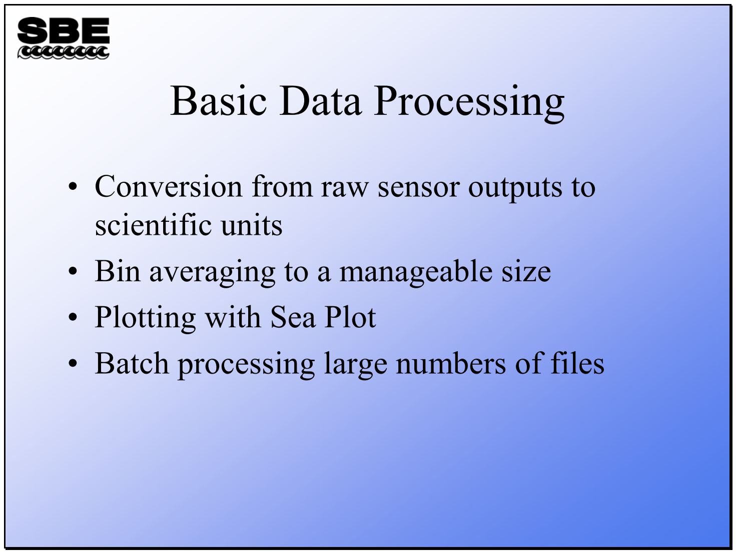
Module 3

Basic Data Processing

Overview

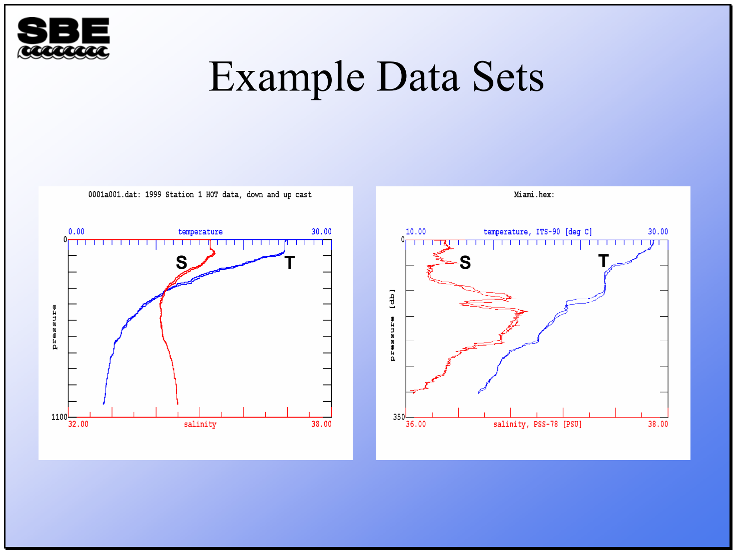


In this module we will cover the data processing basics, no fancy stuff. With this information you can go from raw data file to something you can print or plot.

When we finish this module you should be able to:

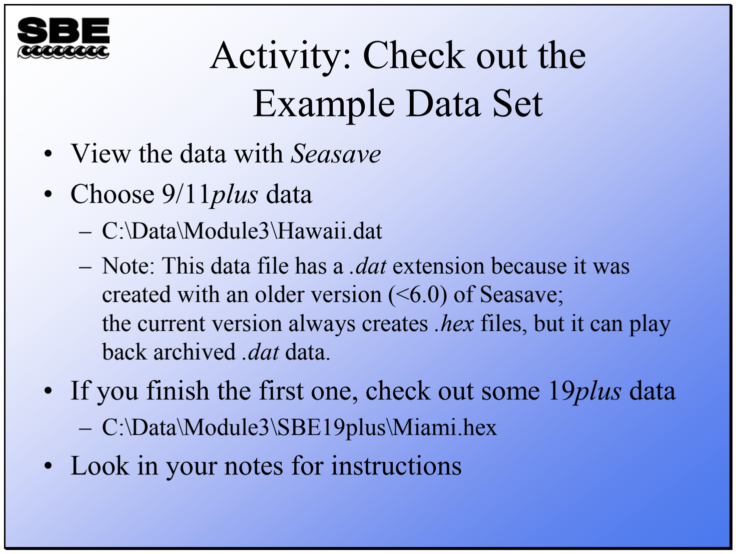
- Run *Data Conversion* and convert your raw data to scientific units.
- Run *Bin Average* to reduce your data set, producing data at even depth values.
- Run *Sea Plot* to display your work.
- Process large numbers of data files by making script files.

Basic Data Processing: Example Data



Here are Seasave plots of some data collected on Hawaii Ocean Time-series cruise 101 and a cruise between Miami and Bimini on a RSMAS vessel.

Activity: View the Example Data



Run Seasave

Set up the plot - right click on the plot and select *Modify*:

For the SBE 9/11*plus* data:

Y axis: Pressure 0 to 1200 decibars

X axis: Temperature 0 to 30 degrees C

X axis: Salinity 34 to 36 PSU

Click OK.

Click the menu *Archived Data* -> *Start:*

Click *Select Data File*:

Select C:\Data\Module3\Hawaii.dat and click Open.

Seasave asks if you want to change .con file to C:\Data\Module3\Hawaii.con, click Yes.

Click *Start*.

Click the menu *Archived Data* -> *No Wait*

If you have time, plot SBE 19*plus* data:

Y axis: Pressure 0 to 350 decibars

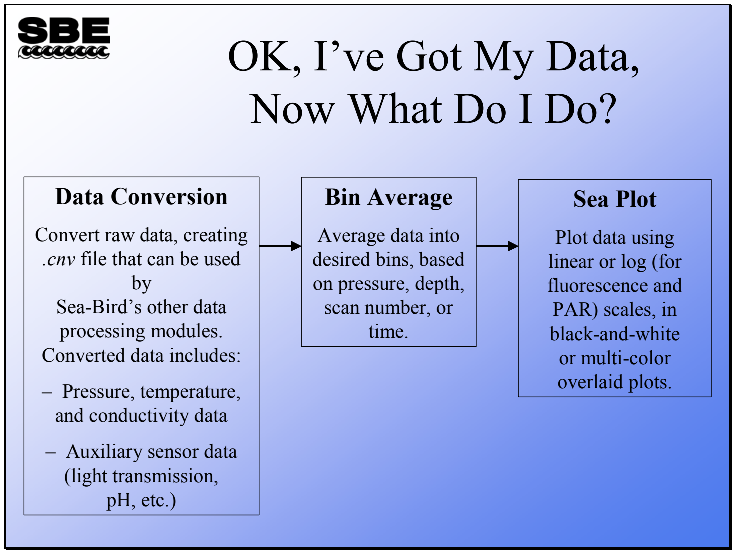
X axis: Temperature 10 to 30 degrees C

X axis: Salinity 36 to 38 PSU

C:\Data\Module3\SBE19plus\Miami.hex

C:\Data\Module3\SBE19plus\Miami.con

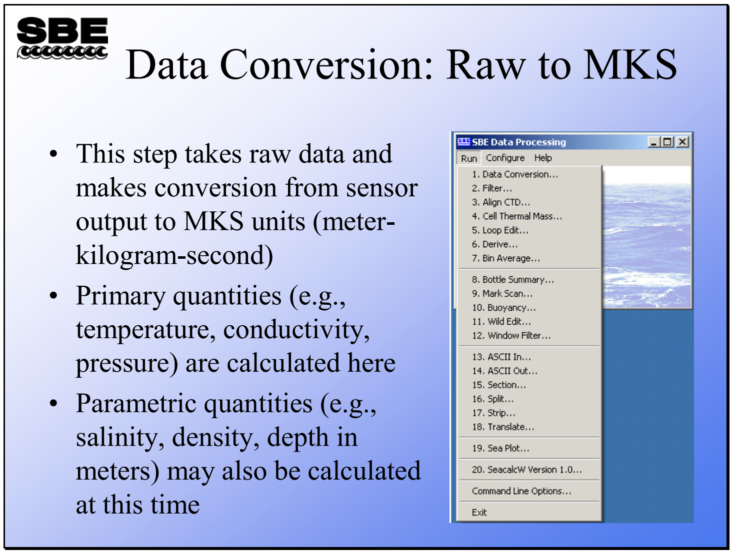
Data Processing: Flow Chart



Data processing involves operating on your raw data (which came from the sensors attached to the instrument) to convert it from the sensor outputs to scientific units, calculate any additional oceanographic parameters of interest, and reduce the data set to a tractable size. It is always the best practice to archive your raw data, because there is no going backwards once you have processed it. Should you discover calibration errors, omit necessary parameters, or make processing errors, it is imperative to return to your raw data set.

