Hydrobase3

Technical Report

Ruth Curry and Carolina Nobre
Woods Hole Oceanographic Institution
Sep 2013

This work was funded by the National Science Foundation through GRANT OCE-0751896

Hydrobase3

Table of Contents

Table of Figures

Database Structure

SECTION I

About Hydrobase 3

Hydrobase3 is a tool for climatological analysis of hydrographic properties. The package is comprised of software and database products that together provide a flexible means of constructing and analyzing datasets customized to investigators' research needs. The database is comprised of original, quality-controlled hydrographic profiles of pressure, temperature, salinity, oxygen, nutrients and tracers (from CTD, bottle, and float platforms) and 3-dimensional gridded fields of pressure, temperature and salinity including mean and error variance statistics.

The software enables users to navigate through these databases and produce a variety of analysis products (e.g. property plots, vertical sections, surface maps, time series). It can be installed and run on multiple operating systems (LINUX/UNIX, MacOS, Windows OS, Matlab), utilizing the power of shell environments to extract and analyze hydrographic data as a series of flexible commands. Installation on your local platform requires an **ANSI-C compiler** (such as gcc or MinGW) plus the **netCDF libraries** (Rew et al., 1997; UNIDATA website).

Download and Build

Where to Install

The recommended directory structure is:

HB3/src – contains all source code and a makefile
HB3/lib **-** topography, neutral density, country and ship code tables, covariance parameters.
HB3/lists –files with sigma levels and standard depths for creating gridded climatologies.
HB3/bin – place to install executables.
HB3/doc - printable technical report (this document) and a tutorial.

What is needed

1)**netCDF** (version 3.6 or later)
2)**HydroBase3 source distribution** (instructions for installing *HydroBase3* are detailed below)
3)**MinGW** (for installing on Windows Platforms)

How to Install

1) You need the netCDF C libraries version 3.6 or later

A tar distribution from UCAR is included on this ftp site.
Or go to web site: http://www.unidata.ucar.edu/packages/netcdf/index.html
untar and build **netcdf-#.#.#.tar.gz**
there are README and INSTALL files in this UCAR distribution

2) Create a directory for installing *HydroBase3* (i.e.   mkdir /usr/local/HB3)
This will be referred to as ***$HB_HOME***
cd *$HB_HOME*/src
copy **makefile.dist** to **makefile**
edit **makefile** with appropriate values for the following:

```
*************************************
These lines must be edited:
*************************************
INCLUDE= .                                  # location of HydroBase3 *.h files
NETCDFLIB = /usr/local/netcdf-3.6/lib   # location of netcdf libraries locally
NETCDFINC = /usr/local/netcdf-3.6/include   # location of netcdf.h locally
BINDIR = ../bin                              # location of HB3 executables
CC= gcc                                      # invoke your C compiler
CFLAGS=-c -I$(INCLUDE)                      # compile, include
LINK= gcc
LDFLAGS =                                    # any linker flags
LIBS= -lm -lc           # math, c libs          # if you get link errors, try omitting –lc
*************************************
```

3)Edit **hb_paths.h** with local directories to the *$HB_HOME /lib* and */lists* subdirectories where the topography data and other necessary files are stored.

4) create the directory to store executables if it does not exist (see BINDIR above)

5) compile and install:
Type the following commands:

make hydrobase
make install
make clean

to add OPTIONAL modules:

make other
make install_other
make clean_all

File Types and Formats

The *HydroBase3* database utilizes four types of files:
Observed Profile Data (quality-controlled) in **ascii text** files
Gridded Climatology Products: monthly and annual objectively mapped fields of pressure, temperature, salinity at various resolutions in **netCDF format**
Topography (ETOPO1) at 1 minute resolution in **netCDF format**
Global parameters file specifies spatial correlation functions and length scales in **netCDF format**.

Hydrobase3 Station Format

Observed profile data are stored as ascii text and are therefore straightforward to read. A file can contain any number and type of profiles. Each individual profile stores metadata in two header lines and any number of properties, measured or derived (see appendix for complete description of the structure of a *HydroBase3* Station Format file).

Hydrobase3 Gridded Data

Gridded files are stored in netCDF format which is platform independent and self-describing. Observed properties are gridded in latitude, longitude and depth dimensions using Gauss-Markov optimal interpolation methods. The objective mapping is implemented along isopycnal surfaces and the resulting profiles are interpolated back onto prescribed depths. Isopycnal methods more faithfully reproduce observed property-property relationships compared to mapping along depth or pressure surfaces (e.g. Lozier et al., 1994; Chang & Chao, 2000). A set of covariance function parameters and length scales, which have been evaluated as a function of depth and geographic location, are used in the optimal interpolation. The file of global parameters, *lib/global_oi_parms.nc*, is supplied in netCDF format as part of *HydroBase3*. A topography database (ETOPO1) provides bathymetry; properties are evaluated along the seafloor in addition to the set of standard depths. Grid points are assigned a mask value (+9.0e35) where there is no ocean, to differentiate them from missing values (-9.0e35). This feature is exploited in the optimal interpolation process to prevent mixing water masses across topographic boundaries and barriers. Gridded properties include the following statistics:

- Mean value
- Number of observations
- Square root of the error variance
- x- and y-length scales employed in the optimal interpolation at each grid node

A complete description of the structure of the *HydroBase3* netCDF file format is provided in the appendix. Details of the optimal interpolation techniques are described in section III.

Hydrobase3 Topography Files

HydroBase3 includes the ETOPO1 (Amante and Eakins, 2009) global topography database gridded at 2 resolutions: 1 arc–minute and 0.1-degree. The former is distributed by the National Geophysical Data Center (http://www.ngdc.noaa.gov/mgg/global/global.html) and utilizes a netCDF format that is compatible with the GMT plotting package (http://gmt.soest.hawaii.edu). The lower resolution version is a subset of this product.

HydroBase3 uses topography in the following ways:

- to define the shape of ocean basins in constructing 3D gridded property fields
- to mask bottom topography in vertical sections
- to add bathymetry to horizontal surface plots

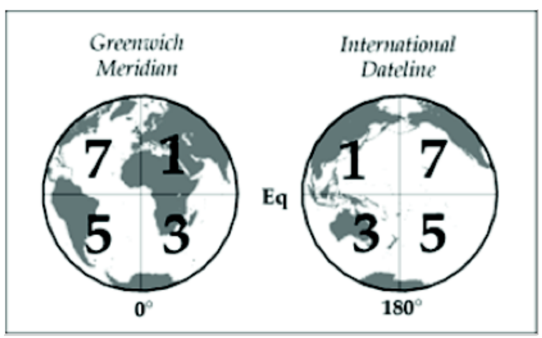
Data Sources and Organization

The global set of observed profile data were collated from several principal sources and converted to *HydroBase3* **station format**: *World Ocean Database*, *WOCE Hydrographic Programme*, *CLIVAR & Carbon Hydrographic Data Office (CCHDO)*, the *Argo* program and those obtained directly from the scientists who collected the data (e.g. MacDonald et al., 2001; Kobayashi and Suga, 2006). The profiles include original (not standard depth) observations of pressure, temperature, salinity, oxygen, nutrients, tracers, and carbon properties to which quality control procedures have been applied.

Observed Profile Data

The observed profiles are sorted geographically by 10-degree World Meteorological Organization (WMO) Squares (Fig. 1) and stored in files named *ms10.extent*, where *ms10* is the 4-digit WMO square designation and *extent* describes the type of data (e.g. *.ctd*, *.btl* or *.flt*). *Examples:* 7405.ctd, 1513.btl, 3402.flt

WMO Square is an arithmetic formulation...



WMO Square = *hem* * 1000 + (*lat*/10) * 100 + (*lon*/10)
where *hem* =
1 or 7 for Northern hemisphere
3 or 5 for Southern hemisphere
1 or 3 for Eastern hemisphere
5 or 7 for Western hemisphere
lat = (integer) latitude, *lon* = (integer) longitude

Figure 1 – Definition of World Meteorological (WMO) Squares

The WOCE and CLIVAR synoptic section data are included as individual *HydroBase3* files, organized by the WHPO section designation and year of occupation (e.g. **P6e_2010**). Software modules (**woce_ctd2hb**, **woce_btl2hb**, **wocex_ctd2hb** and **wocex_btl2hb**) readily convert either the original WOCE format or WOCE exchange format into *HydroBase3* station files. Similarly, **wod_convert** will import *World Ocean Database* observed level or standard level files, while **argo_convert** will import Argo float profiles.

Gridded Climatology Products

The gridded climatology products are distributed as netCDF files and are organized by ocean basin (e.g. Atlantic, Indian, Pacific, Arctic). The file name is intended to be self-describing and reflects geographic area, grid resolution, time domain, and the module that produced it. File names include some combination of:

Region: Atlantic, Pacific, Indian, Arctic
Resolution: .1deg .halfdeg .quintdeg
Type: .oi3d .bin3d .FG
Month*:* *.jan* *.feb* *.mar* (etc) or *.clim* (for annual averages)

For example, *Pacific.quintdeg.oi3d.mar.nc* contains 0.2-degree gridded fields of pressure, temperature and salinity in the Pacific Ocean for the month of March produced by optimal interpolation (**hb_ncoi3d**). Pacific.1deg.FG.mar.nc is the 1-degree gridded "first-guess" fields supplied as input to **hb_ncoi3d**.

***.bin3d.*.nc** : files produced by hb_bin3d—the preprocessing step for interpolation modules
***.FG.*.nc** : "First Guess" files produced by near-neighbor interpolation and smoothing along isopycnal surfaces (**hb_ncfg3d, hb_smooth3d**).
***.oi3d.*.nc** : the gridded fields produced by optimal interpolation

Software Modules

SECTION II

General Usage

The software modules perform various tasks on the *HydroBase3* database, for example extracting profiles, computing properties, generating sections, maps or 3-D gridded fields. Some tasks require running a sequence of modules in which the output from one is used as input to another. A few rules govern the syntax of all *HydroBase3* modules:

A usage statement is displayed by typing the name of the module with ***–h*** as an argument
Input files are always listed as the first arguments – i.e. immediately following the module name
The order of arguments – except for the list of input files – is not significant.
The structure of an argument is generally: **-O***value1/value2/value3* where a dash precedes the uppercase option letter, followed by an appropriate set of value(s), which are separated by slashes.
No white space is permitted between any parts of a single argument.
Some arguments are required, others are optional.
In this document, **optional arguments** are denoted by square brackets: i.e. [-O*value1/value2*]
while **required arguments** will be shown without brackets.

hb_extract 1106.btl 1106.ctd -D/HB3/Data -Tg64/66/12/13 > profiles.sta

Module Name | Input Files | Arguments | Output File

Figure 2 – Structure of a Hydrobase3 command

Some arguments are common to (most) *HydroBase3* modules:
[-D] for input directory
[-E] for filename extent (multiple –E arguments are permitted)
[-O] for output file

Many of the modules output results to the screen (stdout device) which can be redirected to a file using the **'>'** operator. Status messages are output to the screen (*stderr* device) and can be redirected to a logfile using the **'>&'** operator.

