :nephylometer:v:f10.6: added for Dave Pashinski
 60:O  :OXYGEN (ML/L)            :ox:ml/l:f10.2:Dissolved oxygen calculated from CTD values
 61:BO :BOTTLE OXYGEN (ML/L)     :ox:ml/l: :DO determined by titration. 1E35 ->no sample
 62:OST:OXYGEN, %SAT             :ox:%: :
 64:AOU:APPARENT O2 UTILIZATION  :ox:ml/l: :
 65:O  :OXYGEN (UMOL/KG)         :ox:umol/kg:f10.2:Dissolved oxygen calculated from CTD values
 66:BO :BOTTLE OXYGEN (UMOL/KG)  :ox:umol/kg: :DO determined by titration. 1E35 ->no sample
 70:ST :SIGMA-T (KG/M**3)        :den:kg m-3:f10.2:
 71:STH:SIGMA-THETA (KG/M**3)    :potden:kg m-3 :f10.2:
 72:S2 :SIGMA 2 (KG/M**3)        :potden2:kg m-3:f10.2:sigma-theta ref to 2000m
 73:S3 :SIGMA 3 (KG/M**3)        :potden3:kg m-3:f10.2:sigma-theta ref to 3000m
 74:S4 :SIGMA 4 (KG/M**3)        :potden4:kg m-3:f10.2:sigma-theta ref to 4000m
 75:S5 :SIGMA 5 (KG/M**3)        :potden5:kg m-3:f10.2:sigma-theta ref to 5000m
 76:S1 :SIGMA 1 (KG/M**3)        :potden1:kg m-3:f10.2:sigma-theta ref to 1000m
 77:GN :GAMMA N (KG/M**3)        :gamma_n:kg m-3:f10.2:Neutral Density
 80:SV :SOUND VELOCITY (M/S)     : :m s-1: :
 81:FAC:SND SPD CORR. FACTOR:factor::f10.3:corr. BASS currents for actual sound speed
 84:RHO:DENSITY (KG/M**3)        :den:kg m-3:f10.2:
 85:SPV:SPECIFIC VOLUME          : : : :
 86:SVA:SP VOL ANOM, 1E-8 M**3/KG: :m3 kg-1: :
 90:BV :BVF (CPH)                : : : :Brunt Vaisala Frequency
 91:BV2:BVF SQUARED (CPH**2)     : : : :Brunt Vaisala Frequency squared (N*ABS(N))
 92:R  :RADIUS OF DEFORM         : : : :Integral bvf*dz/pi*f
 93:RI :RICHARDSONS NUMBER       : : : :
100:SN :SCAN NUMBER              : : : :
101:FS :FRAME SYNC               : : : :FOR TOPS
102:CL :PTG CLOCK WORD           : : : :
103:BTL:NISKIN BOTTLE NUMBER     : : : :For Bottle files
104:IQ :WOCE QUALITY NUMBER      : : : :See WOCE Document
105:NAV:NUMBER POINTS IN AVERAGE : : : :
106:BAT:BATTERY VOLTAGE          :bat:volts:f6.2:
107:TRN:TRANSMISSOMETER VOLTAGE 7: : : :for P18 CTD data set
110:OXC:OXYGEN CURRENT           : : : :
111:OXT:OXYGEN TEMPERATURE       : : : :
112:DOC:dOXC/dt                  : : : :Time rate of change of Oxygen current
113:BAT1:BATTERY VOLTAGE 1       :bat:volts:f10.3:
114:BAT2:BATTERY VOLTAGE 2       :bat:volts:f10.3:
115:BAT3:BATTERY VOLTAGE 3       :bat:volts:f10.3:
120:T' :GRADIENT DT/DZ           : : : :Program CTDGRD2
121:DZT:GRADIENT DZ/DT           : : : :Program CTDGRD3
122:T'X:GRADIENT DT/DX           : : : :X is longitude
123:T'Y:GRADIENT DT/DY           : : : :Y is latitude
124:UTX:U * dT/dx                : : : :X is longitude
125:DZT:GRADIENT DZ/DT           : : : :Program CTDGRD3
126:ADT:AD TEMP GRAD,DEG-C/DB    : : : :
130:HTC:HEAT CONTENT (JOU/M**2)  : :J m-2: :
131:SpH:SPECIFIC HEAT, J/KG-DEG-C: :J kg-1 C-1: :
132:Qs :SHORTWAVE RADIATION      : :cal day-1 cm-2: :
133:Qs :SHORTWAVE RADIATION      : :W m-2: :
134:HQ :LOCAL HEATING            : : : :
135:Ql :LONGWAVE RADIATION       : :cal day-1 cm-2: :
136:Ql :LONGWAVE RADIATION       : :W m-2: :
137:QH :LATENT HEAT              : :W m-2: :
138:QB :SENSIBLE HEAT            : :W m-2: :
140:QI :PENETRATIVE RADIA (PING) : : : :
141:QT :QT=QS-QB-QE (PING)       : : : :
142:QN :NET HEATING              : :W m-2: :
143:VMF:VERTICAL+MERID HEAT FLUX : :W m-2: :
144:ZMA:ZONAL+MERID HEAT ADVECT  : :W m-2: :
145:VHF:VERTICAL HEAT FLUX       : :W m-2: :
146:MHF:MERIDIONAL HEAT FLUX     : :W m-2: :
147:ZHA:ZONAL HEAT ADVECTION     : :W m-2: :
148:MHA:MERIDIONAL HEAT ADVECTION: :W m-2: :
150:FR :FREON                    : : : :
152:NO3:NITRATE (MG AT/L)        :NO3:mg at/l: :
154:NO2:NITRITE (MG AT/L)        :NO2:mg at/l: :
156:PO4:PHOSPHATE (MG AT/L)      :PO4:mg at/l: :
```

```
157:Ptl:TOTAL PHOSPHORUS         : :mg at/l: :J. Reid data
158:SI :SILICATE (MG AT/L)       :SiO4:mg at/l: :mg at/l
159:pH :pH                       : : : :J. Reid data
170:SPC:SPICE                    :spc : : :From Phyllis Stabeno (CTD data)
181:SI :SiO4                     :SiO4:mg at/l: :J.Murray
182:NO3:NITRATE (micromoles/l)   :NO3:uM/l: :From Jim Murray
184:NO2:NITRITE (micromoles/l)   :NO2:uM/l: :From Jim Murray
185:NN :Nitrate + Nitrite        :NO2+NO3:umoles/l: :Total nutrients
186:PO4:PHOSPHATE (micromoles/l) :PO4:uM/l: :From Jim Murray
187:UR :UREA (micromoles/l)      :UR:uM/l: :From Jim Murray
188:SI :SILICATE (micromoles/l)  :SiO4:uM/l: :From Jim Murray
189:NH4:AMMONIA (micromoles/l)   :NH4:uM/l: :From Jim Murray
200:DIS:VERTICAL DISPLACEMENT    : : : :From mean CTD profile
210:SL :GEOSAT SEA LEVEL (M)     : :m: :Mean value over grid block
220:AZ :AZIMUTH                  :az:degrees:f8.2:measure of ice floe rotation
221:VAR:VARIANCE                 :var:cm s-1: :sqrt(u**2 + v**2) (current variance)
250:GHR:GLOBAL HORIZNTL RADIATION:ghr:kJ s-2 d-1:f8.2:global horizontal radiation
251:PAR:PHOTOSYN ACTIVE RADIATION:par:M ph m-2 d-1:f5.2 :photosynthetically active radiation
252:GHR:GLOBAL HORIZNTL RADIATION:ghr:W m-2:f8.2:global horizontal radiation
282:NO3:NITRATE (micromoles/kg)  :NO3:uM/kg: :From Jim Murray
284:NO2:NITRITE (micromoles/kg)  :NO2:uM/kg: :From Jim Murray
286:PO4:PHOSPHATE (micromoles/kg):PO4:uM/kg: :From Jim Murray