Sea-Bird's data processing program, SBE Data Processing, includes a number of modules for processing data; just three are listed above.

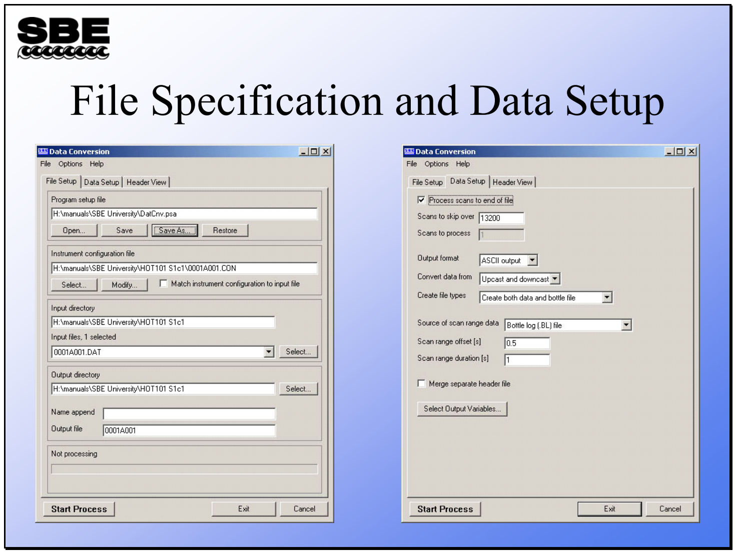
Data Processing: Conversion to Scientific Units



For the basics, we will concern ourselves with Data Conversion and Bin Averaging. Data conversion is the first step.

Data Conversion takes your raw data (.dat or .hex) file and, with the information contained in the .con file, converts it to scientific units. The form the converted data takes is set up in the SBE Data Processing-Win32 Data Conversion dialog. In addition to quantities like temperature and conductivity, parameters that are calculated from these, such as salinity and density, are also available. If you plan to do more advanced processing of your data, you should wait until the advanced processing is complete before calculating parameters derived from the measured parameters.

Data Conversion: File Specification



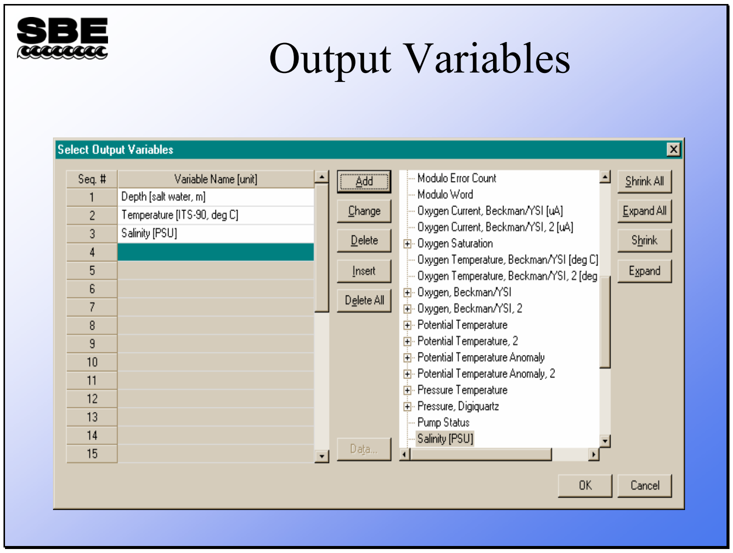
The program setup (.psa) file contains information regarding how the program was set up the last time you used it. This relieves you of the task of choosing the variables and other processing options. In addition, if you have more than one preferred processing protocol, you can store them as separate .psa files. As always, you must select the appropriate .con file for your instrument. You can select multiple files for processing; a separate .cnv file will be created for each file that is processed.

You can process all or part of the data file and skip the beginning (during the time you were checking your instrument before the cast). You can convert the whole cast or just the downcast. If you collected water samples, you can generate a file containing some CTD scans that were collected while the water bottle was being closed. The means of determining where these scans should come from are:

- For the 9*plus* with a G.O. 1015 water sampler, the data stream is marked with a status bit when the bottle closure occurs.
- For internally recording instruments with an AFM, the file uploaded from the AFM (an *.afm* file) can be used.
- A *.bsr* file can be made from a *.mrk* file (collected while running Seasave) with the Mark Scan module.
- If using an SBE 32 Carousel, Seasave writes a *.bl* file that contains the information required.

Once you have settled on the source of information for indicating bottle closure times, you have to select the amount of data to be included in the file for each closure. *Scan range offset* is how far back Data Conversion looks in your data stream in seconds and *Scan range duration* is how long it looks, again in seconds. For example, an offset of -5 seconds and duration of 10 seconds means you will get 5 seconds of data before the bottle closure and 5 seconds after the closure.

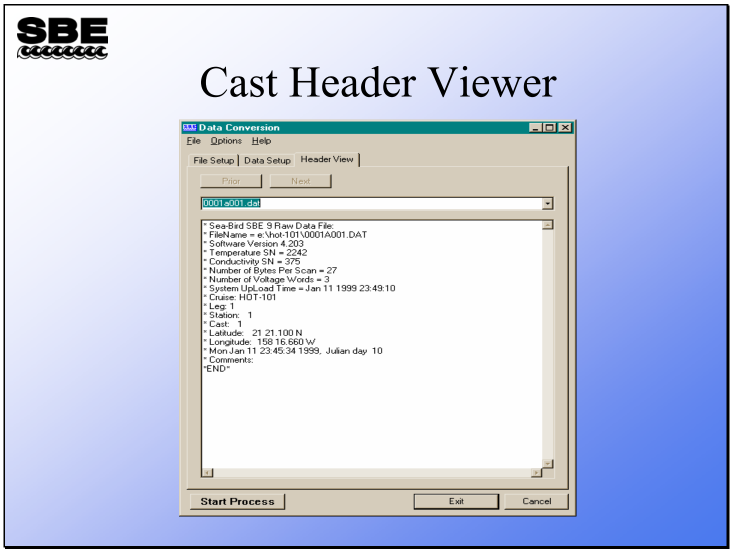
Data Conversion: Output Variable Choices



The output variables dialog allows you to select a table of items that will be calculated and the order in which they will be calculated.