Properties Supported by HydroBase3

Individual properties are specified by a 2- or 3-character string mnemonic (bold-faced type below). Note the difference in temperature variables (e.g. **te** vs. **t90** and **th** vs. **th9**) and oxygen variables (**o2** vs. **ox**). An **N** in front of the property mnemonic designates number of observations for that property, **V** is for square root of the error variance, and **Q** is for quality flag.

pr : pressure [dbars]
de : depth [meters]
te : temperature IPTS-68 [degrees C]
th : potential temperature IPTS-68: pref= 0. [degrees C]
t90 : temperature ITS-90 [degrees C]
th9 : potential temperature ITS-90: pref= 0. [degrees C]
sa : salinity [psu]
ox : oxygen [ml/liter]
o2 : oxygen [micromole/kg]
n2 : nitrite [micromole/kg]
n3 : nitrate [micromole/kg]
p4 : phosphate [micromole/kg]
si : silicate [micromole/kg]
ht : dynamic height [dyn. meters (= *10 m**2/s**2)]
pe : potential energy anomaly [10**6 g/sec**2 (ergs/cm**2)]
s0 : potential density: pref = 0. [kg/m**3]
s1 : potential density: pref = 1000. [kg/m**3]
s2 : potential density: pref = 2000. [kg/m**3]
s3 : potential density: pref = 3000. [kg/m**3]
s4 : potential density: pref = 4000. [kg/m**3]
s_ : potential density: pref = ?. [kg/m**3]
bf : buoyancy frequency [* 1.e-5 radians/sec]
pv : potential vorticity [* 1.e-12 m^-1 sec^-1]
sv : specific volume [* 1.e-8 m**3/kg]
va : specific volume anomaly [* 1.e-8 m**3/kg]
f1 : cfc-11 [picomole/kg]
f2 : cfc-12 [picomole/kg]
f3 : cfc-113 [picomole/kg]
he : helium [micromole/kg]
tu : tritium [tritium units]
gn : neutral density (gamma-n) [kg/m**3]
ge : gamma-n errorbar [kg/m**3]
vn : velocity north [m/sec]
ve : velocity east [m/sec]
vs : sound velocity [m/sec]
dr : density ratio []
al : thermal expansion [10**7 alpha]
be : haline contraction [10**7 beta]
ccl : carbon tetrachloride [picomole/kg]
sf6 : sulfur hexafluoride [femtomoles/kg]
co2 : total carbon dioxide [micromole/kg]
alk : total alkalinity [micromole/kg]
doc : dissolved organic carbon [micromole/kg]
tdn : total dissolved nitrogen [micromole/kg]

c13 : radiocarbon-13 [/mille]
c14 : radiocarbon-14 [/mille]

Quality Flags:

0 = no quality assessment
1 = good measurement
2 = probably good measurement
3 = probably bad data point
4 = bad data point
6 = interpolated data point
9 = missing data

Although *HydroBase3* supports the use of quality flags, they are not presently being utilized in the main profile database.

hb_extract, hb_propcalc, hb_xyprop, hb_xyzprop, hb_mssort, hb_stationsort, hb_siftsta, hb_siftlevs, hb_columns, hb_getpos

Working with Profile Data

Extracting	Properties	Sorting	Positions
hb_extract	hb_propcalc	hb_mssort	hb_getpos
hb_siftsta	hb_xyprop	hb_stationsort	
hb_siftlevs	hb_xyzprop		
	hb_columns		

Input		Module		Output
HB3 Station Format Files	➤	hb_extract hb_stationsort hb_mssort hb_siftsta hb_siftlevs hb_propcalc	➤	HB3 Station Format Files
HB3 Station Format Files	➤	hb_xyprop hb_xyzprop hb_columns hb_getpos	➤	Columns of Ascii Text

Figure 3 –Modules for working with *HydroBase3* station files

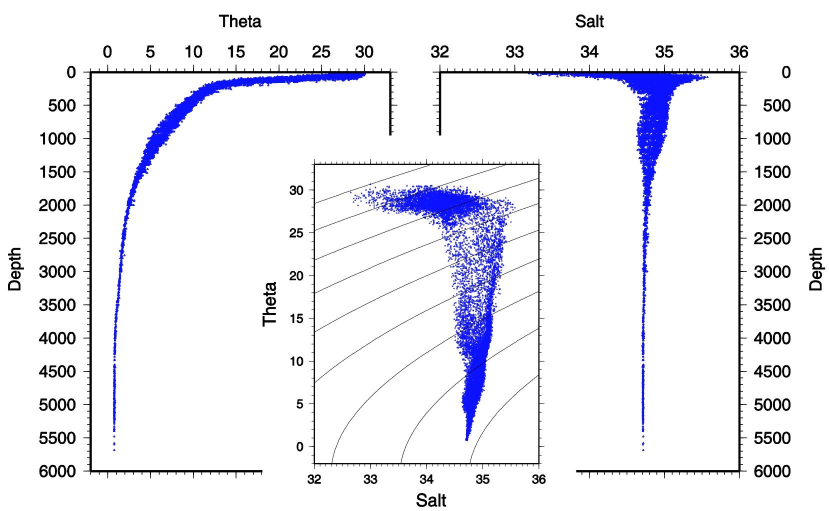


Figure 4 – Example of profiles of potential temperature and salinity for all bottle data in WMO square 3009 (eastern equatorial Indian Ocean). Potential density contours have been added to the T-S diagram.

Extracting Profiles

hb_extract searches for and retrieves profiles that meet specified criteria.

> **hb_extract** 7403 7404 7405 7503 7504 7505 **–D**./data **–E**.btl **–E**.ctd **–Tg**-55/-35/40/55 **-Ty**1900/1980 **–R**grandbanks.post1980.sta > grandbanks.pre1980.sta

extracts profiles from the list of input files (7403.btl 7403.ctd, …etc) located in the directory *./data*. The profiles before and including 1980 that fall within the specified geographic bounds will be written to the file *grandbanks.pre1980.sta* and any rejected profiles will be written to *grandbanks.post1980.sta. [*Refer to the Appendix for complete usage descriptions of all modules.]

hb_siftsta and **hb_siftlevs** search for and list profiles that meet user-specified criteria. They are similar, but **hb_siftsta** outputs entire profiles while **hb_siftlevs** outputs selected levels from a profile. Each determines whether an observed property (specified with **–P**), falls within the user-defined range. The criteria can additionally include specified temperature, pressure, or density ranges, specified with **–T**, **–Z**, and/or **–S**. Observed levels (or profiles) that meet these criteria are written to the screen – or can be redirected to an output file specified by **–O**. Scans/profiles that do not meet the criteria are written to a reject file specified by **–R**.

> **hb_siftlevs** 7700_A.nodc.btl.deep.rchk **–P**sa/34.93/34.945 **–T**-1/0.02 **–Z**1000/4000 **–R**7700_A.nodc.btl.deep.goodlevs **–O**7700_A.badlevs

finds all observation levels that meet the following criteria: salinity in the range 34.93 – 34.945, potential temperature in the range -1.0 - +0.02C, and pressure between 1000-4000 db. The levels will be output to the file **7700_A.badlevs** in *HydroBase3* station format. All other levels will be output to the file **7700_A.nodc.btl.deep.goodlevs**. Both **7700_A.badlevs** and **7700_A.nodc.btl.deep.goodlevs** can be read and used with other *HydroBase3* utilities.

Computing Properties

hb_propcalc computes hydrographic properties at each level in a *HydroBase3* station file. For each input station, a profile is output that contains the properties requested – to the extent that they are available. Requested properties that are not present in the input, but can be computed from available properties, will be output.

> **hb_propcalc** 7203.btl **–D**./data **–P**pr/t90/th9/sa/s3/s4/p4/f1/f2 > 7203.btl.prop

reads each profile in the *HydroBase3* file ./data/7203.btl and outputs a file in which each profile lists pressure, temperature, potential temperature, salinity, sigma-3000, sigma-4000, plus phosphate, freon-11, freon-12, if present. Output is directed to *7203.btl.prop*.

hb_xyprop creates a listing of property pairs specified by their character mnemonics with **–X** and **–Y**. Use the **–M** option to separate profiles with a '**>**' character (or append a character of choice to **–M**). Use the **–Z** option to delimit a depth range for output properties.

hb_xyzprop creates a listing of x,y,z property triplets. As with **hb_xyprop**, use the **–M** option to separate profiles with a '**>**' character.

Example:

hb_xyprop 7306 7307 **–D**/d4/hbase/natl/Monthly **–E**.jan.flt **–X**sa **–Y**pr **–M** **–Z**0/1000 **–O**subtrop.jan.sapr

outputs salinity/pressure pairs between 0 and 1000 meters for all profiles from files *7306.jan.flt* and *7307.jan.flt* in the directory */d4/hbase/natl/Monthly*. The individual profiles will be separated by a line containing '>' (which GMT interprets as a segment divider). An alternate character can be specified, e.g. **–M**% for the Matlab separator.

hb_tssig pref s_min s_max s_incr t_min t_max t_incr > tssig0.xyz

creates a file of salinity, potential temperature, potential density triplets for adding density contours to a T-S plot.

hb_columns was designed to create files that can be easily imported into *Matlab*. In the output listing, each row is an observation level and the columns contain either a property or information such as year, month, or position.

hb_columns station_s.1965.btl **–Y** **–M** **–P**pr/t90/sa/pv **–W**100/10 **–O**station_s.1965.dat

creates a file with year in the first column, month in the second, followed by pressure, *in situ* temperature, salinity, and potential vorticity respectively. **–W** specifies the length of a vertical window over which potential vorticity is computed.