287:UR :UREA (micromoles/kg)     :UR:uM/kg: :From Jim Murray
288:SI :SILICATE (micromoles/kg) :SiO4:uM/kg: :From Jim Murray
289:NH4:AMMONIA (micromoles/kg)  :NH4:uM/kg: :From Jim Murray
300:CS :CURRENT SPEED (CM/S)      :vspd:cm s-1 :f8.2:oceanographic (going to)
301:BS :DRIFTER BUOY SPEED (M/S) : :m s-1: :calculated from lat,lon
310:CD :CURRENT DIRECTION (T)     :vdir:degrees:f8.2:oceanographic (going to), using true north
311:BD :DRIFTER BUOY DIRECTION   : :degrees: :oceanographic (going to)
318:BU :DRIFTER BUOY U (M/S)     :u:m s-1: :calculated from lat,lon
319:BV :DRIFTER BUOY V (M/S)     :v:m s-1: :calculated from lat,lon
320:U  :ZONAL CURRENT (CM/S)     :u:cm s-1: :measured w/r true N
321:V  :MERIDIONAL CURRENT (CM/S):v:cm s-1: :measured w/r true N
322:U  :ZONAL CURRENT (CM/S)     :u:cm s-1: :relative w/r true N
323:V  :MERIDIONAL CURRENT (CM/S):v:cm s-1: :relative w/r true N
324:U  :ZONAL CURRENT (CM/S)     :u:cm s-1: :instrument w/r true N
325:V  :MERIDIONAL CURRENT (CM/S):v:cm s-1: :instrument w/r true N
326:U  :ZONAL CURRENT (CM/S)     :u:cm s-1: :tracked w/r true N
327:V  :MERIDIONAL CURRENT (CM/S):v:cm s-1: :tracked w/r true N
328:W  :VERTICAL VELOCITY (CM/S) :w:cm s-1: :tracked TOPS
329:W  :VERTICAL VELOCITY (CM/S) :w:cm s-1:f8.2:
330:U  :ZONAL CURRENT (CM/S)     :u:cm s-1: :measured w/r magnetic N
331:V  :MERIDIONAL CURRENT (CM/S):v:cm s-1: :measured w/r magnetic N
332:U  :ZONAL CURRENT (CM/S)     :u:cm s-1: :relative w/r magnetic N
333:V  :MERIDIONAL CURRENT (CM/S):v:cm s-1: :relative w/r magnetic N
334:U  :ZONAL CURRENT (CM/S)     :u:cm s-1: :instrument w/r magnetic N
335:V  :MERIDIONAL CURRENT (CM/S):v:cm s-1: :instrument w/r magnetic N
336:U  :ZONAL CURRENT (CM/S)     :u:cm s-1: :tracked w/r magnetic N
337:V  :MERIDIONAL CURRENT (CM/S):v:cm s-1: :tracked w/r magnetic N
340:Ug :GEOSTROPHIC ZONAL (CM/S) : :cm s-1: :geostrophic
341:Vg :GEOSTR MERIDIONAL (CM/S) : :cm s-1: :geostrophic
342:Umg:Umodel-Ugeost (cm/s)     : :cm s-1: :
344:Ut :TRACK NORMAL GEOST (CM/S):un:cm s-1: :geostrophic vel normal to track
350:HX :COMPASS HX               : : : :
351:HY :COMPASS HY               : : : :
360:AX :ACCELERATION AX          : : : :
361:AY :ACCELERATION AY          : : : :
362:AU :ACCELERATION AU          : : : :
363:AV :ACCELERATION AV          : : : :
370:X  :X POSITION IN NET        : : : :tracked TOPS position
371:Y  :Y POSITION IN NET        : : : :tracked TOPS position
372:Z  :Z POSITION IN NET        : : : :tracked TOPS position
375:U1 :U Tidal Amplitude (CM/S) species1:u1:cm s-1: :measured w/r true N
376:V1 :V Tidal Amplitude (CM/S) species1:v1:cm s-1: :measured w/r true N
377:U2 :U Tidal Amplitude (CM/S) species2:u2:cm s-1: :measured w/r true N
378:V2 :V Tidal Amplitude (CM/S) species2:v2:cm s-1: :measured w/r true N
380:ZT :ZONAL TRANSPORT, M**2/SEC: :m2 s-1: :total transport per unit width
381:MT :MERIDIONAL TRANSPRT,M^2/S:U:m2 s-1: :total transport per unit width
382:ZT+:ZONAL TRANSPORT (+)      :V:m2 s-1: :positive zonal transport per unit width
383:MT+:MERIDIONAL TRANSPORT (+) : :m2 s-1: :positive meridional transport per unit width
```

```
384:ZT-:ZONAL TRANSPORT (-)      : :m2 s-1: :negative zonal transport per unit width
385:MT-:MERIDIONAL TRANSPORT (-) : :m2 s-1: :negative meridional transport per unit width
386:ZT :ZONAL TRANSPORT, M**3/SEC:U:m3 s-1: :volume transport
387:MT :MERIDIONAL TRANSPRT,M^3/S:V:m3 s-1: :volume transport
388:TR :TRK NORMAL TRNSPRT, M^3/S: :m3 s-1: :volume transport normal to track
390:ZU :ZONAL TRANSPORT UPPER LIM: :m2 s-1: :transport per unit width
391:MU :MERIDIONAL TRNS UPPER LIM: :m2 s-1: :transport per unit width
392:ZL :ZONAL TRANSPORT LOWER LIM: :m2 s-1: :transport per unit width
393:ML :MERIDIONAL TRNS LOWER LIM: :m2 s-1: :transport per unit width
394:ZCW:ZONAL CURRENT WIDTH      : : : :
395:MCW:MERIDIONAL CURRENT WIDTH : : : :
396:Dx :ZONAL MOORING LINE DRAG  : : : :
397:Dy :MERIDIANAL MOOR LINE DRAG: : : :
398:DRG:DRAG ON MOORING LINE     : : : :
400:WS :WIND SPEED (CM/S)        : :cm s-1: :
401:WS :WIND SPEED (M/S)         : :m s-1: :
402:WG :WIND GUST (M/S)          : :m s-1: :greatest speed in interval (drifters)
410:WD :WIND DIRECTION           : :degrees: :     0-360 degrees
411:WD :WIND DIRECTION           : :degrees: :     -180 to 180 degrees
420:WU :WIND U (CM/S)            :u:cm s-1: :oceanographic sense (going to)
421:WV :WIND V (CM/S)            :v:cm s-1: :oceanographic sense (going to)
422:WU :WIND U (M/S)             :u:m s-1: :oceanographic sense (going to)
423:WV :WIND V (M/S)             :v:m s-1: :oceanographic sense (going to)
440:T-X:ZONAL WIND STRESS        :u:dyne cm-2: :Zonal Wind Stress
441:T-Y:MERIDIONAL WIND STRESS   :v:dyne cm-2: :Meridional Wind Stress
442:PTx:ZONAL PSEUDO WIND STRESS : :m2 s-2: :Zonal pseudo wind stress
443:PTy:MERID PSEUDO WIND STRESS : :m2 s-2: :Meridional pseudo wind stress
444:Txy:TOTAL PSEUDO WIND STRESS : :m2 s-2: :sqrt(PTx**2 + PTy**2)  (Prog WNDSTR)
446: Tx:ZONAL WIND STRESS        : : : :Zonal true wind stress
447: Ty:MERID WIND STRESS        : : : :Meridional true wind stress
448:Txy:TOTAL WIND STRESS        : : : :sqrt(Tx**2 + Ty**2) (Prog WNDSTR)