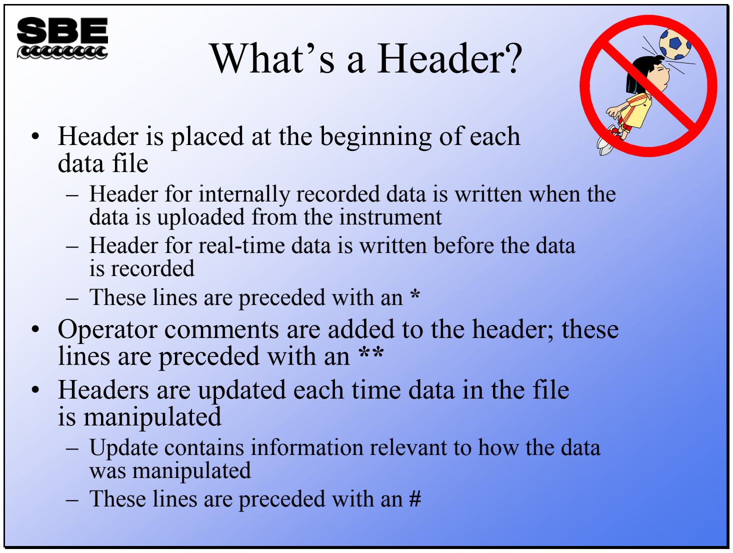
Note that the list on the right reflects the information in the selected .con file – only data measured by the sensors indicated in the .con file, and data that can be calculated from those measurements, are available for output.

Data Conversion: Cast Headers



The header view can reassure you that you are working with the instrumentation you think you are.

Data Conversion: File Headers Revealed



* Sea-Bird SBE 9 Raw Data File:
* FileName = H:\tests\USERS\mikev\Tau Test 2.hex
* Software Version Seasave Win32 v%s
* Temperature SN = 4022
* Conductivity SN = 4022
* Number of Bytes Per Scan = 14
* Number of Voltage Words = 0
* System UpLoad Time = Jun 11 2001 09:51:47
** Ship: RV TestBath
** Cruise: Tau test
** Station: Bellevue WA
** 11 Jun 01 bath 0 Tau Test with 19plus #2
# nquan = 3
# nvalues = 2596
# units = specified
# name 0 = timeS: Time, Elapsed [seconds]
# name 1 = tv290C: Temperature [ITS-90, deg C]
# name 2 = flag: 0.000e+00
# span 0 =   0.000,   648.750
# span 1 =   1.7905,   6.5876
# span 2 = 0.0000e+00, 0.0000e+00
# interval = seconds: 0.25
# start_time = Jun 11 2001 09:51:47
# bad_flag = -9.990e-29
# sensor 0 = Frequency 0 temperature
# sensor 1 = Frequency 1 conductivity, 4022, 25-Apr-01, cpcor = -9.5700e-08
# sensor 2 = Frequency 1 conductivity
# sensor 3 = Pressure Voltage
# datcnv_date = Jun 11 2001 10:39:01, 1.00
# datcnv_in = H:\SBE University\data\19p tau test\Tau Test 2.hex H:\SBE University\data\19p tau test\4022.con
# datcnv_skipover = 0
# file_type = ascii
END

Data Conversion: Output File Format

Data Conversion Output, *.cnv*

```
# datcnv_date = Jun 25 2001 10:21:54, 5.2
# datcnv_in = HOT101\0001a001.dat HOT101\S1C1\0001a001.con
# datcnv_skipover = 12000
# file_type = ascii
*END*
    10.115   24.7800   35.2446  0.000e+00
    10.178   24.7799   35.2442  0.000e+00
    10.170   24.7798   35.2443  0.000e+00
    10.115   24.7799   35.2444  0.000e+00
       ↑        ↑         ↑        ↑
       P        T         S      Error
     (db)     (°C)      (PSU)    flag
```

```
* Sea-Bird SBE 9 Raw Data File:
* FileName = D:34-8.DAT
* Software Version 4.0.b.g
* Temperature SN = 1123
* Conductivity SN = 915
* Pressure SN = 43440
* Number of Bytes Per Scan = 15
* System UpLoad Time = January 24, 1992  3:39:09 pm
* Ship:    R/V Poseidon
* Cruise:   189-1a
* Station:  34
* Cast:    8
* Latitude: 35 22.00' N
* Longitude:12 43.36' W
* Second set of CTD trials, station 34 cast 8, SBE Mk3 Mk5a Mk5b probably no FSI trial.
# nquan = 5
# nvalues = 77205
# units = metric
# name 0 = depS: depth, salt water [m]
# name 1 = t068: temperature, IPTS-68 [deg C]
# name 2 = c0S/m: conductivity [S/m]
# name 3 = sal00: salinity, PSS-78 [PSU]
# name 4 = flag:  0.000e+00
# span 0 = 1.519, 3555.084
# span 1 = 2.4822, 17.1124
# span 2 = 3.257845, 4.669567
# span 3 = 34.9097, 36.4804
# span 4 = 0.0000e+00, 0.0000e+00
# interval = seconds: 0.0416667
# start_time = Jan 24 1992 15:39:09
# bad_flag = -9.990e-29
# sensor 0 = Frequency  0  temperature, 1123
# sensor 1 = Frequency  1  conductivity, 915, cpcor = -9.5700e-08
# sensor 2 = Frequency  2  pressure, 43440
# sensor 3 = Extrnl Volt  0  oxygen, current, 130259
# sensor 4 = Extrnl Volt  1  oxygen, temperature, 130259
# datcnv_date = Jun 22 2001 16:19:48, 4.249
# datcnv_in = 34-8.DAT 34-8.CON
# datcnv_skipover = 0
# file_type = ascii
*END*
    1.636   16.9770   4.654863   36.4699  0.000e+00
    1.691   16.9784   4.655062   36.4703  0.000e+00
```

Selected output variables / headings for columns of data below (error flag column added automatically by Data Conversion)

Data Conversion: Water Sampler Output File Format

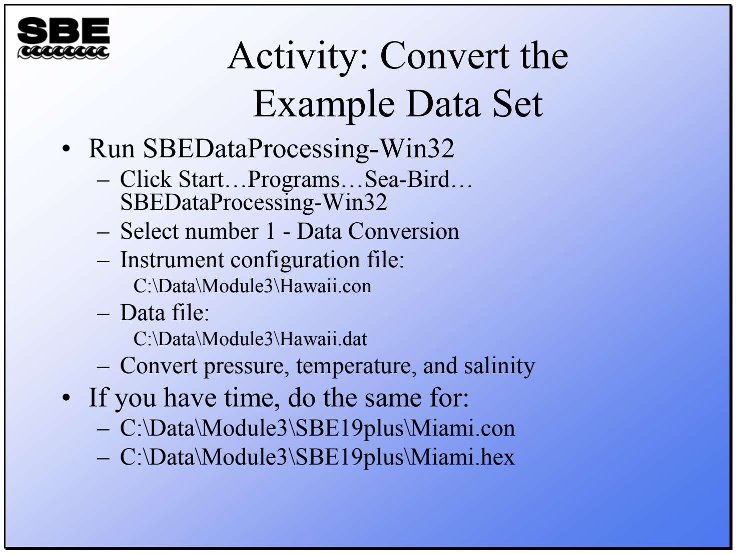
SBE

Data Conversion Output, *.ros*

1011.823	4.0297	34.5000	43234	1	0.000e+00
1011.696	4.0297	34.5000	43235	1	0.000e+00
1011.886	4.0297	34.5000	43236	1	0.000e+00
1011.759	4.0299	34.5000	43237	1	0.000e+00
1011.839	4.0303	34.4997	43238	1	0.000e+00
1011.759	4.0308	34.4998	43239	1	0.000e+00

The data conversion process also writes a file with data collected while water bottles were being closed. You choose the time window of data you want written to the file. For example, you might select to have all scans collected starting 5 seconds before the bottle closes and ending 5 seconds after the bottle closure written to the file. We will discuss the use of this data in correction of CTD conductivity data in the advanced data processing topics.