Listing Station Positions

hb_getpos lists longitude/latitude position of profiles.

hb_getpos 7306.btl 7306.ctd 7306.flt **–D**/d4/hbase/natl **–B**-66/-62/31/35 **-O**BATS.stapos

extracts all station positions in the geographic area bounded by 66-62°W, 31-35°N and outputs longitude, latitude to the file *BATS.stapos*.

Sorting profiles

hb_mssort will read and sort any number of files by geographic position into files representing various sizes of meteorological squares. A new file is opened each time a station is encountered which does not belong in a currently open file. A summary of station counts is displayed on the screen (*stderr* device) at the end of the sort. The naming convention of the output files is determined by the size of the meteorological square:

10° files are named *msq10.extent* where *msq10* is the 4-digit WMO square designation and *extent* is user specified. For smaller areas, the files are named *ms10_##.extent.*

	values of ##
5° files (4 per 10° square) :	_0 _1 _2 _3
2.5° files (16 per 10° square) :	_0h _1h _2h _3h…. _9h _Ah _Bh _Ch _Dh _Eh _Fh
1° files (100 per 10° square) :	_00…. _99

Example:

hb_mssort 7103 **–E**.btl **–E**.ctd **–E**.flt **–S**5 **–N**.all **–T** **–O**/d0/data

sorts ctd, bottle and float profiles into 5° squares and saves the output files (*7103_0.all, 7103_1.all, 7103_2.all* and *7103_3.all*) to directory */d0/data*, truncating (overwriting) existing files of the same name.

hb_stationsort sorts *HydroBase3* profiles into individual files. If *bats.1973.sta* contains 15 profiles (station #1,2,3,....15), then
Example:

hb_stationsort *bats.1973.sta*

will generate 15 files named (by default) according to country code, ship, cruise and station number: e.g. 310G3456.0001, 310G3456.0002, etc.

The output file name can be set with **–N**, to which the 4-digit profile number will be added as a suffix:
Example:

hb_stationsort bats.1973.sta **–N**bats_1973

will generate 15 files named *bats_1973.0001*, *bats_1973.0002*, etc…

Interpolating pressure series

hb_prseries reads a station format file and outputs each profile with observations interpolated onto a specified pressure interval.
Example:

hb_prseries 7013.btl **-D**/d1/HB3/Data **–P**10 **–O**7013.10db.btl

interpolates the bottle profiles onto 10 db levels.

Working with Sections

hb_ncslice, hb_section, hb_gridsection, hb_toposlice, hb_getbottom

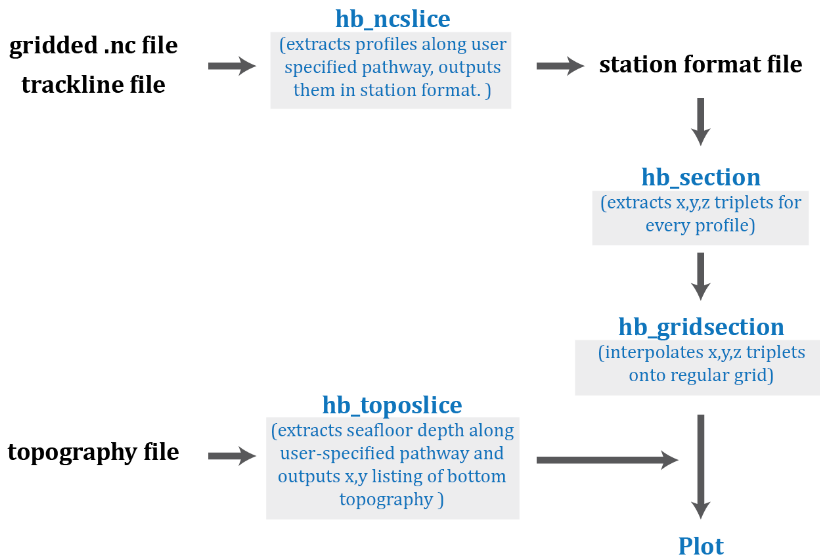


Figure 5 – HydroBase3 modules used to create vertical property sections.

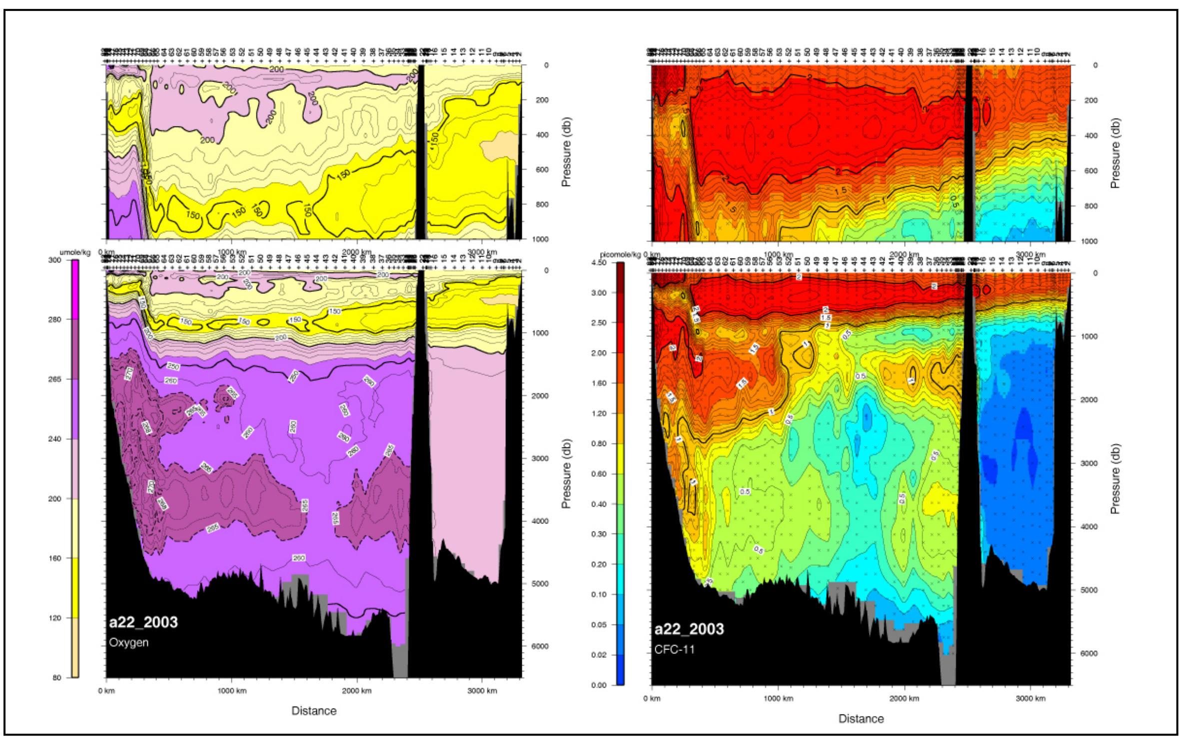


Figure 6 – Vertical sections of oxygen and CFC-11 generated from a synoptic set of stations (WOCE Line A22 – 66°W in Atlantic).

Extracting Sections from Gridded Climatologies

hb_ncslice slices vertically through a gridded climatology along a pathway specified as a list of longitude/latitude pairs. It outputs a *HydroBase3* **station format** file in which "stations" are profiles positioned at the intersection of the pathway and the grid mesh. Each time the pathway crosses a latitude or longitude of the grid, a profile is computed from the two grid nodes that bracket that position along the latitude or longitude. Note that the resulting profiles will be irregularly spaced if the pathway is at an angle to the gridded mesh.

Example:

> **hb_ncslice** Atlantic **–D**/d1/HB3/Gridded **–E**.1deg.oi3d.clim.nc
> **–P**pr/de/t90/sa **–I**1 <<ENDLIST > MBN.1deg.sta
> -81 26
> -64 32
> -53 47
> ENDLIST

generates a *HydroBase3* station file containing profiles from the gridded climatology that lie along a pathway from Miami to Bermuda to Newfoundland (the longitude, latitude pairs specified above). Note that these could be put into a file and supplied to the module with **-L**). The intersection of this pathway with the 1-degree grid in the file */d1/HB3/Gridded/Atlantic.1deg.oi3d.clim.nc* produces irregularly spaced output profiles. Use **hb_section** followed by **hb_gridsection** to create regularly spaced gridded data suitable for contouring. Use **hb_toposlice** to construct a masking file of bottom topography along this pathway.

Sections from Observed Profiles

Sections may also be plotted from observed profiles (e.g. a set of stations from a cruise). *HydroBase3* distributes a collection of WOCE and CLIVAR Repeat Hydrography Sections in **Station Format** files as part of its database.

Generating an XYZ section file

hb_section computes x, y, z values to represent a vertical property section. The **–X** options include **la**titude, **lo**ngitude, or **di**stance. **–Y** and **–Z** can be any property supported by *HydroBase3.*

Example:

> **hb_section** MBN.1deg.sta **–X**la **–Y**de **–Z**th9 **–O**MBN.de_th.xyz

reads the file created by **hb_ncslice** (in the above example) and produces a file of potential temperature as a function of latitude and depth.

Example:

> **hb_section** a22_2003.hb3_ctd **–X**di **–Y**pr **–Z**o2 **-K** **–O**a22_2003.pr_o2.xyz

creates a file of distance, pressure, oxygen triplets, using kilometers (**-K**) for distance units from a synoptic cruise, in this example, along WOCE/CLIVAR line A22 (Fig. 6).

Interpolating onto a regular grid

hb_gridsection interpolates vertical section data onto a regular grid and optionally applies a smoothing filter. The module takes a file of xyz triplets as input (regularly or irregularly spaced) and projects the z values onto the x-y grid described by **–B<***bounds***>** and **–I<***dx/dy***>** parameters. Bottom depths along the section can be constrained (to avoid extrapolation in the gridding) using the output of **hb_getbottom** or **hb_toposlice** .