449:GP :Geopotential             :GP:m2/s2:f10.2 :geopotential energy
450:WW :WIND WORK                : :m3 s-3: :Wind work
460:U'X:GRADIENT DU/DX           : :1E-5 s-1: :X is longitude
461:V'X:GRADIENT DV/DX           : :1E-5 s-1: :X is longitude
462:U'Y:GRADIENT DU/DY           : :1E-5 s-1: :Y is latitude
463:V'Y:GRADIENT DV/DY           : :1E-5 s-1: :Y is latitude
480:OLR:OLR                      : :W/M**2: :Outgoing long wave radiation
485:RN :RAIN                     : :MM: :Rainfall derived from OLR
500:LAT:LATITUDE                 :lat:degree_north:f10.4:EPIC-- N is positive, S is negative
501:LON:LONGITUDE                :lon:degree_west:f10.4:Classic EPIC-- W is pos, eg, 165W=165, 170E=190
502:LON:LONGITUDE                :lon:degree_east:f10.4:New EPIC-- E is pos, eg, 165E=165, 170W=190
600:TIM:TIME (EPIC)              :time: :f10.0:EPIC Time series time hhmm as real number
620:DAT:DATE (EPIC)              :time: :f10.0:EPIC Time series date yymmdd as real number
621:DAT:MMDD.HHMM                :time: : :Date,Time without year (mmdd.hhmm, real no)
622:TIM:GREENWICH DAYS           :time: : :Days from the start of the century
623:TIM:GEOSAT DAYS              :time: : :Days after 85/01/01
624:TIM:EPIC SYSTEM TIME         :time: : :2 integers, jul day, millisec since midnight
625:TIM:SECONDS SINCE START      :time:s: :integer, delta seconds
626:TIM:DAYS SINCE START         :time:d: :real, delta seconds
627:TIM:Julian Days              :time: : :Real number, decimal part gives time of day
650:H  :LAYER THICKNESS          :H:m: :Mick Spillane's model
651:n  :LAYER NUMBER             :n: : :Mick Spillane's model
700:SD :STANDARD DEVIATION       : : : :
710:N  :NUMBER OF POINTS         : : : :
721:PG :PERCENT GOOD PINGS       : : : :Acoustic Doppler
722:EV :ERROR VELOCITY           : : : :RD Acoustic Doppler
723:AMP:AMPLITUDE                : : : :RD Acoustic Doppler
810:SDD:STAND. DEV. (DYN.HT)     : : : :
820:SDT:STAND. DEV. (TEMP)       : : : :
840:SDS:STAND. DEV. (SAL)        : : : :
841:SD+:SAL+STAND. DEV.          : : : :
842:SD-:SAL-STAND. DEV.          : : : :
843:S+2:SAL+2 STAND. DEV.        : : : :
844:S-2:SAL-2 STAND. DEV.        : : : :
850:SDP:STAND. DEV. (PRESS)     :pres:mbar:f10.5:std. deviation of burst pressures
851:SED:STAND. ERR. (DYN.HT)     : : : :
852:SET:STAND. ERR. (TEMP)       : : : :
853:SES:STAND. ERR. (SAL)        : : : :
854:SEP:STAND. ERR. (PRESS)      : : : :
```

```
 901:Fvt:Fluorometer Volts(CTD)   : :Volts: :
 902:Irr:Irradiance(CTD)          : :uEin cm-2 s-1: :
 903:F  :Fluorometer (CTD)        : :mg m-3: :J.Murray 5/13/92
 904:Tr :Transmissometry (CTD)    : :%: :J.Murray 5/13/92
 905:PAR:Photosynthetic Active Radiation:par:uEin m-2 s-1: :J.Murray 5/13/92
 906:Fch:Chlorophyll A            : :ugrams/l: :(fluorometric) J.Murray 5/9/94
 907:Fph:Phaeopigments      : :ugrams/l: :(fluorometric) J.Murray 5/9/94
 910:RH :RELATIVE HUMIDITY (%)    : :%: :PERCENT
 911:RH :RELATIVE HUMIDITY MIN (%): :%: :
 912:RH :RELATIVE HUMIDITY MAX (%): :%: :
 913:SH :SPECIFIC HUMIDITY (G/KG) : :g kg-1: :For Ferret
 915:BP :BAROMETRIC PRESSURE (MB) : :mbar: :
 920:CC :CLOUD COVER (OKTAS)      : : : :OKTAS
 931:Chv:Chlorophyll-a Volts(CTD) : :Volts: :
 932:Cla:Chlorophyll-a(CTD)       : :ugrams/l: :
 935:VA3:ChlAM Voltage            : :volts:WetLabs A-3
 936:AA3:Chlorophyll-a Absorption : :m-1:WetLabs A-3
 937:CA3:Chlorophyll-a Conc       : :ugrams/l:WetLabs A-3
 941:DB :Dry Bulb Temp            : :degree C: :
 942:WB :Wet Bulb Temp            : :degree C: :
 951:DP :Dew Point                : :degree C: :
 961:Rn :Rainfall                 : :10-2 in: :
 971:rFv:raw fluorometer Volts(CTD)   : :Volts: :
 972:rCv:raw chl-a Volts(CTD)     : :Volts: :
 990:BUG:Plankton Concentration   : :number/l: : individuals per liter
1141:YDF:MERID DIFFUS HEAT FLUX   : :W m-2: :ping 12/10/90
1142:MHF:MERID ADVECT HEAT FLUX   : :W m-2: :ping 12/10/90
1143:VE :VERTICAL ENTRAINMENT     : :W m-2: :ping 12/10/90
1144:Qt :HEAT CONTENT CHANGE      : :W m-2: :ping 12/11/90
1145:Qt*:QT*=QNET+QU+QW+QZ        : :W m-2: :ping 12/11/90
1200:Tim :Ensemble Length          :ens_length:s: :ADCP Ensemble Length
1201:Werr:Error Velocity           :w:cm/s:F8.1:ADCP Error Velocity
1202:AGC :Avg. Echo Intensity (AGC):AGC:counts:F5.1:ADCP Beam Averaged AGC
1203:PGd :Percent Good Pings       : : : :ADCP Percent Good Pings
1204:w   :Vertical Velocity        :w:cm/s: :Vertical Velocity
1205:u   :Eastward Velocity        :u:cm/s: :Eastward Velocity
1206:v   :Northward Velocity       :v:cm/s: :Northward Velocity
1207:Rec :Records                  : : : :ADCP Records
1208:Png :Pings                    : : : :ADCP Pings
1209:EISD:Echo Intensity (AGC) Std. Dev.:AGC:counts:F5.1:ADCP AGC (1202) Std Dev
1210:BIT :ADCP BIT Status          : : : :ADCP Built-in Test Status
1211:Tx  :ADCP Transducer Temp.    :temp:deg. C:F10.2:ADCP Transducer Temp
1215:Hdg :INST Heading             :hdg:degrees:F10.2:ADCP heading
1216:Ptch:INST Pitch               :ptch:degrees:F10.2:ADCP pitch
1217:Roll:INST Roll                :roll:degrees:F10.2:ADCP roll
1218:HSD :Heading Std. Dev.        :hdg:degrees: :ADCP Hdg. Std. Dev.