Activity



Click Start -> Programs -> Sea-Bird -> SBEDataProcessing-Win32
Click Run -> Data Conversion

Click Select for the Instrument configuration file, and select C:\Data\Module3\Hawaii.con
Click Select for Input files, and select C:\Data\Module3\Hawaii.dat

Click the Data Setup tab:

Process to the end of the file
Skip 0 scans
Select output format ascii
Convert data from upcast and downcast
Create .cnv file only

Click the Select Output Variables button

Click on the first Variable Name. Select Pressure, Digiquartz ->db. Click Add or Change
Click on the second Variable Name. Select Temperature -> ITS90 -> deg C. Click Add or Change
Click on the third Variable Name. Select Salinity. Click Add or Change
Click OK

Click the File Setup tab, and save Program setup file as C:\Data\Module3\HawaiiDatCnv.psa
Click Start Process
Open the .cnv file in Notepad or Wordpad and take a look at the header and data.

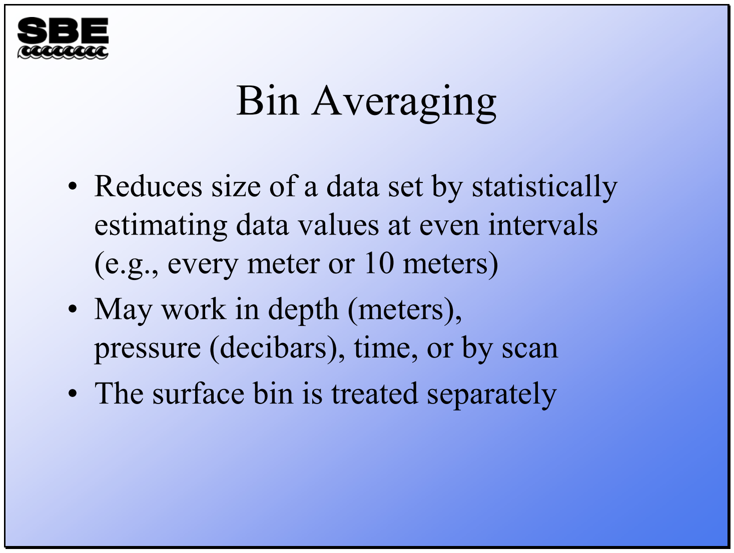
If you have time, repeat this for:

C:\Data\Module3\SBE19plus\Miami.con
C:\Data\Module3\SBE19plus\Miami.hex

Choose Pressure, Strain Gauge->db, Temperature -> ITS90 -> deg C, and Salinity

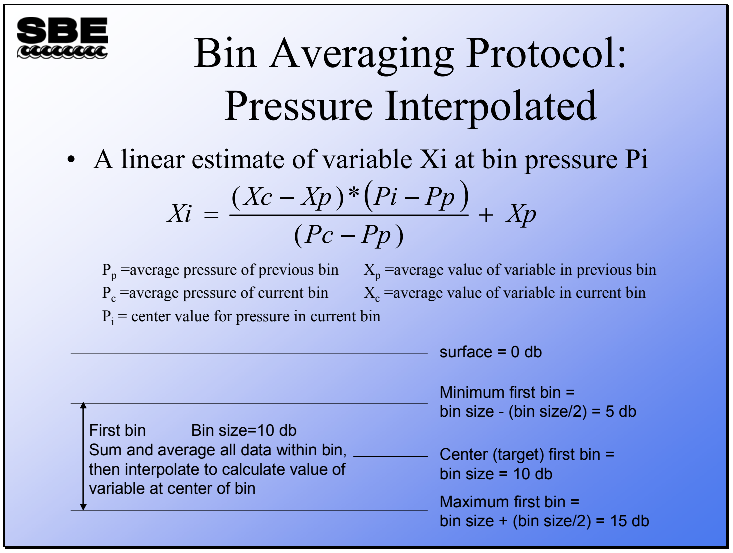
We thank the Hawaii Ocean Time Series program and the Rosenstiel School of Marine and Atmospheric Science for sharing this data with us.

Bin Averaging



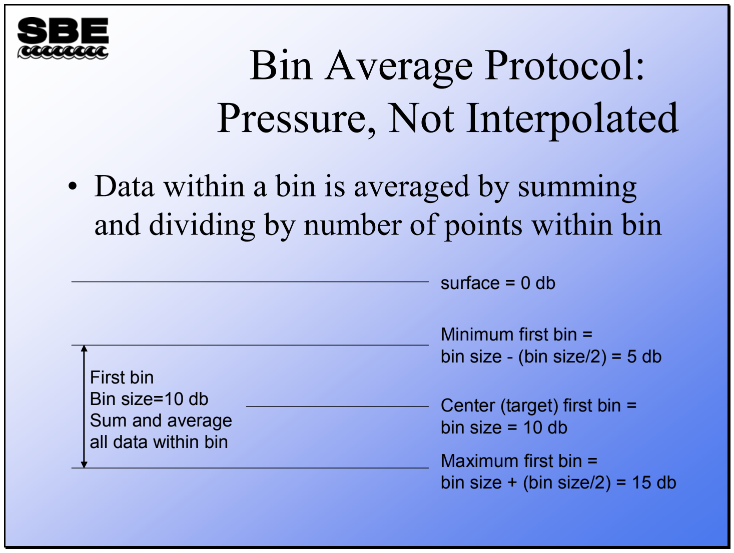
Bin averaging is a means of reducing your data set to a more tractable, and perhaps a more meaningful, size. The Bin Average module makes a statistical estimate of data values at a user-prescribed interval based on the surrounding data. You can bin data on the even meter or 10 meters. You can bin data with a bin size that represents the resolution of your instrument. For time series measurements, you can bin on time interval.

Bin Averaging*:* Processing Protocol



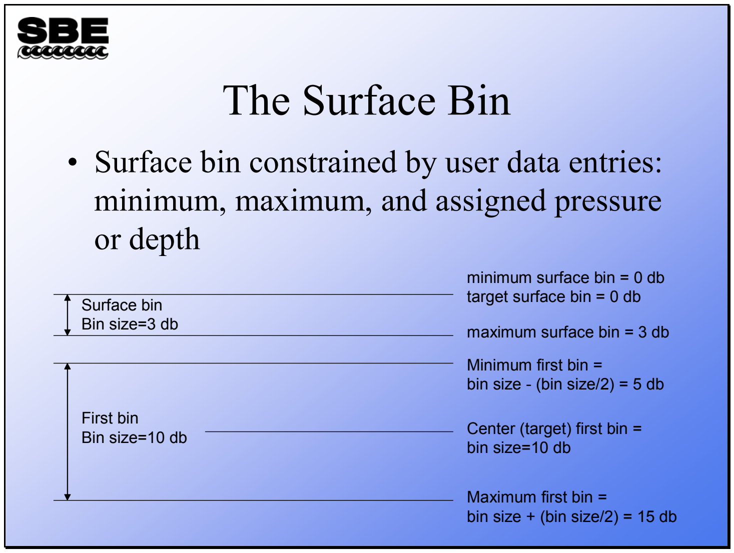
An estimate of each variable is made using the average value of that variable and pressure in the previous bin, and the average values of the variable and pressure in the current bin.

Bin Averaging: Processing Protocol



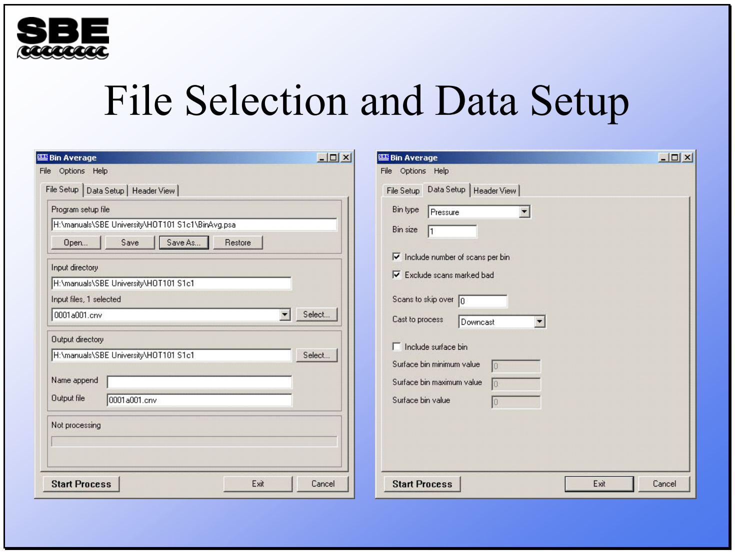
This protocol averages all the data within the bin, producing uneven bin pressures or depths. For example, if you are binning on 10-meter intervals, the first bin start is 5 meters and the end is 15 meters. All data within this window is averaged, producing a bin depth of approximately 10 meters (e.g., 10.123 meters).