Example:

hb_gridsection MBN.de_th.xyz **–O**MBN.de_th.smooth **–I**0.2/10
–B23/45/0/6000 **–L**10/10 **–M**MBN.topomask **–S**10/5

interpolates the output from **hb_section** onto a grid with 0.2 degree horizontal resolution and 10 meter vertical resolution with x-limits 23° – 45° latitude, and y-limits 0-6000 meters depth. The search ellipse is set to 10 points in each direction (~2 degrees horizontal, 100 meters vertical). A topography mask (generated with **hb_toposlice**) is used to define the seafloor at each gridnode. A gaussian smoothing filter is applied with horizontal scales of 10 grid points (~2 degrees) and vertical scale of 5 grid points (~50 meters).

Example:

hb_gridsection a22_2003.pr_o2.xyz **–O**a22_2003.pr_o2.smooth **–I**5/50
–B0/3300/0/6500 **–L**20/4 **–M**a22_2003.pr.btm_mask **–S**5/3

creates an xyz file of distance/pressure/oxygen values along A22 (created in the **hb_section** example above). The synoptic input values are unevenly spaced, and the output will be gridded at 5 km intervals in the x-direction and 50 db in the vertical (**-I**). The search ellipse (**-L**) will be 20 gridpoints in the x-direction (~100 km) and 4 gridpoints in the vertical (~200 db). A smoothing filter [**-S**] with radius of 5 gridnodes (~25 km) in the x-direction and 3 gridnodes (~150 db) in the y-direction is applied.

Generating bottom topography or mask

hb_getbottom finds the deepest observation level for the specified z-property and lists corresponding x, y values. It is used to create a bottom mask to limit interpolation at the bottom of each profile (e.g. avoids extrapolation of property values beneath the deepest observation).
Example:

hb_getbottom MBN.sta –**X**di **–Y**de **-Z**th9 **–K** **-M** > MBN.bottom_mask

extracts the depth of the deepest temperature observation as a function of cumulative distance for the profiles in *MBN.sta*. It outputs columns of distance and depth for each profile, with 2 lines inserted at the beginning and end to create a masking polygon recognized by **hb_gridsection**.

hb_toposlice extracts bathymetry values along a pathway specified as a list of longitude/latitude pairs. It is a companion to the module **hb_ncslice**, which extracts profiles along a pathway to construct a vertical section. The output from **hb_toposlice** can be used as a mask in **hb_gridsection** and for overplotting bathymetry on a section.
Example:

hb_toposlice **–D –K** **-M –O**MBN.topomask <<ENDLIST
-81 26
-64 32
ENDLIST

Working with Surfaces

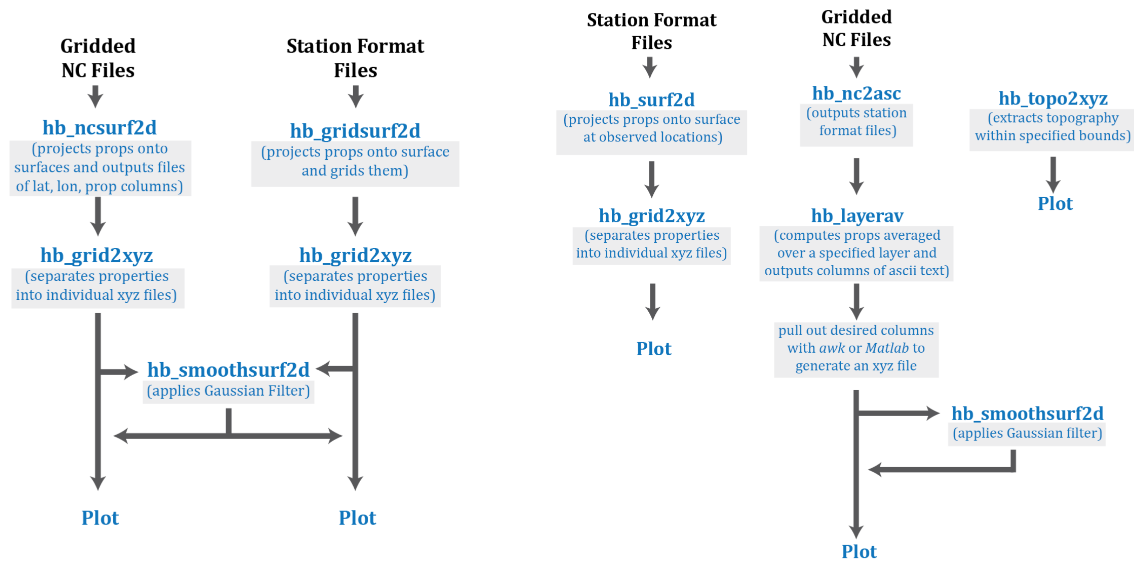


Figure 7 – Hydrobase3 modules for generating surfaces.

Surfaces from Gridded Climatologies

hb_ncsurf2d extracts properties from gridded climatologies projected onto one or more horizontal surfaces. These surfaces can be specified in a file or supplied interactively (the module prompts for input if no file is supplied) in the following way:

surf.list :

```
s0     26.80   sig2680.oct.grid
s2     36.80   sig3680.oct.grid
bot    bottom.oct.grid
end
```

For each surface a property, value, and output filename are required. The string "**bot**" is a special option to extract properties along the seafloor. The list of surfaces is terminated by the string "**end**".

Example:

hb_ncsurf2d Atlantic **-D**/d1/HB3/Data **–E**.1deg.oi3d.oct.nc **-P**pr/de/th9/sa **-I**1/1 **-S**surf.list

extracts pressure, depth, potential temperature, and salinity values from the gridded file named */d1/HB3/Data/Atlantic.1deg.oi3d.oct.nc* along surfaces specified in the file *surf.list*.

Surfaces from Observed Profiles

hb_gridsurf2d and **hb_surf2d** project properties onto horizontal surfaces in a similar fashion. **hb_layerav** averages properties over a vertical layer.

hb_gridsurf2d computes mean and standard deviation values for properties projected onto gridded horizontal surfaces. The grid spacing and bounds are specified with **–B** and **-I**. All points within a grid square are used to compute the mean and are weighted equally.
Example:

> **hb_gridsurf2d** 7105 7106 7205 7206 **–B**-70/-55/10/30 **–I**.5 **–S**surf.list **–P**pr/th9/sa/o2 **–D**./data **–E**.ctd **–E**.btl

will project pressure, potential temperature, salinity, and oxygen onto the surfaces listed in *surf.list* for the area spanning 70-55°W, 10-30°N. The output files are also specified in *surf.list*. The output grid will have 0.5-degree spacing in the x and y directions.

hb_surf2d creates column listings of properties and other information pertaining to the individual profiles. Specified properties are projected onto one or more surfaces, similar to **hb_gridsurf2d**, but the output values are not gridded. One or more output options may be specified: year, month, latitude, longitude, station id (cruise and station#). Geographical bounds can be optionally specified to delimit the output values.
Example:

> **hb_surf2d** 7105 7106 7205 7206 **-D**/d1/HB3/Data **–E**.ctd **–E**.btl **-P**pr/th9/sa **-I**0.5 **–S**surf.list

hb_layerav computes properties averaged over a vertical layer defined by two surfaces. Each point contributing to the average is weighted by the pressure range over which it was observed. The module outputs columns of properties.
Example:

> **hb_layerav** NWAtlantic.5_65N.quintdeg.asc **-1**s4/45.89 **-2**s4/45.91**b** **–P**th9/sa **-L** **-H** > 5_65N.sig4589.layerav

will compute potential temperature and salinity averaged for the sigma-4000 density layer 45.89 – 45.91 (or 45.89 – bottom if the layer grounds out). The output will include latitude, longitude, and thickness (H) of the layer in addition to average properties.

Creating XYZ files for surfaces

hb_grid2xyz separates a *HydroBase3 .grid* file (output by **hb_gridsurf2d, hb_ncsurf2d, hb_surf2d**) into individual property files containing longitude, latitude, property triplets for plotting.

The program outputs up to 4 files for each property specified:

1) property info: lon, lat, prop [, n]
2) blank gridpts: lon, lat
3) error info : lon, lat, error [, n]
4) error blanks: lon, lat

Error information is output if the **-E** option is specified.
Blanking files are output only if the **-B** option is specified.
Number of observations is included in the above output if the **-N** option is specified.

Output files are named:

<root><property_id>.xyz
<root><property_id>.blk
*<root><property_id>*_err.xyz
*<root><property_id>*_err.blk

where *root* is supplied by the **-O** option and *property_id* is the same as the property_list_ids in the **-P** option.

Example:

hb_grid2xyz sig2680.oct.grid **–O**sig2680.oct. **–P**pr/th9/sa **-B**

will decompose *sig2680.oct.grid* into 3 separate .xyz files plus 3 files listing grid squares with no data (blanks):

sig2680.oct.pr.xyz	sig2680.oct.pr.blk
sig2680.oct.th9.xyz	sig2680.oct.th9.blk
sig2680.oct.sa.xyz	sig2680.oct.sa.blk

Smoothing property fields along a surface

hb_smoothsurf2d smooths a 2D gridded surface using a distance weighted Gaussian filter with variable x-, y-radii, and outputs longitude, latitude, property triplets. Radii for the smoothing ellipse can be specified as distances (km) or by number of grid points. An optional weighting factor [**-A**] provides additional control over the filter.

Example:

hb_smoothsurf2d sig2680.oct.th9.xyz **–B**-100/0/0/65 **–I**0.5/0.5 **–M**sig2680.oct.th9.blk **–S**100 **-O**sig2680.oct.th9.smooth.xyz

will smooth the horizontal surface of potential temperature (here on sigma-0 = 26.8 kg/m^3) from the above example. Both the xyz input file and the masking file were created by **hb_gridsurf2d**. The masking file contains positions of grid cells to be masked. The grid resolution is 0.5 deg with bounds specified by **–B**. The smoothing radius is 100 km in both x- and y- directions. The smoothed values will be output to *sig2680.oct.th9.smooth.xyz*.

Generating Topography files

hb_topo2xyz outputs longitude, latitude, bathymetry triplets from the *Etopo1* topography data included in the *HydroBase3* package (in the */lib* subdirectory). The default resolution is 0.1 degree. Higher resolution (1-minute) is obtained with the **–H** option. An alternate file can be specified with **–T** if it uses a compatible netcdf format. Use **–B** to specify the bounds of the output area.

