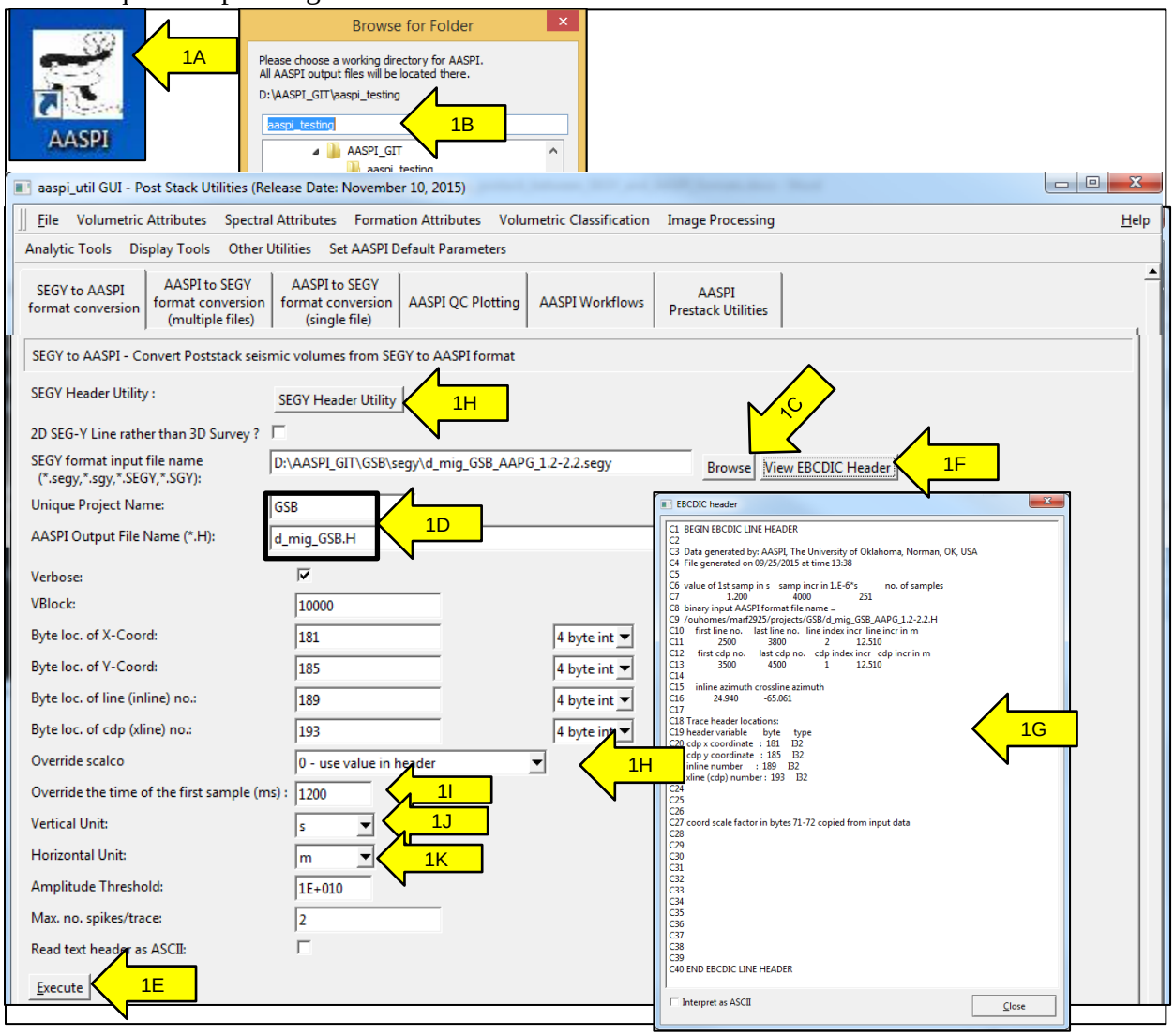
Converting Poststack Between SEGY and AASPI Formats

Poststack Data Conversion from SEGY to AASPI format

Data conversion will be the first and most difficult step in the AASPI workflow. Most oil companies have a small army of exploration technologists whose most time-consuming jobs in importing and exporting data in and out of workstation software. Their salaries are well earned! All subsequent steps using the AASPI software will be a lot easier.