Bin Averaging: Surface Bin



The surface bin is handled differently because the previous bin would be up in the air. The surface bin is assigned a beginning pressure or depth, an ending pressure or depth, and a target pressure or depth.

Bin Averaging: File Selection and Data Setup



The Data Setup tab allows your choice of pressure, depth, time, or scan bins. You can include the number of scans per bin in the output file, which is useful for evaluating data from instruments with a low sample rate. Bins with 1 sample in them are not very accurate statistically.

You can skip data that you acquired while checking out your instrument before the cast started. Similar to previous processing, you can process the upcast, downcast, or both.

In the advanced data processing portion of the course, we will discuss techniques for removing suspect data. These data are marked in the data set as *bad scans*. When setting up *Bin Averaging*, you may exclude scans marked bad by previous processing steps.

As mentioned earlier, the surface bin is handled separately. Note that in our example the surface bin is not included, because we are binning on a 1-meter interval. If you bin on a small interval, it is very difficult to calculate a surface bin. For example, with 1-meter bins, a surface bin would run from 0 to 0.5 meters with value 0.25 meters; depending on the profiling and sampling speeds, there would be few samples within that depth. The surface bin is useful for a coarser bin size. For example, with 10-meter bins, the first bin starts at 5 meters and runs to 15 meters. You can succeed in calculating a surface bin that runs from 0 to 5 meters with value 2.5 meters.

Bin Averaging: Output Data

SBE

Bin Average: Output Data

```
# binavg_bintype = meters
# binavg_binsize = 1
# binavg_excl_bad_scans = yes
# binavg_skipover = 0
# binavg_surface_bin = no, min = 0.000, max = 5.000, value
    = 2.500
# file_type = ascii
*END*
     1.000    24.9124    35.2455 0.0000e+00     100.00
     2.000    24.9582    35.2463 0.0000e+00      90.00
     3.000    25.0029    35.2477 0.0000e+00      36.00
```

The output columns for the example bin averaged data above are:

- Bin depth
- Temperature (°C)
- Salinity (PSU)
- Error flag
- Number of scans per bin

Activity

SBE

Activity: Bin Average the Example Data

- Use the data you converted: C:\Data\Module3\Hawaii.cnv
- 5 decibar pressure bins
- Include surface bin
  - Minimum value 0
  - Maximum value 5
  - Surface bin value 2.5
- Name append *B*
- If you have time, do the same for C:\Data\Module3\SBE19plus\Miami.cnv

Click Start -> Programs -> Sea-Bird -> SBEDataProcessing-Win32

Click Run -> Bin Average

Click Select for Input files, and select C:\Data\Module3\Hawaii.cnv

Enter *Name append* of *B*

Output file should be Hawaii.cnv

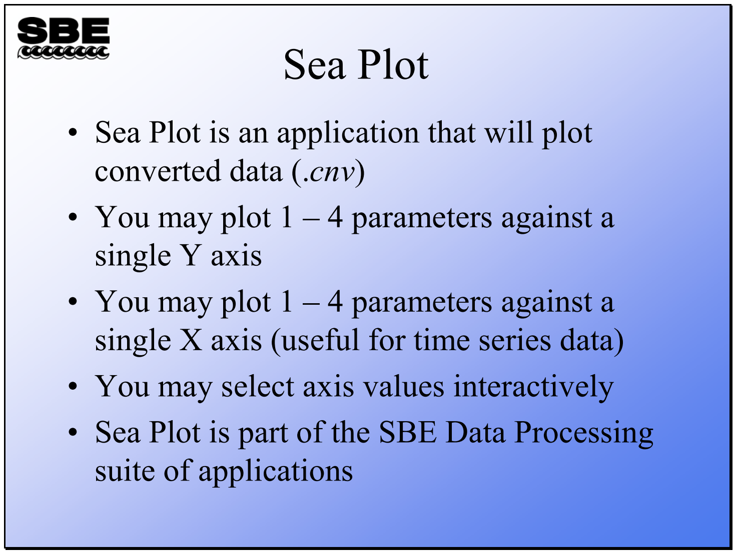
Click the Data Setup tab:
    Choose Pressure for *Bin Type*
    Enter *Bin size* of 5
    Check *Include number of scans per bin*
    Check *Exclude scans marked bad*
    Skip over 0 scans
    Process the downcast

Click Start Process.

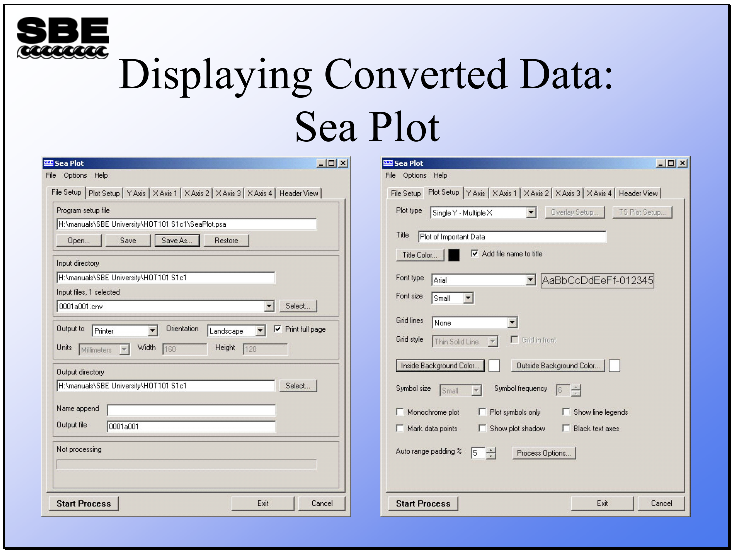
Open HawaiiB.cnv in Notepad or Wordpad and take a look at the header and data.

If you have time, repeat this for C:\Data\Module3\SBE19plus\Miami.cnv, using 1 decibar bins.

Sea Plot: Data Display



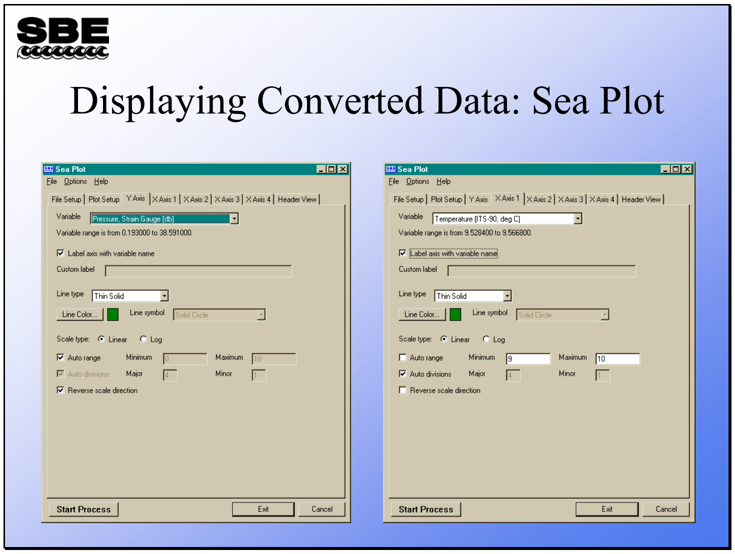
Sea Plot: Displaying Converted Data



The file setup dialog includes an entry for the program setup file or *.psa*. This allows you to store your plot setup for reuse. Output directory settings are included for batch processing.