Example:

hb_topo2xyz **–B**-100/0/0/65 **-H** **-O**natl.hires_bath.xyz

outputs high resolution (1-minute grid) bathymetry values in the region 0-100°W, 0-65°N to a file named *natl.hires.bath.xyz*.

Working with
3D Gridded Climatologies

hb_bin3d, hb_ncoi3d,
hb_ncfg3d, hb_ncsmooth3d,
hb_nc2asc, hb_ncinfo,
hb_ncmerge

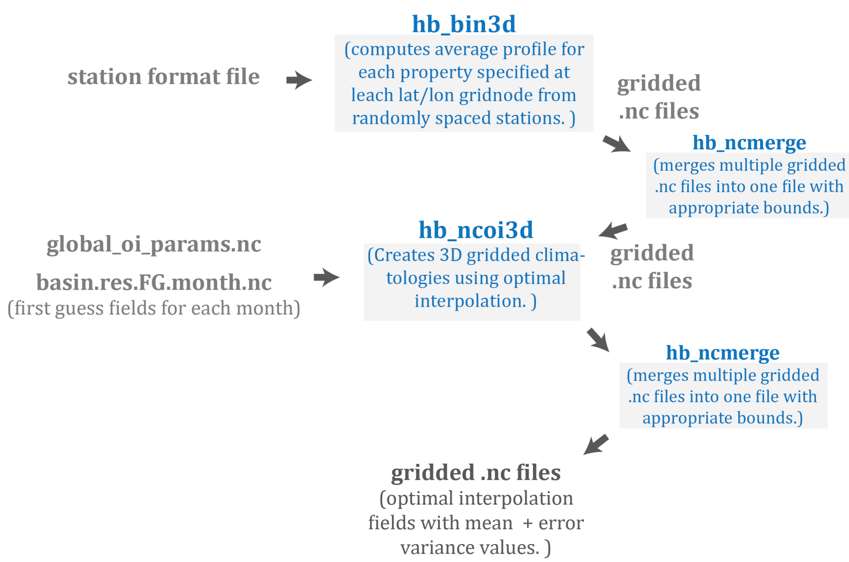


Figure 8 – Hydrobase3 modules for working with 3D gridded climatologies.

Working with Existing Gridded Climatology Files

hb_ncinfo prints information about the contents of a *HydroBase3* netcdf file.

hb_nc2asc converts a *HydroBase3* gridded netcdf file into an ascii *HydroBase3* station file in which each latitude/longitude grid point is represented as a "station" profile.

hb_ncmerge merges multiple *HydroBase3* gridded netcdf files into one file with appropriate bounds and masking. Grid increments and standard depths must be compatible among all files.

hb_ncslice extracts profiles along a user-specified pathway to generate vertical property sections.

hb_ncsurf2d projects properties onto one or more user-specified horizontal surfaces.

Creating Gridded Climatologies

HydroBase3 gridded climatologies include property means, number of observations, error variance statistics and spatial decorrelation scales. These are constructed from observed profiles using optimal interpolation implemented along isopycnal surfaces. Output fields are gridded at user-specified intervals of latitude, longitude and depth in a 2-step process:

1) the observations are binned into small (usually 0.1°) latitude/longitude squares using isopycnal averaging methods (**hb_bin3d**);
2) an optimal interpolation procedure is applied to the binned profiles (**hb_ncoi3d**).

The latter employs a set of modeled covariance functions (specified by semi-variogram parameters in *lib/global_oi_parms.nc*) evaluated for each measured property as a function of latitude, longitude and depth. These covariance parameters – sill-nugget (signal), nugget (noise) and range (decorrelation scale) – have been determined by weighted least squares fits of observed property distributions to an exponential model and were produced by the module **hb_build_oi_parms**. They are stored in a netcdf format file, and provided as part of the basic *HydroBase3* distribution (*/lib/global_oi_parms.nc*). First-guess fields – i.e. 1° gridded mean property fields constructed from observations using a near-neighbor interpolation algorithm and gaussian smoothing along isopycnals (**hb_ncfg3d** and **hb_ncsmooth3d**) – are similarly provided. Further details about all of these are provided in section III.

Preparing profile data for hb_ncoi3d

hb_bin3d reads *HydroBase3* station format files and produces 13 gridded netcdf files – one for each month plus an average for the full year. These reflect observations in the upper ocean seasonal and mixed layers on a month-by-month basis. Below the locally determined mixed layer depth, all observations contribute to the estimated means in all months. The means (and variances) are estimated by a distance weighted average of all observed values within a latitude/longitude bin along isopycnals. A set of potential density surfaces referenced to five pressures with vertical spacing tailored to each 10° WMO square is provided in the */lists* directory. The isopycnally averaged vertical profile for each latitude/longitude bin is then interpolated back onto the set of standard depths that constitutes the vertical coordinate of the output netcdf files.

Because the modules **hb_bin3d** and **hb_ncoi3d** are memory intensive, they are applied to subregions (usually 10° squares) and the resulting output files are subsequently merged into a larger (ocean basin) file. An entire basin can be completed with a simple loop:

Example:

```
set ms10_list = ( 7001 7002 7003 … list of wmo squares for an ocean basin)

foreach ms10 ($ms10_list)
      hb_bin3d ${ms10} –D${indir} –E.ctd –E.flt –B`hb_msq10bounds ${ms10}` -I0.1
     –Ppr/t90/sa –O/d1/HB3/work/${ms10}.quintdeg.bin3d  –Sr${ms10} –L100
     –Z/d1/HB3/lists/stddepths.list
end
```

The above loop will read the CTD and Argo float profile data files for each 10° square (e.g. *7001.ctd*, *7001.flt*) and produce 13 files according to the root name specified with **–O**, to which are appended the month and *.nc*: e.g. *7001.tenthdeg.bin3d.jan.nc*, *7001.tenthdeg.bin3d.feb.nc*, … *7001.tenthdeg.bin3d.clim.nc*. The bounds, **-B**, are supplied with the *HydroBase3* utility

hb_msq10bounds which takes a 4-digit WMO square [*$ms10*] as an argument and returns the longitude/latitude ranges in an appropriate format. Lists of isopycnals appropriate to this WMO square are specified with **–Sr***$ms10***.** The output grid interval [**-I**] is specified here as 0.1 degrees, the list of output depths is provided in the file specified with **-Z**/d1/HB3/lists/stddepths.list. A distance weighting length scale is set to 100 km [**-L**].

The output files are then merged into ocean basin files, one for each month plus the annual average (i.e. *.clim.nc*), by another simple loop:

Example:

foreach month (jan feb mar apr may jun jul aug sep oct nov dec clim)
    hb_ncmerge ${ms10_list} **-D**/d1/HB3/work **–E**.tenthdeg.bin3d.${month}.nc
    –O/d1/HB3/tenthdeg/Atlantic.tenthdeg.bin3d.$month.nc **–P**pr/t90/sa **–L**0
end

Optimal interpolation using hb_ncoi3d

The module **hb_ncoi3d** works in incremental steps (usually 10° square regions) on the merged, monthly files output from the **hb_bin3d/hb_ncmerge** process above. It reads the 13 monthly/annual ocean basin files *(*.bin3d.*.nc*) and outputs 13 monthly/annual files (**.oi3d.*.nc*) for each subregion. Data from areas surrounding the subregion are used in the optimal interpolation. The regional output files are subsequently merged into basin-scale files – again, one for each month plus the annual average. The module can work strictly from the supplied observations, or utilize a set of "first guess" fields and apply the optimal interpolation to residual fields. Note that the grid resolution of the various input/output can be different. The *bin3d* grids are generally at higher resolution (e.g. 0.1°), *first guess* fields are generally 1.0°, and the output grid can be any resolution. The file */lib/global_oi_parms.nc* contains covariance parameters and gradient information required by this module.

As above, a simple loop will accommodate an entire ocean basin:

foreach ms10 ($ms10_list)
    hb_ncoi3d $ms10 –B`hb_msq10bounds $ms10` **-I**0.5 **–P**pr/t90/sa
    -Z/d1/HB3/lists/stddepths.list **–Fr**/d1/HB3/lib/Atlantic.1deg.FG
    –Gr/d1/HB3/tenthdeg/Atlantic.tenthdeg.bin3d
    –Or/d1/HB3/work/$ms10.halfdeg.oi3d
end

The output files are then merged into ocean basin files, one for each month, by another simple loop:

foreach month (jan feb mar apr may jun jul aug sep oct nov dec clim)
    hb_ncmerge ${ms10_list} **-D**/d1/HB3/work **–E**.halfdeg.oi3d.${month}.nc
    –O/d1/HB3/Gridded/Atlantic.halfdeg.oi3d.$month.nc **–P**pr/t90/sa **–L**0
end

Miscellaneous Tools

Here are a few other tools that don't fall within the previous categories.

hb_msq10bounds accepts a 4-digit 10° WMO square and optional increment as arguments and returns geographic bounds in the format : w/e/s/n. Specifying an increment **(-I)** will enlarge the bounds by half the increment which is useful for specifying boundaries of node-grids as opposed to pixel grids.

Example:
hb_msq10bounds 5313 returns: **-140/-130/-40/-30**
while
hb_msq10bounds 5313 **–I**1.0 returns: **-140.5/-129.5/-40.5/-29.5**

---

hb_tssig creates a file of salinity, potential temperature, potential density triplets for adding density contours to a T-S plot.

Example:
hb_tssig 2000 32.0 38.0 0.01 -2 30 0.1 > tssig2.xyz

computes potential density referenced to 2000 db for each 0.01 increment on the salinity range 32.0 to 38.0 and 0.1 increment on the potential temperature range -2 to 30.

Importing other database formats to HydroBase3

Modules that convert profile data from WOCE/CCHDO, World Ocean Database (WOD), and Argo float data into *HydroBase3* station format files are included in the basic software distribution under "Other" utilities (make other install_other in compiling the package).