1219:PSD :Pitch Std. Dev.          :ptch:degrees: :ADCP Pitch Std. Dev.
1220:RSD :Roll Std. Dev.           :roll:degrees: :ADCP Roll Std. Dev.
1221:AGC1:Echo Intensity (AGC) Beam 1:AGC:counts: :ADCP Beam 1 AGC
1222:AGC2:Echo Intensity (AGC) Beam 2:AGC:counts: :ADCP Beam 2 AGC
1223:AGC3:Echo Intensity (AGC) Beam 3:AGC:counts: :ADCP Beam 3 AGC
1224:AGC4:Echo Intensity (AGC) Beam 4:AGC:counts: :ADCP Beam 4 AGC
1231:Sv  :Mean Backscattering Strength:Sv:dB: :ADCP Mn Bksctr, ref to last bin
1232:Sv  :Mean Backscattering Strength:Sv:dB: :ADCP Mn Bksctr, ref to noise cal
1233:Sv1 :Backscattering Strength Beam 1:Sv:dB: :ADCP Bksctr., ref to noise cal
1234:Sv2 :Backscattering Strength Beam 2:Sv:dB: :ADCP Bksctr., ref to noise cal
1235:Sv3 :Backscattering Strength Beam 3:Sv:dB: :ADCP Bksctr., ref to noise cal
1236:Sv4 :Backscattering Strength Beam 4:Sv:dB: :ADCP Bksctr., ref to noise cal
1237:SvSD:Backscattering Strength Std. Dev:Sv:dB: :ADCP Bksctr. Std. Dev.
1241:PGd1:Percent Good Beam 1         : : : :ADCP Percent Good Beam 1
1242:PGd2:Percent Good Beam 2         : : : :ADCP Percent Good Beam 2
1243:PGd3:Percent Good Beam 3         : : : :ADCP Percent Good Beam 3
1244:PGd4:Percent Good Beam 4         : : : :ADCP Percent Good Beam 4
1245:PG4B:Percent 4-Beam Solns. Good  : : : :ADCP Percent 4-Beam Solns. Good
1246:PG3B:Percent 3-Beam Solns. Good  : : : :ADCP Percent 3-Beam Solns. Good
1247:PGWe:Percent Error Vel. Good     : : : :ADCP Percent Error Vel. Good
1251:St1 :Status Beam 1               : : : :ADCP Beam 1 Status
1252:St2 :Status Beam 2               : : : :ADCP Beam 2 Status
1253:St3 :Status Beam 3               : : : :ADCP Beam 3 Status
1254:St4 :Status Beam 4               : : : :ADCP Beam 4 Status
```

```
1255:StBn:Bin Status                   : : : :ADCP Bin Status
1260:BTWe:BT Error Velocity            :w:cm/s: :ADCP Btm. Trk. Error Velocity
1261:BTu :BT Eastward Velocity         :u:cm/s: :ADCP Btm. Trk. Eastward Velocity
1262:BTv :BT Northward Velocity        :v:cm/s: :ADCP Btm. Trk. Northward Velocity
1263:BTw :BT Vertical Velocity         :w:cm/s: :ADCP Btm. Trk. Vertical Velocity
1264:BTR1:BT Range Beam 1              : :m: :ADCP Btm. Trk. Range Beam1
1265:BTR2:BT Range Beam 2              : :m: :ADCP Btm. Trk. Range Beam2
1266:BTR3:BT Range Beam 3              : :m: :ADCP Btm. Trk. Range Beam3
1267:BTR4:BT Range Beam 4              : :m: :ADCP Btm. Trk. Range Beam4
1268:BTRa:BT Average Range             : :m: :ADCP Btm. Trk. Range Average
1269:BTSD:BT Range Std. Dev.           : :m: :ADCP Btm. Trk. Range St Dev
1270:BTP1:BT Percent Good Beam 1       : : : :ADCP Btm. Trk. Beam 1 % Good Pings
1271:BTP2:BT Percent Good Beam 2       : : : :ADCP Btm. Trk. Beam 2 % Good Pings
1272:BTP3:BT Percent Good Beam 3       : : : :ADCP Btm. Trk. Beam 3 % Good Pings
1273:BTP4:BT Percent Good Beam 4       : : : :ADCP Btm. Trk. Beam 4 % Good Pings
1275:Un  :Track Normal Velocity        :un:cm/s: :ADCP Vel. normal to track
1276:Vt  :Track Tangential Velocity    :vt:cm/s: :ADCP Vel. tangential to track
1277:vel1:Beam 1 velocity (mm/s)     : :mm s-1:f10.2:ADCP velocity along beam 1
1278:vel2:Beam 2 velocity (mm/s)     : :mm s-1:f10.2:ADCP velocity along beam 2
1279:vel3:Beam 3 velocity (mm/s)     : :mm s-1:f10.2:ADCP velocity along beam 3
1280:vel4:Beam 4 velocity (mm/s)     : :mm s-1:f10.2:ADCP velocity along beam 4
1281:att1:Beam 1 attenuation         : :counts:i10:ADCP attenuation of beam 1
1282:att2:Beam 2 attenuation         : :counts:i10:ADCP attenuation of beam 2
1283:att3:Beam 3 attenuation         : :counts:i10:ADCP attenuation of beam 3
1284:att4:Beam 4 attenuation         : :counts:i10:ADCP attenuation of beam 4
1285:cor1:Beam 1 correlation         : :counts:i10:ADCP correlation of beam 1
1286:cor2:Beam 2 correlation         : :counts:i10:ADCP correlation of beam 2
1287:cor3:Beam 3 correlation         : :counts:i10:ADCP correlation of beam 3
1288:cor4:Beam 4 correlation         : :counts:i10:ADCP correlation of beam 4
1290:xmtc:ADCP transmit current      :xmitc:amps:f10.3:
1291:xmtv:ADCP transmit voltage      :xmitv:volts:f10.3:
1292:dac :ADCP DAC output            :dac:counts:i10:
1293:vdc :ADCP VDC voltage           :vdc:volts:f10.3:
1301:BPR  :BOTTOM PRESSURE             : :PSIA: :Pounds Per Square Inch Absolute
1371:U48:U Amplitude (CM/S) 48hrs :u48:cm s-1: :measured w/r true N
1372:V48:V Amplitude (CM/S) 48hrs :v48:cm s-1: :measured w/r true N
1380:Res :MTR resistance          :res:ohm:f10.2:Instrument resistance