Plot Setup allows you to choose the plot type (single Y, multiple X or single X, multiple Y), fonts, titles, and color schemes. Symbols are also specified here; if you are plotting to a black and white printer, symbols are very useful. The auto range padding entry sets the amount beyond the range of the data that the minimum and maximum axis values extend.

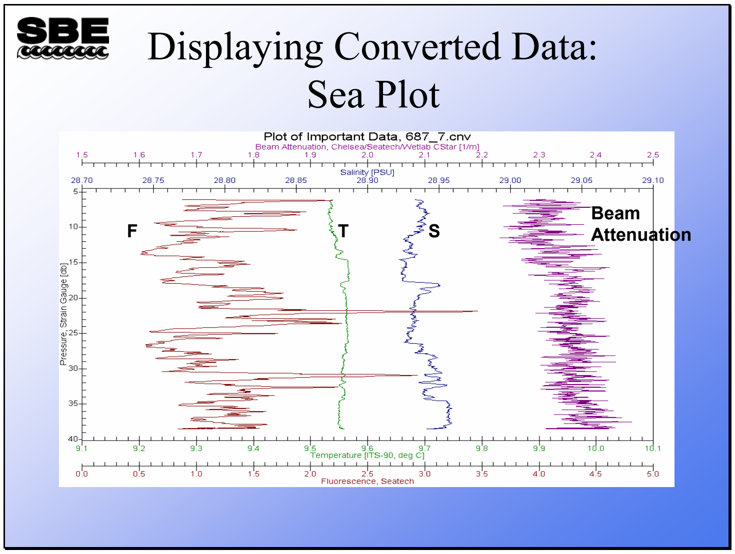
Sea Plot: X and Y-Axis Specification



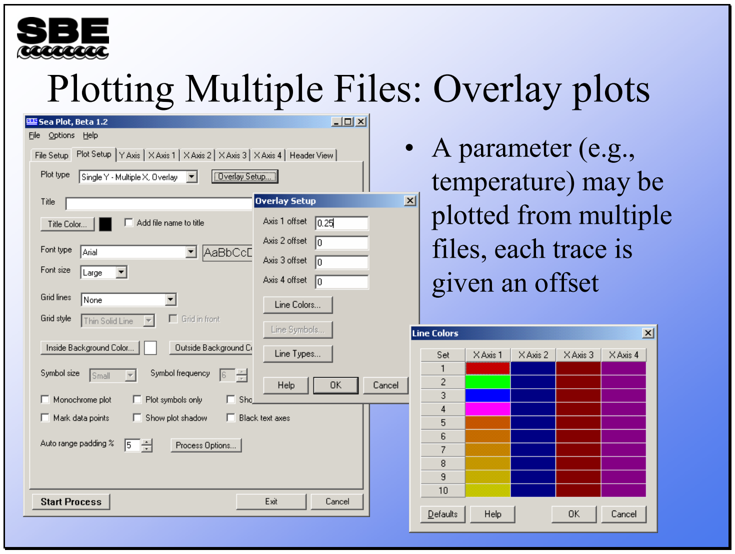
The axis setup tabs allow selection of variable, line type, and color. Plot range and scale direction are also specified here.

Note that the list of variables reflects the information in the selected .cnv file – only variables in the .cnv file are available for plotting, with the following exception. Sea Plot can calculate and plot salinity and/or density *on the fly*, if conductivity, temperature, and pressure data are in the .cnv file. This allows you to skip running the Derive module if you want a quick look at salinity and density before performing other processing steps.

Sea Plot: The Plot

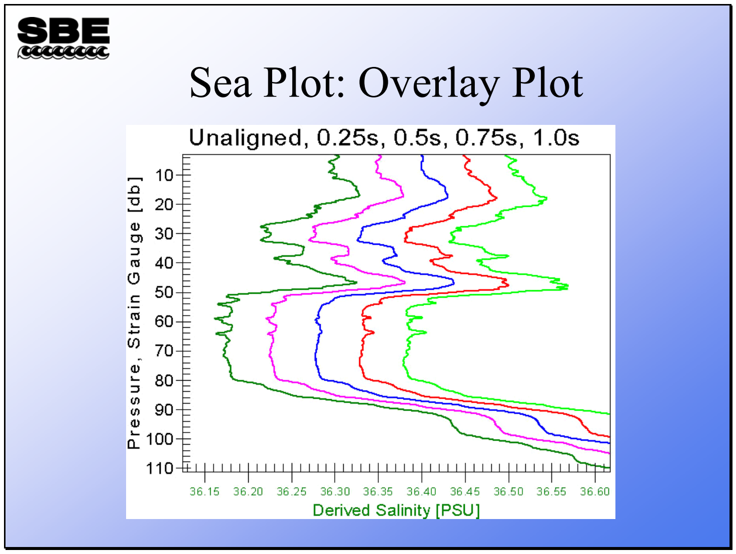


Sea Plot: Plotting Multiple Files, Overlay Plots

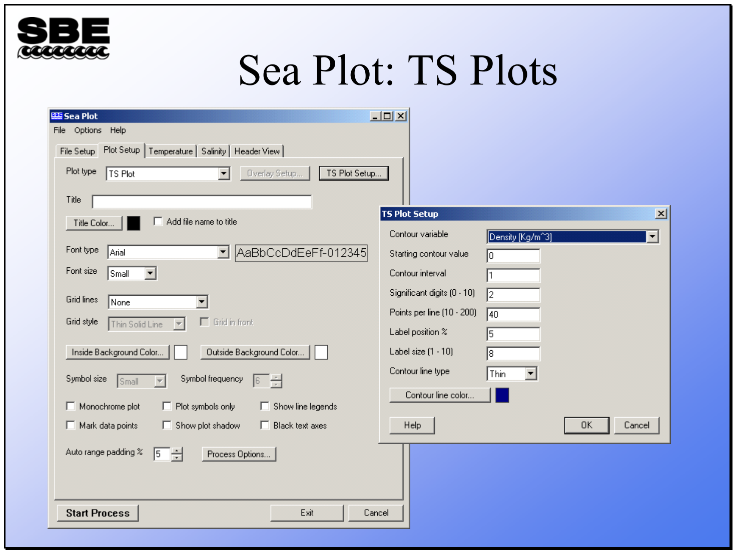


You can plot up to 4 parameters from multiple files by selecting the *overlay* plot type. To use this option you must specify the files in the order (left to right) you want them to appear. You must also enter an *offset* amount; this is amount of space between traces. For example, you might want to space temperature traces from casts taken at hour intervals at the same location by 0.2 degrees. You can also select the colors that each file will be plotted in. There is a color wheel that will allow selection of a graded set of line colors that is available by double clicking the axis identifier.