For WOCE format files**:**
woce_btl2hb a22_2003.hyd **–S**a22_2003.sum > a22_2003.hb3_btl
woce_ctd2hb *$ctd_file_list* **–S**a22_2003.sum **–P**10 > a22_2003.hb3_ctd

For World Ocean Database observed or standard depth files:
wod_convert CTDO7503 -Tc > 7503.wod_noqc.ctd
wod_convert OSD07503 –Tb > 7503.wod_noqc.btl
wod_convert OSDS7503 -Tb > 7503.wod_btl

For World Ocean Atlas conversion to HydroBase3 netcdf gridded format:
woa_convert -I/d1/HB3/woa_data -O/d1/HB3/Gridded/

For Argo float data :

argo_convert infile_list –Oargo.flt –Targo.flt.tmp_only -Rargo.flt.reject –Qargo.flt.no_qc

Importing HydroBase3 data into Matlab:

Cruise Planning Tools:

hb_gettrack computes course heading from point A –> point B and generates positions at user-specified distance intervals along track. The output includes waypoint # (optional), latitude, longitude and heading. It is used to generate segments of cruise tracklines. Append segments into one file to produce a trackline file for an entire cruise.

hb_gettrack -A17.734/-66.0 -B17.63/-66 –D30 –K -S1 > a22.trackline

Given a list of waypoint positions, **hb_distdepth** determines the seafloor depth and distance between points. It outputs a list of waypoint # (optional), latitude, longitude, heading, depth, distance, cumulative distance (in either km or nautical miles).

hb_distdepth –Ia22.trackline –K –Mo –S1 > a22.distdep.degmin

The output from **hb_distdepth** can be fed into a script (e.g. *awk* or *Matlab*) which computes a timeline for a cruise. It requires a slope (m) and intercept (b) obtained, for example, from a linear fit of cast time vs seafloor depth from previous WOCE and CLIVAR hydrography cruises.

Example of an awk script for computing a cruise timeline:

```
set infile = a22.distdep.degmin
set m = 0.00062   # slope
set b = 0.8       # intercept
set speed = 11   # of ship
set starthour = 9
set startday = 21

echo "Estimate using speed = $speed knots"
echo "m = $m"
echo "b = " $b

echo "WP#     lat      lon     dep(m) dist(nm)  steam   cast    cumdays  "
echo "-----  --------- ---------- ------ --------  -----   ----    -----  "

awk -v slope=$m -v intercept=$b -v speed=$speed  -v cumh=$starthour -v
stday=$startday '{if (NR > 2) {steam = $10/speed; if (steam < 1.5  && NR > 3 )
steam = 1.5; eta=(cumh+steam)/24; depth = $9; if ($9 > 6200) depth = 6200;
cast=0; if (depth > 0)cast=depth*slope+intercept; cumh=cumh+steam+cast;
cumd=stday+cumh/24; if(cumd >= 31)cumd=cumd-30; {printf "%4d
%2d%5.1f %c %3d%5.1f %c %5d %7.1f %8.1f %6.1f   %5.1f %s \n", $1, $2, $3,
$4, $5, $6,$7, $9, $10, steam, cast,  cumd, $12} } }' $infile
```

HydroBase3 implements a form of Gauss-Markov mapping or ordinary kriging to produce estimates of the mean and error variance fields through inversion of the covariance matrix (e.g. Bretherton et al. 1976; Zimmerman and Stein, Handbook of Spatial Statistics). The estimator at a particular grid point is a weighted mean of surrounding observations within a search ellipse specified by spatially varying x- and y-radii. (Ellipse radii are empirically derived from the magnitude of the zonal and meridional temperature gradients along isopycnal surfaces at each grid point.) The weights used in the optimal interpolation are derived from the covariance function, **C(h)**, represented by a semi-variogram $\boldsymbol{\gamma}(\boldsymbol{h})$, modeled from existing observations as an exponential function of the spatial lag, **h**, and parameter space, $\boldsymbol{\theta}$:

$$\boldsymbol{\gamma}(\boldsymbol{h};\boldsymbol{\theta}) = \boldsymbol{\theta_0} + \boldsymbol{\theta_1}\,[\mathbf{1} - \boldsymbol{exp}(-\boldsymbol{h}/\boldsymbol{\theta_2}\,)\,]$$

Parameterization of the spatially varying covariance functions has been done *a priori* in a series of steps using the module **hb_build_oi_parms** and stored in *global_oi_parms.nc***.** Initial values for the parameters, $\boldsymbol{\theta}$, which represent the noise, signal and decorrelation scale respectively (Fig. 9), were evaluated by weighted least squares fits to the observations at discrete latitude, longitude and depths throughout the global ocean. Signal-to-noise ratios in these estimates were found to exhibit distinct latitude and depth dependence and a general hemispheric symmetry. This was exploited to construct characteristic depth profiles of signal-to-noise ratios for 10° latitude bands in each ocean basin (e.g. Fig. 10) using maximum likelihood estimates to determine a mean signal-to-noise ratio, assuming a normal probability distribution function, at each depth in a latitude band. These profiles were then applied to obtain (by weighted least square minimization) an optimal set of semi-variogram parameters, $\boldsymbol{\theta}$, as a function of latitude, longitude and depth.

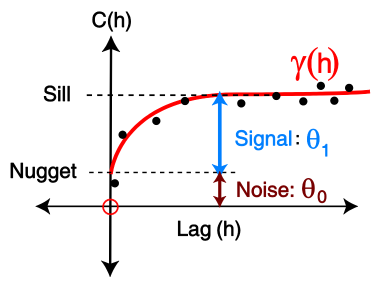


Figure 9. Schematic of the exponential semi-variogram (red curve), modeled from observed covariance as a function of lag (black points). Sill, nugget, signal and noise parameters are diagrammed.

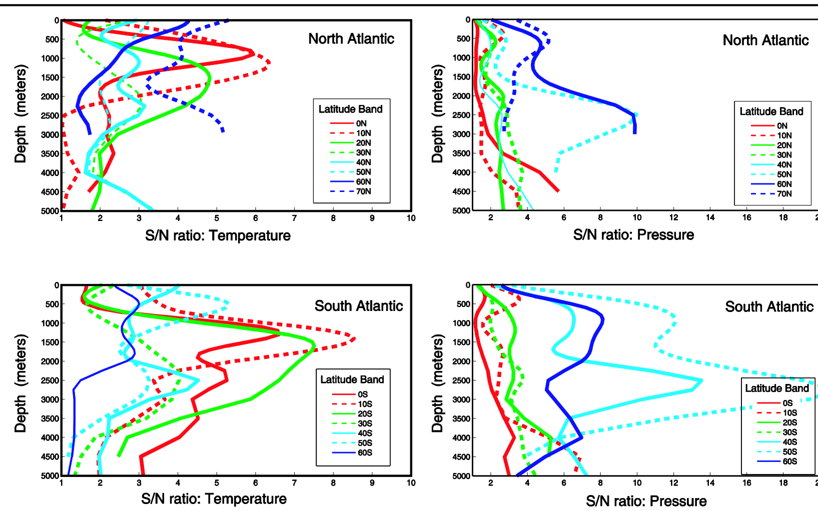


Figure 10. Depth profiles of signal-to-noise ratios averaged in 10° latitude bands for temperature (left panels) and pressure (right panels) along isopycnals. Salinity (not shown) is virtually identical to temperature. The North Atlantic (upper panels) and South Atlantic (lower panels) exhibit roughly symmetric depth and latitudinal dependence.

From the spatial covariance parameters described above, estimates of property means, **Z,** and error variances, $\boldsymbol{\sigma^2}$, are computed using **hb_ncoi3d**. At each gridpoint, **0**, a set of weights, **w**, are derived from the covariance matrix of surrounding observations, **i**, such that they minimize the variance of the estimator, and are further constrained to sum to unity. Once the weights are obtained, estimates of the mean and error variance follow directly:

$$Z_0 = \sum_{i=0}^{n} w_i\, z_i \quad ; \quad \sum w_i = 1;$$

$$\sigma_0^2 = C(0) - \sum_{i=0}^{n} w_i\, C_{i0} - \mu \; ;$$ where μ is the Lagrange parameter

Note that **Z** can be computed either from residuals (bin3d – first-guess values) if first-guess fields are supplied as input to **hb_ncoi3d**, or the actual property values if not. The first-guess, bin3d and oi3d fields can all have different grid dimensions.

The size of the neighborhood or search ellipse around each oi3d grid location, from which bin3d observations are drawn, is empirically set to be inversely proportional to the magnitude of the local zonal and meridional temperature gradient on the isopycnal surface intersecting that point (evaluated by **hb_build_oi_parms** and stored in *lib/global_oi_parms.nc*). If fewer than 30 observations fall within that search ellipse, however, the neighborhood is iteratively expanded until either sufficient observations are encountered – or a maximum radius (e.g. 700 km) is reached. Both parameters (minimum observations and maximum radius) can be optionally prescribed as arguments to **hb_ncoi3d**.

Topographic barriers in OI3d

The ETOPO1 database is used to determine the seafloor depth at each latitude/longitude location. Masked gridpoints (no ocean) are assigned a very large value (+9.0e35) to differentiate them from missing values (-9.0e35) where no estimate can be obtained. At each non-masked oi3d gridpoint, an isopycnal value is determined from either a first-guess field or by a distance-weighted near-neighbor interpolation of surrounding densities at a similar depth in the bin3d fields. The estimator functions above are then applied to surrounding bin3d points at the same isopycnal level within the prescribed search ellipse. If the density surface runs into the seafloor (grounds out) or outcrops at the sea surface, bin3d points at greater distances along the same azimuth (from the central oi3d gridpoint) are excluded from evaluation (i.e. assigned zero weight). This quite effectively prevents mixing water masses across topographic boundaries in these climatologies.

OI3d near the sea surface

A mixed-layer model is implemented to evaluate properties near the sea surface where temporal variations in wind stress, heat, and freshwater fluxes dominate isopycnal processes. A mixed layer is defined for each bin3d profile as the depth range where the potential density (σ_0) is within 0.02 kg m^{-3} of the value at the sea surface (this definition can be optionally specified). Each latitude/longitude gridnode includes 12 monthly mixed-layer structures recording an associated density class, thickness (depth), and uniformly distributed measured properties. The oi3d mixed-layer characteristics (thickness, temperature, salinity, density) for each month are determined from the weighted average (kriging) of surrounding bin3d mixed layers (within the prescribed search ellipse). Properties at levels beneath the mixed layer are determined by kriging along density surfaces that are deeper and denser than the bottom of the mixed layer to avoid creating any static instabilities at the base of the mixed layer.