1401:hdg :AMP compass heading     :comp:degrees:f10.2:compass heading
1402:van :AMP vane                :vane:degrees:f10.2:vane reading
1403:van :VANE ANGLE              :vane:degrees:f10.1:current orientation to instrument
1404:comp:COMPASS (m)             :comp:degreeM:f10.1:instrument orientation to magnetic north
1406:comp:ORIENTATION (BASS)      :comp:degrees:f10.3:orientation of BASS pods
1407:comp:RAW COMPASS VOLTS (MIDAS):comp:counts:i10:MIDAS raw compass voltage
1408:rcomp:ENCODED COMPASS        :comp:128.lvl.binary:i10:vacm grey code compass
1409:rvane:ENCODED VANE           :vane:128.lvl.binary:i10:vacm grey code vane
1411:bear:bearing (m)      :dir:degreeM:f8.2:instantaneous dir., magnetic north (oceanographic)
1412:bear:bearing (t)      :dir:degreeT:f8.2:instantaneous dir., true north (oceanographic)
1413:bear:bearing (r)      :dir:degrees:f8.2:instantaneous dir., rotated (oceanographic)
1414:vdir:vector direction (m):vdir:degreeM:f8.2:averaged dir., magnetic north (oceanographic)
1415:vdir:vector direction (t):vdir:degreeT:f8.2:averaged dir., true north (oceanographic)
1416:vdir:vector direction (r):vdir:degrees:f8.2:averaged dir., rotated (oceanographic)
1500:LAT:START LATITUDE            :start_lat:degree_north:f10.4:N positive S negative
1501:LON:START LONGITUDE           :start_lon:degree_west:f10.4:W positive 165W=165
1502:LON:START LONGITUDE           :start_lon:degree_east:f10.4:E positive 165W=195
1510:LAT:END LATITUDE              :end_lat:degree_north:f10.4:N positive S negative
1511:LON:END LONGITUDE             :end_lon:degree_west:f10.4:W positive 165W=165
1512:LON:END LONGITUDE             :end_lon:degree_east:f10.4:E positive 165W=195
1601:IQ:WEPOCS QUALITY ID          :iq: : :Billy Kessler's WEPOCS CTD
1701:F11:CFC-11 (PMOL/KG)          :f11:pmol/kg:f10.6:trichloroflouoromethane
1702:F12:CFC-12 (PMOL/KG)          :f12:pmol/kg:f10.6:dichlorofluoromethane
1711:PF11:PCFC-11                  :pf11: :f5.2:CFC-11 equilibrium partial-pressure
1712:PF12:PCFC-12                  :pf12: :f5.2:CFC-12 equilibrium partial-pressure
1721:PF11AGE:Pf11age               :pf11age:years:f8.3:cfc11 partial pressure age
1722:PF12AGE:Pf12age               :pf12age:years:f8.3:cfc12 partial pressure age
1723:PFRATAGE:Pfratage             :pfratage:years:f8.3:cfcratio partial pressure age
1731:F11CORR:CORRECTED CFC-11      :f11corr:pmol/kg:f10.6:corrected trichloroflouoromethane
1732:F12CORR:CORRECTED CFC-12      :f12corr:pmol/kg:f10.6:corrected dichlorofluoromethane
1751:TCO2:TOTAL CO2 (micromoles/kg):dic:uM/kg:f6.1:Dissolved Organic Carbon (micromoles/kg)
1752:NTCO2:NORMALIZED TCO2 (uM/kg):ndic:uM/kg:f6.1:Normalized Dissolved Organic Carbon (micromoles/kg)
```

```
1753:DIC:DISS. INORG. CARBON (umol/kg):dic:umol/kg:f6.1:Dissolved Inorganic Carbon (micromoles/kg) from Dick
                            Feely
1754:TALK:TOTAL ALKALINITY (uM/kg):alk:uM/kg:f6.1:Total Alkalinity from Dick Feely
1755:PCO2:PARTIAL CO2 PRESSURE (microATM):pco2:uATM:f6.4:Partial CO2 Pressure from Dick Feely
1900:V00:VARIABLE 0               : : : :
1901:V01:VARIABLE 1               : : : :
1902:V02:VARIABLE 2               : : : :
1903:V03:VARIABLE 3               : : : :
1904:V04:VARIABLE 4               : : : :
1905:V05:VARIABLE 5               : : : :
1906:V06:VARIABLE 6               : : : :
1907:V07:VARIABLE 7               : : : :
1908:V08:VARIABLE 8               : : : :
3001:Tec :ADCP Electronics Temp. :temp:deg. C:F10.2:ADCP Electronics Box Temp.
3002:Vs  :ADCP Voltage          : :v:F10.2:ADCP Supply Voltage
3003:sBP:seas barometric pressure:BP:mbs:f10.2:seas barometric pressure
3004:sWB:seas wet bulb temp:wbT:degree C:f10.2:seas wet bulb temp
3005:sAT:seas air temp:atemp:degree C:f10.2:seas air temp
3006:sWD:seas wind direction:WD:degree:f10.2:seas wind direction
3007:mWB:manual wet bulb temp:wbulb:degree C:f10.2:manual wet bulb temp
3008:mAT:manual air temp:dbulb:degree C:f10.2:manual air temp
3009:mT:manual water temp:temp:degree C:f10.2:manual water temp
3010:MFdep:Miller Freeman depth:depth:meters:f10.1:depth from Miller Freeman
3011:lat1:gps1 lat 1 min:lat:degrees:f10.3:gps1 lat 1 min
3012:lon1:gps1 lon 1 min:long:degrees:f10.3:gps1 lon 1 min
3013:sct:seacat temperature:temp:degree C:f10.2:seacat temperature
3014:scc:seacat conductivity:cond:mmhos:f10.2:seacat conductivity
3015:scs:seacat salinity:sal:ppt:f10.2:seacat salinity
3016:scz:seacat depth:depth:meters:f10.2:seacat depth
3017:Tx  :ADCP Transducer Temp.     :temp:deg. C:F10.2:ADCP Transducer Temp.