Sea Plot: Plotting Multiple Files

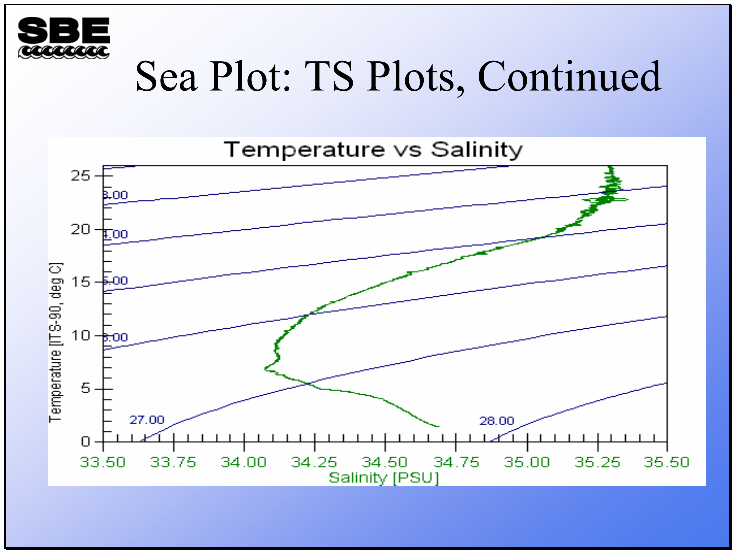


Sea Plot: TS Plots



Temperature *versus* salinity plots are useful for discerning water masses. It is helpful if contours of constant density are placed on the plot as well. Sea Plot's TS plotting format allows you to do this. The file and axis menus above are as expected for Sea Plot. In addition you may define the density contours that are most useful for your application.

Sea Plot: TS Plots (*continued*)



Activity

Activity: Plot your Data

- Use *Sea Plot* to plot your converted files
- First do C:\Data\Module3\Hawaii.cnv
- Then do C:\Data\Module3\HawaiiB.cnv
- If you processed the Miami data, plot it too

Click Start -> Programs -> Sea-Bird -> SBEDataProcessing-Win32

Click Run -> Sea Plot

Click Select for Input files, and select C:\Data\Module3\Hawaii.cnv

Click the Plot Setup tab
    Choose a *Plot type* of Single Y - Multiple X

Click the Y Axis tab
    For *Variable*, choose Pressure, Digiquartz [db]
    Check *Auto range*

Click the X Axis 1 tab
    For *Variable*, choose Temperature
    Uncheck *Auto range*
    Enter a minimum of 0, a maximum of 30

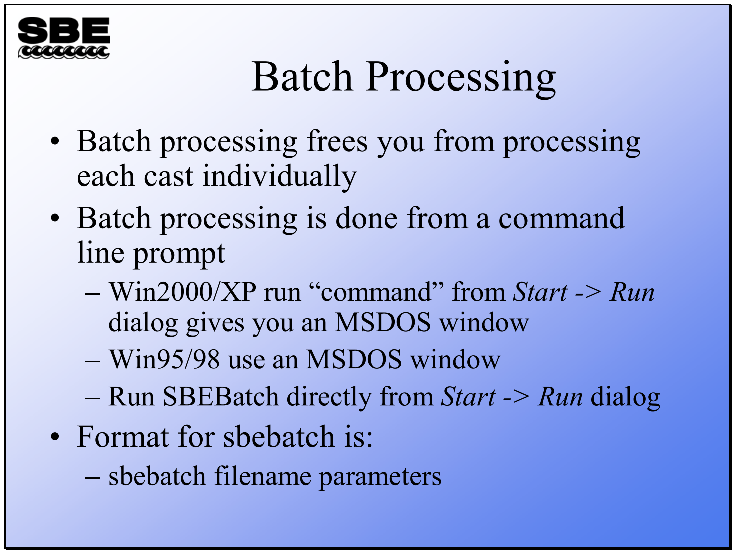
Click the X Axis 2 tab
    Check *Include axis*
    For *Variable*, choose Salinity
    Uncheck *Auto range*
    Enter a minimum of 34, a maximum of 36

Click Start Process

Repeat for C:\Data\Module3\HawaiiB.cnv

Repeat for Miami data

Data Processing of Large Numbers of Files



For processing large sets of cast data, batch mode processing automates the job. You can use the windows scripting host or a program provided with SBE Data Processing, SBEBatch.exe. Your batch file can take advantage of command line parameters and wild card characters.

You can run SBEBatch from a DOS window or from the Windows Run dialog (Start -> Run). In all the examples we'll use today (and the examples in the SBE Data Processing manual), we're assuming you are running from the Windows Run dialog box.

Data Processing of Large Numbers of Files (*continued*)

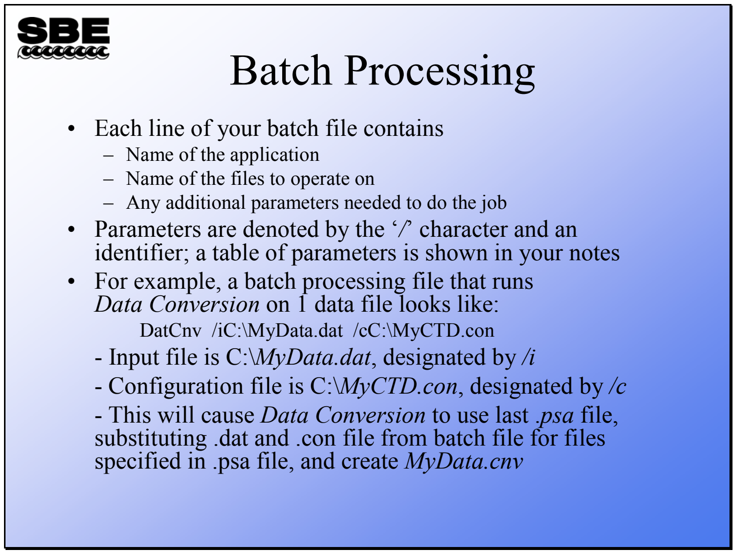
Batch Processing

- Batch processing uses an application that runs other applications (*i.e.,* data processing applications)
- You may use the Windows Scripting Host or an application Sea-Bird provides, *SBEBatch*
- The applications that the batch processor runs are listed in a text file that you make with a text editor like Notepad
  - A list of applications are shown in your notes
- SBEBatch reads each line of the text file and runs each application in turn

Applications:

Module	**Process Name**
Align CTD	Alignctd
ASCII In	Asciin
ASCII Out	Asciiout
Bin Average	Binavg
Bottle Summary	Bottlesum
Buoyancy	Buoyancy
Cell Thermal Mass	Celltm
Data Conversion	Datcnv
Derive	Derive
Filter	Filter
Loop Edit	Loopedit
Mark Scan	Markscan
Sea Plot	Seaplot
Section	Section
Split	Split
Strip	Strip
Translate	Trans
Wild Edit	Wildedit
Window Filter	Wfilter

Data Processing of Large Numbers of Files (*continued*)



A list of the most commonly used parameters follows; see the SBE Data Processing manual for a complete list:

Parameter	**Description**
/cString	Use *String* as instrument configuration (.con) file. *String* must include full path and file name. Note: If using /cString, must also specify input file name (using /iString).
/iString	Use *String* as input file name. *String* must include full path and file name. This parameter supports standard wildcard expansion: • ? matches any single character in specified position within file name or extension • * matches any set of characters starting at specified position within file name or extension and continuing until end of file name or extension or another specified character
/oString	Use *String* as output directory (not including file name).
/fString	Use *String* as output file name (not including directory).
/aString	Append *String* to output file name (before extension).
/pString	Use *String* as Program Setup (.psa) file. String must include full path and file name.
/xModule: String	Use *String* to define an additional parameter to pass to Module. Not all modules have x parameters; see module descriptions for details. If specifying multiple x parameters, enclose in double quotes and separate with a space. *Example*: Run Data Conversion, telling it to skip first 1000 scans: /xdatcnv:skip1000
#m	Minimize SBE Data Processing window while processing data, allowing you to do other work on computer.

If specifying multiple parameters, insert a space between each parameter in the list.