OI3d near the sea floor

In addition to its array of standard depth levels, *HydroBase3* implements a variable bottom depth level at each grid node reflecting realistic seafloor bathymetry. This more accurately reproduces abyssal water mass properties and gradients which can be lost in a strict standard depth implementation. In the initial binning stage, the deepest/densest observation at each latitude/longitude node is preserved. In the next stage of analysis, **hb_ncoi3d** assesses whether this bottom observation is near enough to the actual seafloor depth to represent the bottom properties. If not, an attempt is made to estimate a more appropriate value from surrounding profiles.

Vertical Data Gaps

Interpolating between vertically spaced observations can be a problem for hydrographic data. Where property gradients are large and varying relative to the vertical spacing, interpolation may produce invalid estimates of the properties between observations. The non-linearity of the equation of state will then introduce significant error to derived properties such as density. No single method, linear or splines of many varieties, produces an adequate solution to missing data in all cases. Under the assumption that *no* data is better than *bad* data, *HydroBase3* checks for and does *not* interpolate over vertical datagaps. In the thermocline (upper 1000 meters) a datagap is defined as >200 meter separation between observations, while below that level, the limit is relaxed to >600 meters. This check applies to all stages of the gridding process – the observations which are interpolated onto isopycnal surfaces – and when interpolating the isopycnally averaged profiles back onto depth levels. A check of vertical gaps in the bin3d and oi3d profiles is made to distinguish pycnostads (weak vertical density gradient) from gaps that result from no data. For pycnostads, property values are estimated by linear interpolation between the top and bottom of that layer.

Quality Control

Observed data are subjected to a battery of checks to identify and remove unrealistic points— which are revealed by viewing property profiles as a function of depth and theta. A combination of statistical and visual methods identifies both systemic problems (malfunctioning equipment) and random errors. "Bad" points are removed and stored in a separate file.

T-S Profile Plots

Groups of profiles are plotted in both pressure and theta space for an initial assessment of the data. If necessary, they are divided into subgroups based on geography or characteristic T-S profiles.

Systemic Problems

Profiles with common deviation (salt shift, for example) are removed. These will often hail from a single cruise. If necessary, the entire cruise is extracted to a separate file and plotted in order to determine whether the entire cruise should be eliminated.

Range Checking

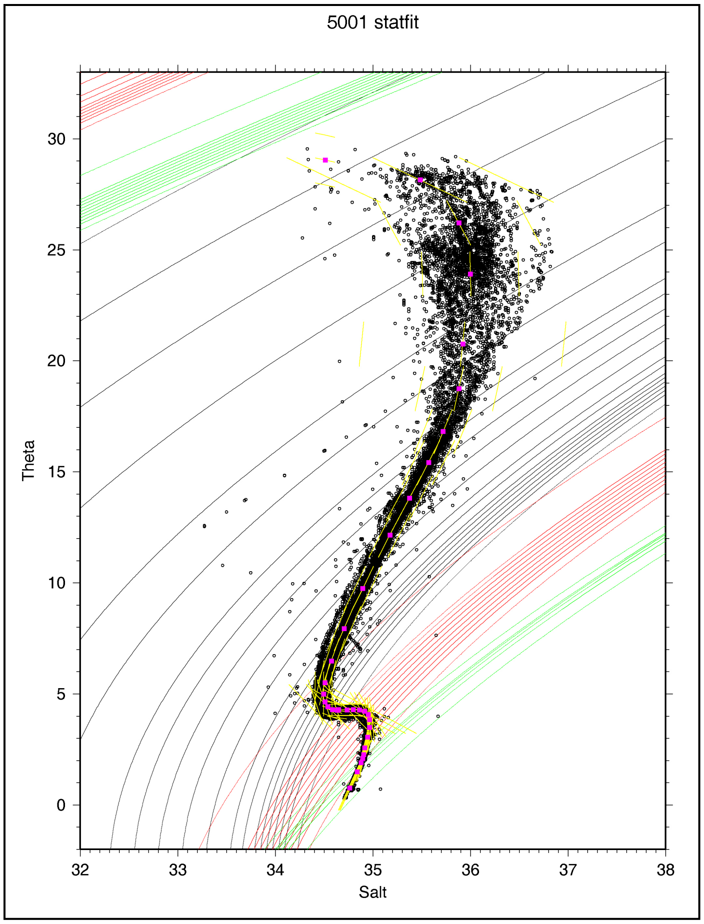
A preliminary sieve of data eliminates values that are obviously extreme. Acceptable ranges are defined for temperature and salinity as a function of depth in a user-supplied file.

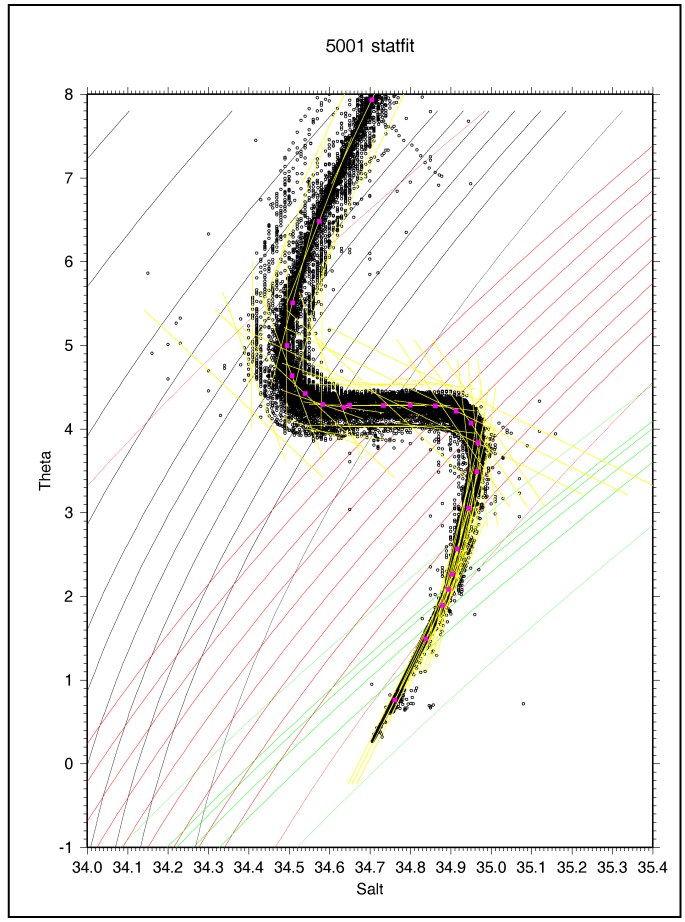
Statistical Fitting and Checking

Here we compute a characteristic T-S curve and standard deviation envelope as a function of density for a group of data. This is then used to identify and remove outliers from the observed data in the next step.

A set of sigma bins appropriate for the area is specified. These sigma bins divide up the T-S curve into segments that together characterize the total T-S relation. For depths 0-1000, sigma-0 densities are used. Sigma-2000 bins are used for depths 1000-3000 m, and sigma-4000 bins are used for depths > 3000m. The statistical checking is done with **hb_statchk_ts** which identifies points that lie outside of the standard deviation envelope specified. The "good" and "bad" points are output to separate files and plotted to get a visual picture of what is being identified.

An example of an typical statistical fit is shown below. On the left is a plot of all the observed T-S points (black) with mean T-S value (pink square) in each sigma-bin which are depicted by the density contours. Black contours are the sigma-0 bins, red contours are sigma-2000 bins, and green are sigma-4000 bins. The yellow lines show the linear T-S relation computed for each bin plus the envelope equivalent to 2-standard deviations away from each linear fit. At right is plotted the deep portion of the T-S curve with expanded axes.





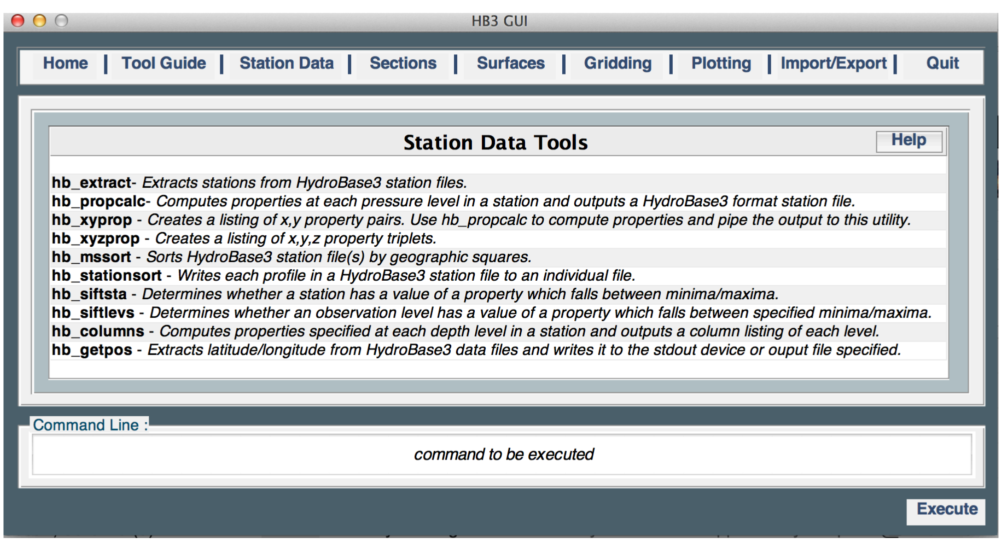
Hydrobase3
Matlab Tools

SECTION IV

Importing HydroBase files directly into Matlab

Matlab GUI

A graphical user interface (GUI) was created to facilitate the usage of *HydroBase3* tools within a *Matlab* environment. The GUI generates *HydroBase3* commands, executes them outside the *Matlab* environment. The results can be imported into *Matlab* for further manipulation and visualization.