3018:avTD:avg true wind dir:WD:degree:f10.2:avg true wind dir
3019:avTS:avg true wind speed:WS:m/sec:f10.2:avg true wind speed
3020:avC:avg conductivity:C:mmhos:f10.2:avg conductivity
3021:avT:avg temperature:T:degree C:f10.2:avg temperature
3022:avS:avg salinity:S:ppt:f10.2:avg salinity
3023:rWS:rm young wind speed:WS:knots:f10.2:rm young wind speed
3024:rWD:rm young wind dir:WD:degree:f10.2:rm young wind dir
3025:rAT:rm young air temp:atemp:degree C:f10.2:rm young air temp
3026:ocWD:oc true direction:WD:degree:f10.2:oc true direction
3027:ocWS:oc wind speed:WS:knots:f10.2:oc wind speed
3028:ocWDA:OC Wind Dir Avg:WD:degree:f10.2:OC Wind Dir Avg
3029:ocWSA:OC Wind Spd Avg:WS:knots:f10.2:OC Wind Spd Avg
3030:light:light intensity:light:degrees C:f10.2:light intensity
3031:fluor:fluorometer:fluor:volts:f10.2:fluorometer
3032:EKdep:EK500 depth:depth:meters:f10.1:depth from MF EK500
3033:EQdep:EQ50 depth:depth:meters:f10.1:depth from MF EQ50
3034:rWD:RMY true direction:WD:degree:f10.2:oc true direction
3035:rWS:RMY wind speed:WS:knots:f10.2:oc wind speed
3036:SD:gps ship direction:SD:degrees:f10.2:ship direction
3037:SS:gps ship speed:SS:knots:f10.2:ship speed
3038:su:gps ship u component:su:knots:f10.2:ship east vel
3039:sv:gps ship v component:sv:knots:f10.2:ship north vel
4000:upr:UPPER ROTOR SPEED     :rotor:counts:f10.1:basic measurement, vmcm
4001:upr:UPPER ROTOR SPEED     :rotor:cm/s:f10.3:interval upper rotor speed, Seadata tripod
4002:lowr:LOWER ROTOR SPEED    :rotor:counts:f10.1:basic measurement, vmcm
4003:lowr:LOWER ROTOR SPEED    :rotor:cm/s:f10.3:interval lower rotor speed, Seadata tripod
4004:rdif:ROTOR SPEED DIFFERENCE  :rotor:cm/s:f10.3:difference, interval rotor speed, Seadata tripod
4005:rot:ROTOR COUNTS             :rotor:counts:i10:vacm rotor counter
4006:rspd:ROTOR SPEED             :speed:cm s-1:f8.2:scalar speed from rotor counts
4007:vspd:VECTOR SPEED            :vspd:cm s-1:f8.2:vector speed from U and V
4008:cnt:clock counter            :count:counts:i10:clock counter
4009:stead:STEADINESS            :steadiness: :f10.4:VACM rotor spd normalized by vector spd to measure wave
                            effect
4010:tran:TRANSMISSION (VOLTS)     :trans:volts:f10.3:basic measurement, transmissometer
4011:ptrn:PERCENT TRANSMISSION     :trans: :f10.1:normalized by 95% of air transmission
4012:rtrn:RAW TRANSMISSOMETER VOLTS:trans:counts:i10:MIDAS raw transmissometer voltage
4015:rtltx:RAW TILTX VOLTS          :tilt:counts:i10:MIDAS raw tiltx voltage
4016:rtlty:RAW TILTY VOLTS          :tilt:counts:i10:MIDAS raw tilty voltage
4017:tiltx:TILT IN POD X-Z PLANE    :tilt:degrees:f10.3:orientation of BASS pods
4018:tilty:TILT IN POD Y_Z PLANE    :tilt:degrees:f10.3:orientation of BASS pods
```

```
4019:tilta:TILT ANGLE               :tilt:degrees:f8.2:averaged instrument tilt FSIacm
4020:P   :AVERAGE RELATIVE PRESSURE:pres:mbar:f10.5:burst pressure minus interval pressure, Seadata tripod
4022:P   :INTERVAL PRESSURE        :pres:mbar:f10.3:interval pressure measurement, Seadata tripod
4023:P   :AVERAGE BURST PRESSURE   :pres:mbar:f10.3:average of burst pressures
4026:rpres:RAW PRESSURE (PAROS,COMP,AB):pres:counts:i10:Paroscientific, temp. compensated, A,B calibration
4027:rpres:RAW PRESSURE (PAROS,UNCOMP,AB):pres:counts:i10:Paroscientific, not temp. compensated, A,B cali-
                            bration
4028:rpres:RAW PRESSURE (PAROS,COMP,CD):pres:counts:i10:Paroscientific, temp. compensated, C,D calibration
4029:rpres:RAW PRESSURE (PAROS,UNCOMP,CD):pres:counts:i10:Paroscientific, not temp. compensated, C,D cali-
                            bration
4030:rpres:RAW PRESSURE (VACM):pres:counts:i10:vacm raw pressure
4031:rpres:RAW PRESSURE (MX2):pres:counts:i10:mx-vacm with 2 mx-variables
4032:rpres:RAW PRESSURE (MX4):pres:counts:i10:mx-vacm with 4 mx-variables
4041:rtemp:RAW TEMPERATURE (SEABIRD) :temp:counts:i10:Seabird cell raw temperature
4042:rtemp:RAW TEMPERATURE (MIDAS THERM) :temp:counts:i10:MIDAS thermister raw temperature
4043:rtemp:RAW TEMPERATURE (VACM) :temp:counts:i10:vacm raw temperature
4045:rtemp:RAW TEMPERATURE (MX2):temp:counts:i10:mx-vacm with 2 mx-variables
4046:rtemp:RAW TEMPERATURE (MX4):temp:counts:i10:mx-vacm with 4 mx-variables
4050:UVAR:EAST VELOCITY VARIANCE   :uvar:cm2/s2:f10.2:variance of east component, burst BASS data
4051:UVCOV:U-V VELOCITY COVARIANCE :uvcovar:cm2/s2:f10.2:covariance of east and north, burst BASS data
4052:VVAR:NORTH VELOCITY VARIANCE  :vvar:cm2/s2:f10.2:variance of north component, burst BASS data
4053:UWCOV:U-W VELOCITY COVARIANCE :uwcovar:cm2/s2:f10.2:covariance of east and up, burst BASS data
4054:VWCOV:V-W VELOCITY COVARIANCE :vwcovar:cm2/s2:f10.2:covariance of north and up, burst BASS data
4055:WVAR:UP VELOCITY VARIANCE     :wvar:cm2/s2:f10.2:variance of up component, burst BASS data
4056:peru:ZERO CROSSING PERIOD, U  :period_u:s:f10.2:zero crossing period of east component, burst BASS data
4057:perv:ZERO CROSSING PERIOD, V  :period_v:s:f10.2:zero crossing period of north component, burst BASS
                            data
4060:wp  :average wave period       :wave_period:s:f10.2:
4061:wh  :significant wave height   :wave_height:m:f10.2:
4070:BASA:BASS COUNTS, AXIS A      :bass:counts:f10.1:basic data of BASS pod
4071:BASB:BASS COUNTS, AXIS B      :bass:counts:f10.1:basic data of BASS pod
4072:BASC:BASS COUNTS, AXIS C      :bass:counts:f10.1:basic data of BASS pod
4073:BASD:BASS COUNTS, AXIS D      :bass:counts:f10.1:basic data of BASS pod
4074:BadBA:BAD BASS COUNT, AXIS A   :badbass:count:f10.1:bad data cycle count, BASS pod
4075:BadBB:BAD BASS COUNT, AXIS B   :badbass:count:f10.1:bad data cycle count, BASS pod
4076:BadBC:BAD BASS COUNT, AXIS C   :badbass:count:f10.1:bad data cycle count, BASS pod
4077:BadBC:BAD BASS COUNT, AXIS D   :badbass:count:f10.1:bad data cycle count, BASS pod
4080:u   :East(m)                   :u:cm s-1:f8.2:U component (magnetic)
4081:v   :North(m)                  :v:cm s-1:f8.2:V component (magnetic)
4082:u   :East(t)                   :u:cm s-1 :f8.2:U component (true north)
4083:v   :North(t)                  :v:cm s-1:f8.2:V component (true North)
4084:u   :East(r)                   :u:cm s-1 :f8.2:U component (rotated)
4085:v   :North(r)                  :v:cm s-1:f8.2:V component (rotated)
4086:u   :instant East(m)           :u:cm s-1:f8.2:instantaneous east comp (magnetic) FSIacm
4087:v   :instant North(m)          :v:cm s-1:f8.2:instantaneous north comp (magnetic) FSIacm
4088:u   :instant East(t)           :u:cm s-1:f8.2:instantaneous east comp (true north) FSIacm
4089:v   :instant North(t)          :v:cm s-1:f8.2:instantaneous north comp (true north) FSIacm
4091:w   :instant vertical velocity :w:cm s-1:f8.2:instantaneous vertical velocity FSIacm
4095:re  :east register             :u:counts:i10:vacm east register
4096:rn  :north register            :v:counts:i10:vacm north register
```