Data Processing of Large Numbers of Files (*continued*)

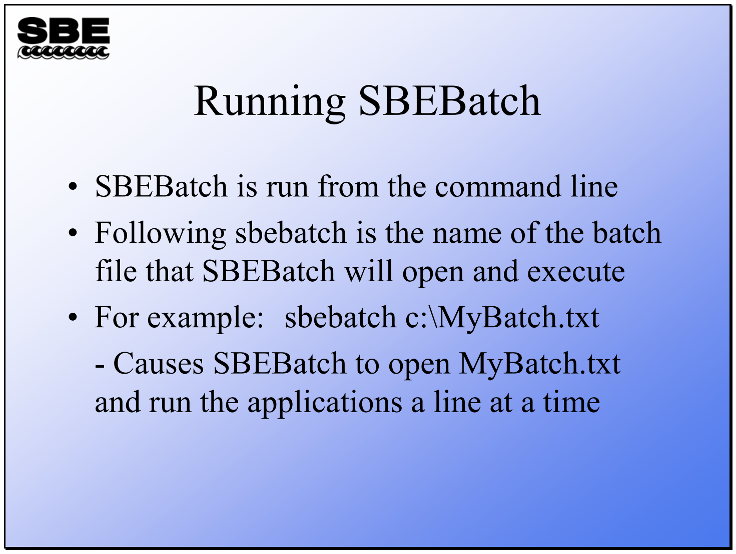
Batch Processing Script

- To process all the files in a folder use a wildcard: the '*' character
- For example, a batch processing file that runs Data Conversion on all data files in a folder looks like:

```
datcnv  /iC:\Data\*.dat  /cC:\Data\MyCTD.con
```

- Input files are all *.dat* files in C:\Data\
- Configuration file is C:\Data*MyCTD.con*

Data Processing of Large Numbers of Files (*continued*)



Here's an example of how to use batch processing to run Data Conversion and Derive to process all the files in C:\Data:

1. Run Data Conversion, entering the desired choices in the File Setup and Data Setup dialog boxes. Upon completing setup, press Save or Save As on the File Setup tab. The configuration is stored in the .psa file.
   Repeat for Derive.

2. Create a batch file named batch.txt in C:\Data, which contains the following lines:
   ```
   @Lines starting with @ are comment lines, and have no effect on the result
   @Use these to document what you are doing in the batch file
   @Processing data from February 2006 Cruise
   datcnv /iC:\Data\*.dat /cC:\Data\MyCTD.con
   derive /iC:\Data\*.cnv /cC:\Data\MyCTD.con
   ```

3. Select Run in the Windows Start menu. The Run Dialog box appears.
   Type in the batch processing program name and the .txt file name:
   ```
   sbebatch c:\Data\batch.txt
   ```

The results:

1. Data Conversion uses its last .psa file, substituting the .con file from the batch file for the .con file specified in the .psa file, and processes **all** .dat files in C:\Data, creating a .cnv file from each .dat file.
2. Derive uses its last .psa file, substituting the .con file from the batch file for the .con file specified in the .psa file, and processes **all** .cnv files in C:\Data (which were just created by Data Conversion), creating a .cnv file from each .cnv file.

Data Processing of Large Numbers of Files (*continued*)

Batch Processing Script

- Remember that the format for running SBEbatch is:
  sbebatch filename *parameters*
- You can operate on files in different folders with the same batch file by using command line *parameters*
- These are entered after the batch file name and are denoted by the '%' character and a number
  - The first command line parameter is %1, the second is %2, etc.
- Your batch file must have entries that use the '%' parameters

Data Processing of Large Numbers of Files (*continued*)

SBE

Batch Processing Script

- For example, a batch file that has this line in C:\MyBatch.txt

  DatCnv /i%1*.dat /c%1\MyCTD.con

  Executed with this command line

  SBEBatch C:\MyBatch.txt C:\Data
  (C:\Data is the %1 parameter)

  Will cause Data Conversion to be run like this:

  DatCnv /iC:\Data*.dat /cC:\Data\MyCTD.con

  All the .dat files in C:\Data will be converted

- For the same batch file, if the command line is

  SBEBatch C:\MyBatch.txt C:\NewData

  All the .dat files in C:\NewData will be converted

Now let's add a bit more flexibility to the process. Here's an example of how to use batch processing to run Data Conversion and Derive to process all the data files in C:\Data\Leg1, C:\Data\Leg2, and C:\Data\Leg3:

1. Run Data Conversion, entering the desired choices in the File Setup and Data Setup dialog boxes. Select *Match instrument configuration to input file* on the File Setup tab. Upon completing setup, press Save or Save As on the File Setup tab. The configuration is stored in the .psa file.
   Repeat for Derive.

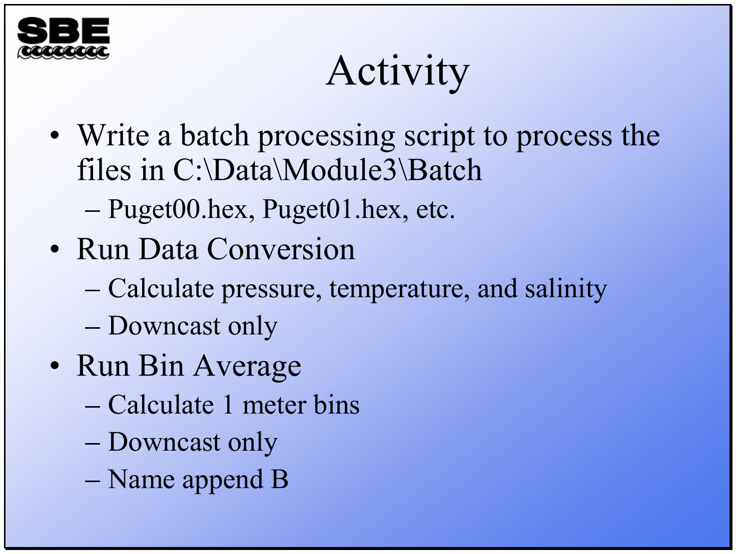
2. Create a batch file named batch.txt in C:\Data, which contains the following lines:
   @Processing data from 3 legs of February 2006 Cruise
   datcnv /i%1*.dat
   derive /i%1*.cnv

3. Select Run in the Windows Start menu. The Run Dialog box appears.
   Type in the batch processing program name, the .txt file name, and the %1 parameter:
   sbebatch C:\Data\batch.txt C:\Data\Leg1
   Repeat for the files in Leg2 and Leg3:
   sbebatch C:\Data\batch.txt C:\Data\Leg2
   sbebatch C:\Data\batch.txt C:\Data\Leg3

The results:

1. Data Conversion uses its last .psa file, substituting the matching .con file for the .con file specified in the .psa file, and processes **all** .dat files in C:\Data\Leg1, creating a .cnv file from each .dat file.
2. Derive uses its last .psa file, substituting the matching .con file for the .con file specified in the .psa file, and processes **all** .cnv files in C:\Data\Leg1 (which were just created by Data Conversion), creating a .cnv file from each .cnv file.
3. Steps 1 and 2 are repeated for the files in C:\Data\Leg2 and C:\Data\Leg3.

Activity



1. Prepare Data Conversion and Bin Average to operate in batch mode:
   a. Process Puget00.hex with Data Conversion, using Puget.con
   b. Save the .psa file as C:\Data\Module3\Batch\DatCnv.psa
   c. Process Puget00.cnv with Bin Average
   d. Save the .psa file as C:\Data\Module3\Batch\BinAvg.psa
   e. Check to see that you were successful with Notepad
   f. Delete the .cnv files that resulted

2. Use Notepad to write your batch file, using a %1 parameter for the file locations

3. Save your batch file as C:\Data\Module3\Batch\MyBatch.txt

4. Launch a command line session: Click Start > Run.

5. Run your batch file with the command line:
   Sbebatch C:\Data\Module3\Batch\MyBatch.txt C:\Data\Module3\Batch

6. Check to see that you were successful with Notepad