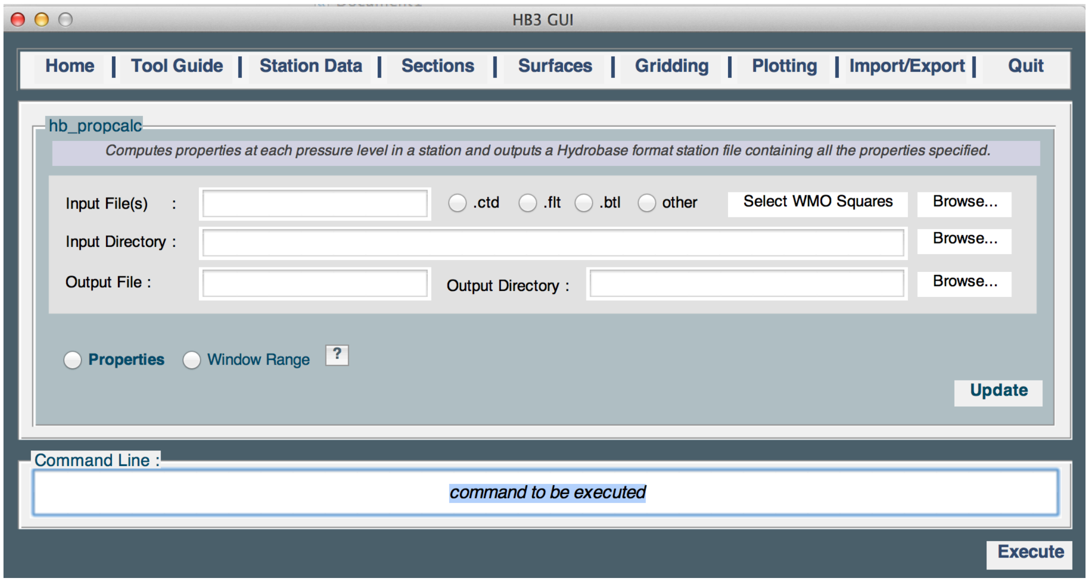


Figure 9 – Screenshots of *HydroBase3* Matlab GUI

Usage

- The *HydroBase3* Matlab GUI was developed in order to make it easier to construct the *HydroBase3* commands and manipulate the database.

- All HB3 tools are sorganized by task categories: Profiles, Sections, Surfaces and Gridding Tools. Once a module is selected the user is presented with a panel containing the required and optional parameters for running a module.

- Once the desired parameters for a given module have been chosen, the “Update” button will construct the HydroBase3 command in the “Command Line” below.

- Finally, the “Execute” button will execute the command outside the Matlab environment generating the output file(s) as specified by the user.

Help

Questions and/or suggestions are welcome and should be sent to rcurry@whoi.edu and cnobre@whoi.edu.

References

Amante, C. and B. W. Eakins, ETOPO1 1 Arc-Minute Global Relief Model: Procedures, Data Sources and Analysis. NOAA Technical Memorandum NESDIS NGDC-24, 19 pp, March 2009.
Macdonald A. M., T. Suga, and R. G. Curry, An isopycnally averaged North Pacific climatology, *Journal of Oceanic and Atmospheric Technology*, 18, 394-420, 2001. .

Orsi, A. H., T. Whitworth III and W. D. Nowlin, Jr., 1995: On the meridional extent and fronts of the Antarctic Circumpolar Current, Deep-Sea Res. I, 42, 641-673.

WOCE Southern Ocean Atlas: http://woceSOatlas.tamu.edu

Rew, R. K., G. P. Davis, S. Emmerson, and H. Davies, **NetCDF User's Guide for C, An Interface for Data Access, Version 3**, April 1997.

UNIDATA website: *http://www.unidata.ucar.edu/software/netcdf*

Structure of a *HydroBase* station file

[Boldface type shows a typical profile, description of fields is in box below]

```
      a  b    c     d   e    f  g hi    j       k       l    m     n   o  p     q
1►   31 RH  1469    7 1951  2 12 1b  30.583 -41.017  3255 0001  18   4 7304  1
2►   pr de te sa
3►      0.0     0.0  19.7800  36.5900
        1.0     1.0  19.7800  36.5900
        9.1     9.0  19.7700  36.6000
       28.2    28.0  19.7500  36.5000
       46.3    46.0  19.7500  36.5800
       92.6    92.0  19.7000  36.6600
      185.2   184.0  18.7900  36.5600
      231.5   230.0  18.0900  36.4700
      277.8   276.0  17.8300  36.4400
      370.5   368.0  16.3300  36.2400
      463.2   460.0  15.2200  36.0800
      556.0   552.0  13.8200  35.9100
      622.5   618.0  12.8600  35.7600
      713.3   708.0  11.6600  35.6200
      804.2   798.0  10.2400  35.3500
      896.1   889.0   8.8000  35.3600
     1836.3  1818.0   3.9800  35.0900
     2323.7  2298.0   3.2300  34.9300
4►   **
```

1 ► Header line containing information about the profile, fields are separated by white space 76 bytes total):

- a: country code (2 char)
- b: ship code (2 char)
- c: cruise number (5-digit)
- d: station number (4-digit)
- e: year (4-digit)
- f: month (2-digit)
- g: day (2-digit)
- h: origination code (1 char)
- i: instrument type (1 char)
- j: latitude (N.positive)
- k: longitude (E positive)
- l: seafloor depth (meters)
- m: quality control status code (4-char)
- n: number of observation levels
- o: number of properties
- p: 10-degree WMO designation (4-digit)
- q: 1-degree WMO designation (2-digit)

2 ► 2-char property IDs show type and order of properties in profile.

3 ► Observed data: one line per depth or pressure level

4 ► End-of-station indicator

Structure of the gridded data files (netCDF) format:

```
netcdf Atlantic.1deg.FG.jun {
dimensions:
        lat = 160 ;
        lon = 140 ;
        de = 85 ;
        time = 1 ;
variables:
        float latitude(lat) ;
                latitude:units = "degrees_north" ;
                latitude:long_name = "Latitude" ;
                latitude:generic_name = "latitude" ;

        float longitude(lon) ;
                longitude:units = "degrees_east" ;
                longitude:long_name = "Longitude" ;
                longitude:generic_name = "longitude" ;

        float de(de) ;
                de:units = "meters" ;
                de:long_name = "Depth (m)" ;
                de:generic_name = "depth" ;

        int minyear(time) ;
                minyear:units = "years" ;

        int maxyear(time) ;
                maxyear:units = "years" ;

        float bottom(time, lat, lon) ;
                bottom:units = "meters" ;
                bottom:long_name = "Bottom Depth" ;

        float pr(time, lat, lon, de) ;
                pr:units = "dbars" ;
                pr:MissingValue = -9.e+034f ;
                pr:_FillValue = -9.e+034f ;
                pr:MaskValue = 9.e+034f ;
                pr:long_name = "pressure" ;

        float t90(time, lat, lon, de) ;
                t90:units = "degrees C" ;
                t90:MissingValue = -9.e+034f ;
                t90:_FillValue = -9.e+034f ;
                t90:MaskValue = 9.e+034f ;
                t90:long_name = "temperature ITS-90" ;

        float sa(time, lat, lon, de) ;
                sa:units = "psu" ;
                sa:MissingValue = -9.e+034f ;
                sa:_FillValue = -9.e+034f ;
                sa:MaskValue = 9.e+034f ;
                sa:long_name = "salinity" ;

        short pr_cnt(time, lat, lon, de) ;
                pr_cnt:MissingValue = 0s ;

        float pr_err(time, lat, lon, de) ;
                pr_err:units = "dbars" ;
                pr_err:MissingValue = -9.e+034f ;
```

```
                    pr_err:long_name = "pressure error variance (sqrt)" ;

          short t90_cnt(time, lat, lon, de) ;
                    t90_cnt:MissingValue = 0s ;

          float t90_err(time, lat, lon, de) ;
                    t90_err:units = "degrees C" ;
                    t90_err:MissingValue = -9.e+034f ;
                    t90_err:long_name = "temperature ITS-90 error variance (sqrt)" ;

          short sa_cnt(time, lat, lon, de) ;
                    sa_cnt:MissingValue = 0s ;

          float sa_err(time, lat, lon, de) ;
                    sa_err:units = "psu" ;
                    sa_err:MissingValue = -9.e+034f ;
                    sa_err:long_name = "salinity error variance (sqrt)" ;

          short de_cnt(time, lat, lon, de) ;
                    de_cnt:MissingValue = 0s ;
```

// global attributes:

```
                    :latmin = -80.f ;
                    :latmax = 80.f ;
                    :latincr = 1.f ;
                    :lonmin = -110.f ;
                    :lonmax = 30.f ;
                    :lonincr = 1.f ;
                    :node_offset = 1 ;
                    :nprops = 3 ;
                    :counts_included = 3 ;
                    :title = "HydroBase" ;
                    :command = "hb_ncmerge 1000 … 5701 -D/data1/NewGrids/ATLANTIC/work_1deg
-E.1deg.sm3d.jun.nc  -O/data1/NewGrids/cdf_1deg/Atlantic.1deg.FG.jun.nc -Ppr/t90/sa -L0" ;
                    :compliance = "HydroBase3" ;
```

data:

latitude = 79.5, 78.5, 77.5, 76.5, 75.5, 74.5, 73.5, 72.5, 71.5, 70.5, 69.5
68.5, 67.5, 66.5, 65.5, 64.5, 63.5, 62.5, 61.5, 60.5, 59.5, 58.5, 57.5 …. rest of latitudes

longitude = -109.5, -108.5, -107.5, -106.5, -105.5, -104.5, -103.5, -102.5,
-101.5, -100.5, -99.5, -98.5, -97.5, -96.5, -95.5, -94.5, -93.5, -92.5 …. rest of longitudes

de = 0, 10, 20, 30, 50, 75, 100, 125, 150, 200, 250, 300, 350, 400, 450,
500, 550, 600, 650, 700, 750, 800, 850, 900, 950, 1000, 1100, 1200, 1300,
1400, 1500, 1600, 1700, 1800, 1900, 2000, 2100, 2200, 2300, 2400, 2500,
2600, 2700, 2800, 2900, 3000, 3100, 3200, 3300, 3400, 3500, 3600, 3700,
3800, 3900, 4000, 4100, 4200, 4300, 4400, 4500, 4600, 4700, 4800, 4900,
5000, 5100, 5200, 5300, 5400, 5500, 5600, 5700, 5800, 5900, 6000, 6200,
6400, 6600, 6800, 7000, 7500, 8000, 9000, -99 ;
}