CHAPTER 9

Data Management

The EPIC data selection process uses a relational data base. When you type EPIC and answer questions about data selection criteria, this information is translated into a data base query, which returns a list of data files meeting the data selection criteria to the requesting user, in the form of the “EPIC file” described previously. At present, the VAX/VMS version of EPIC utilizes the Boeing RIM relational data base, which is not available for Unix systems. Since the relational data base is used only for the data selection process, only a very few programs are dependent on the data base. The data select module reads or queries the data base to locate data file names, and the data loading modules read the data files to make entries in the data base. PMEL is in the process of choosing a relational data base for Unix. However, it is anticipated that EPIC dependence on any specific commercial data base product will be very minimal, and that users outside PMEL will be able to utilize any different relational data base with minimal difficulty.

The remainder of this chapter discusses the present data base in use at PMEL for VAX/VMS, which relies on Boeing RIM. A similar structure for data management will be used with a more modern relational data base for Unix. However, at this time, there is no EPIC data selection progam available for Unix.

RIM

When you type “EPIC” to start the data selection process, and answer questions about your data selection criteria, this information is put into a file (ZZRIMRUN.ZZZ) in the form of RIM directives. RIM then searches the

RIM data base tables to locate the data you've selected and writes an output file (the "EPIC file"). The ZZRIM-RUN.ZZZ file is then deleted. The user need know nothing about RIM.

The public data base tables containing the index for local (NOAA) data are in the directory EP$RIM:. Within PMEL, there is a RIM data base for each Division, sometimes each project, within the Laboratory. These are each maintained by people designated as "responsible" for processing and managing data for that Division or project. The RIM data bases tables can have any external name, but the internal data base name will always be SHAYES, and they will be referred to by the VMS logical names SHAYES1, SHAYES2, SHAYES3 by all programs. They are RIM SHARED data bases, and reside in the EPIC directory EP$RIM:. Each of these data bases can be accessed by many users simultaneously (concurrent access). These tables do not contain any of the data sets. They contain an index to the data files, which includes such information as geographic location, time, date, and VAX data file specifications. The data files reside on whatever disk has been designated as the data disk by the project owning the data.

Updating

In general, putting data into EPIC is a two stage process. First, the data sets are created in EPIC format. Then entries are made in EPIC's RIM data base tables by running a FORTRAN program which utilizes RIM's application program interface. It reads the data files and loads the header information into a local copy of the data base.

The usual loading procedure includes getting a copy of the appropriate public data base tables with the utility GETDB and loading new data information into the local copy, with one of the loading programs. This procedure is described completely in the on-line HELP ($ HELP EPIC RIM DATA_LOADING). Data loading programs include CTDLOAD (for CTD data), XBTLOAD (for XBT data), MTRLOAD, FOCILOAD and TIMLOAD (for time series data in different formats). There are sample command files to run each of these in the same directory with the source code and the executables. These programs are documented in the disk file EP$TIME:LOAD.DOC and in the VAX on-line help RIM. Some documentation on PMEL data bases is in the document EP_DOC:DBTABLES.DOC.

It is sometimes desirable to load information into an empty set of data base tables, and in that case you can get an empty set of data base tables with "$ COPYDB DH:[NNS.DBASEBCK]EMPTY HAYES". NOTE: You must use GETDB to get copies of the public data base tables to load into. You must use COPYDB to get copies of the empty data base tables. The reason is explained in the next paragraph.

There are two kinds of RIM data base tables: shared and single user. The public data base tables are shared with an internal name of SHAYES. The empty data base tables are single user with an internal name of HAYES. In the single user environment, if any user has a data base open, no other user can open any data base with the same internal name. This restriction is lifted in the shared environment. However, in the shared RIM environment, loading information into a data base requires that all other users of a data base with the same name (ie, SHAYES) be locked out during the loading process (regardless of the fact that the data base tables are a copy). The solution is for the loading to be done into copies of the data base which have been made single user and given the internal name of HAYES. There are utilities available to make this process easy. See the on-line HELP and see the following section for a description of the PMEL utilities for working with RIM data bases.

Utilities

The following are some useful utilities for handling the data base tables. All EPIC users have these utilities defined.

GETDB EP$RIM:SHAYES disk:[dir]name

The utility GETDB gets a copy of the public data base tables, changes the internal name to HAYES and makes them single-user.

PUTDB disk:[dir]name EP$RIM:FOREIGN

After the loading is complete, the utility PUTDB puts a copy of this single user data base with an internal name of HAYES into the specified directory, converts it to a shareable data base and changes the internal name to SHAYES. (NOTE - You can't "PUTDB old_name new_name" if the destination directory has a database with external name of "old_name".)

COPYDB name1 name2

Copies data base tables without changing any of their characteristics. Example: COPYDB EP$RIM:SHAYES DISK$HAYES:[NNS.DBASEBCK]SHAYES

RENAMEDB old_name old_version new_name new_version

Renames the data base tables VAX file names without changing any of their data base characteristics (such as the internal name). Example: RENAMEDB HAYES 3 GFDL 1

DELETDB name ver

Deletes data base tables with VAX file names name1.dat;ver, name2.dat;ver, name3.dat;ver. Example: DELETDB GFDL 1

DEFDB internal_name name

Sets up a VAX/VMS logical to allow access to a data base with an internal name of "internal_name" where the physical data base files are named "name1.dat", "name2.dat", "name3.dat". For example, if you have data base tables with the internal name HAYES and the physical files are named GFDL1.DAT, GFDL2.DAT, GFDL3.DAT, then you could give the command $ DEFDB HAYES GFDL. After that, you could use RIM directly to inspect or modify the data base.

ASSDB name internal_name

Sets up a VAX/VMS logical to allow access to a data base with an internal name of “internal_name” where the physical data base files are named “name1.dat”, “name2.dat”, “name3.dat”. For example, if you have data base tables with the internal name HAYES and the physical files are named GFDL1.DAT, GFDL2.DAT, GFDL3.DAT, then you could give the command $ ASSDB GFDL HAYES. After that, you could use RIM directly to inspect or modify the data base.

DEASSDB internal_name

Deassigns the logicals “internal_name1”, “internal_name2”, “internal_name3”, ie, reverses the ASSDB previously described. Example: DEASSDB HAYES undoes the assignment ASSDB GFDL HAYES

EPCOPYDB

Prompts user for name of originating directory, for destination name of the data base tables, and for a comment line. Does a PUTDB of the local tables (single user with internal and external name of HAYES) to EP$RIM: and makes a backup copy on EP$BCK:. Makes an entry in EP$RIM:dbname.LOGFILE for this update. THIS UTILITY REQUIRES EPIC SYSTEM PRIVILEGES.

PEPIC

Used to access privately owned (single user, internal name HAYES) data base tables, such as those which you are loading information into with CTDLOAD, XBTLOAD or MTRLOAD, etc. In the process of loading you have gotten a private copy of the data base tables with either GETDB (from EP$RIM:) or with COPYDB (from EP$RIM:EMPTY) and put this copy into one of your directories with the local name HAYES. Give this directory the logical name DTEST with $ DEFINE DTEST disk:[dir] and then type PEPIC. It will run the EPIC program on your own copy of the data base.

RIM Attributes

CTD

Here are the attributes for CTD-type data in the RIM data base tables:

FILENAME	TEXT	14 CHARS	VAX FILE NAME FOR DATA SETS
TOPFILE	TEXT	14 CHARS	ASSOCIATED TOPS FILE NAME

CRUISE	TEXT	10 CHARS	CRUISE IDENTIFICATION
CAST	INT	1	CTD, XBT, TOPS CAST NUMBER
ZI	REAL	1	INITIAL DEPTH OF CAST (M)
ZF	REAL	1	FINAL DEPTH OF CAST
DZ	REAL	1	CAST DATA INCREMENTAL DEPTH
LATI	REAL	1	- IS SOUTH AND + IS NORTH
LONGI	REAL	1	WEST IS + AND EAST IS 360-LONGITUDE
TIMI	REAL	1	START TIME IN HOURS SINCE START OF CENTURY
DATEXT	TEXT	3	CHARS DATA TYPE, E.G., CTD,XBT,TOP
INSTYP	TEXT	26	CHARS INSTRUMENT TYPE
DIRECT	TEXT	20	CHARS VAX DATA FILE DIRECTORY
DATORIG	TEXT	26	CHARS DATA ORIGIN INFORMATION
TEMPFLG	TEXT	1	CHARS EQUALS T IF GOOD TEMP DATA
SALFLG	TEXT	1	CHARS EQUALS S IF GOOD SAL DATA
UVFLG	TEXT	1	CHARS EQUALS U IF GOOD CURRENT DATA
FILCREAT	TEXT	15	CHARS DATE OF VAX DATA FILE CREATION
NVAR	INT	1	NUMBER OF VARIABLES PER LINE
VAR1	INT	1	VARIABLE CODE FOR VARIABLE 1
VAR2	INT	1	
VAR3	INT	1	
VAR4	INT	1	
VAR5	INT	1	
VAR6	INT	1	
VAR7	INT	1	
VAR8	INT	1	
VAR9	INT	1	
VAR10	INT	1	
VAR11	INT	1	
VAR12	INT	1	
VAR13	INT	1	
VAR14	INT	1	
VAR15	INT	1	
VAR16	INT	1	
VAR17	INT	1	
VAR18	INT	1	
VAR19	INT	1	
VAR20	INT	1	
LATITUDE	TEXT	8	CHARS TEXT LATITUDE
LONGITUD	TEXT	9	CHARS TEXT LONGITUDE
DATE	TEXT	14	CHARS TEXT DATE FIELD
DISK	TEXT	4	CHARS VAX DISK NAME

Time Series

Here are the attributes for time series data in the RIM data base tables:

FILE	TEXT	22	CHARS TIME SERIES DATA FILE NAME
LATI	REAL	1	- IS SOUTH AND + IS NORTH
LONGI	REAL	1	WEST IS + AND EAST IS 360 LONGITUDE
TIMI	REAL	1	START TIME IN HOURS SINCE START OF CENTURY
DATEXT	TEXT	3	CHARS DATA TYPE - EX, CTD,TOP,TMP,PRS
INSTYP	TEXT	26	CHARS INSTRUMENT TYPE
DIRECT	TEXT	20	CHARS VAX DATA FILE DIRECTORY
DATORIG	TEXT	26	CHARS DATA ORIGIN INFORMATION
FILCREAT	TEXT	15	CHARS DATE OF VAX DATA FILE CREATION
NVAR	INT	1	NUMBER OF VARIABLES PER LINE
VAR1	INT	1	VARIABLE CODE FOR VARIABLE 1
VAR2	INT	1	
VAR3	INT	1	
VAR4	INT	1	
VAR5	INT	1	
VAR6	INT	1	
VAR7	INT	1	
VAR8	INT	1	
VAR9	INT	1	
VAR10	INT	1	
VAR11	INT	1	
VAR12	INT	1	
VAR13	INT	1	
VAR14	INT	1	
VAR15	INT	1	
VAR16	INT	1	
VAR17	INT	1	
VAR18	INT	1	
VAR19	INT	1	
VAR20	INT	1	
LATITUDE	TEXT	8	CHARS TEXT LATITUDE
LONGITUD	TEXT	9	CHARS TEXT LONGITUDE
DATE	TEXT	14	CHARS TEXT DATE FIELD
DISK	TEXT	4	CHARS VAX DISK NAME
START	TEXT	14	CHARS SERIES START TIME/DATE
ENDTIME	TEXT	14	CHARS SERIES END TIME/DATE
DEPTH	REAL	1	DEPTH OF MOORED INSTRUMENT (M)

DELTIME	REAL	1	SERIES DELTA-TIME (MINUTES)
DEP	TEXT	2	CHARS DEPTH-CONSISTENCY FLAG
POS	TEXT	2	CHARS POSITION-CONSISTENCY FLAG
TIMF	REAL	1	SERIES END TIME IN HOURS FROM START OF CENTURY
MOORNME	TEXT	5	CHARS MOORING ID
EXPNME	TEXT	5	CHARS EXPERIMENT ID
PROJNME	TEXT	10	CHARS PROJECT NAME
DAYS	REAL	1	LENGTH OF TIME SERIES IN DAYS
DESCRIPT	TEXT	26	CHARS DATA SERIES DESCRIPTER

Index