To start AASPI, double click on the AASPI shortcut on your desktop (1A) (Yes, the logo is indeed a snake grasping the letter π). A directory browser will pop up (1B), allowing you to change the working directory of AASPI (i.e. the directory into which the AASPI output files will be written). Navigate to the desired directory and click OK. The aaspi_util window will pop up. In Linux, simply navigate to your project directory and type:

```
aaspi_util &
```

To begin, you need to convert SEGY-format data to AASPI-format data. We think of the Society of Exploration Geophysics SEGY format as not so much a standard, but as a suggestion. Many companies were acquiring 3D seismic data and writing 3D interpretation software before the 3D standard was adopted in 2002. The AASPI software uses a format that is compatible with that written by the Stanford Exploration Project (SEP format) and workers at UT-Austin, and Colorado School of Mines (Madagascar format). Our format only differs in that we carry around more headers (such as mute zones) and the explicit orientation of the survey.

The first tab of aaspi_util is defines this format conversion. Browse to your SEGY file (1C) and specify a unique project name, as well as the output AASPI-format file (preferably with .H extension) (1D). If you are using the New Zealand Great South Basin (GSB dataset), enter the project name to be "GSB", "class", or whatever you wish. Our practice is to name the migrated data something like d_mig_GSB.H, where "d" indicates the file contains seismic data, the "mig" that it is migrated, the "GSB" that it is the GSB or Great South Basin survey, and the ".H" that it is a "header" in the SEP format. The only required part of this naming convention is the *.H at the end, since the GUIs will search for data files with this ending. Next you need to know where the headers are located on your data. For the GSB data volume, these will be the default 2002 SEGY standard locations, so you can simply click the Execute button (1E). You will see data being read (twice!) in the black window. In the first pass it breaks the data into the "SEP" four-file format: an ASCII-format *.H history file, an ASCII-format *.H@@ file containing a map of the trace headers, an IEEE binary format *.H@ file containing the seismic (or attribute) floating point samples, and finally, an IEEE binary format *.H@@@ file containing the (typically integer) header values. (Dude! Why did those California folk like the "@" so much?). As complicated (and annoying) as this may seem, this "broken file" format is common to most larger processing centers. In this way, sorting the data requires only reading and sorting the small header files which can remain in computer memory, in contrast to reading multiple times through the large SEGY format file containing headers and data together.

If you are loading your own dataset, you may want to see header information by clicking on (1F) "View EBCDIC header" button. This will display the text-format "line header" in your SEGY file (1G). There are several fundamental header values you need in order to load SEGY data, including the X and Y coordinates, of each trace, the line number, the CDP (or crossline) number, the computer TYPE of those values (e.g. 4 byte integer vs. floating point IBM…), the coordinate scale factor, the first time sample, and the time and distance units. With the exception of the byte locations, most of these variables are well defined. However, be aware that it is common for commercial interpretation software to scale the x and y locations to the correct world coordinates, but do not change the SEGY coordinate scaling value, resulting in bin sizes that may be 10 or 100 times too large. If this happens, you will need to override this value (1H).

Another way to get determine the header information is to use the SEGY header utility (1H). This utility allows you not only to view the ASCII header of your SEGY file, but also scan through headers of a number of traces in the dataset. Click Open SEGY (2A) and navigate to your SEGY file. If you have the byte locations of inline number, crossline number, and x and y coordinates, fill them in (2B). Usually you don't need more than 1000 traces to be scanned (2C). After that, click Submit (2D), and you will see the header values (2E) in order to check if the header byte location is correct. If there is no information in the ASCII header, or if those information are inaccurate, it will take some hit-and-miss guesses before you find the correct header byte locations. Common location for line no. locations are 189, 5, 9, and 17, for cdp no. locations 193, 17, 21, for cdp x locations 181, 73, 81, and for cdp y locations 185, 77, and 85. See how headacheful headers can be? Yes, this is why most large companies have exploration technologists who load the data for you. Good luck!

The AASPI data structures are identical to those designed by the Stanford Exploration Project (SEP) and adopted in most part by the Madagascar work group. As of 2012, all AASPI application programs use *aaspi_io* rather than SEP input/output routines. With this new software we can now read and write all the 2003 SEGY formats, such as 32-bit IEEE format, 16-bit integer format, ASCII headers, as well as the more common 32-bit IBM format and EBCDIC headers. These new programs and associated I/O library also circumvent all of the previous big-endian/little-endian headaches, as well as limitations to the length of the file name encountered in our previous releases. This conversion allows us to easily compile on 64-bit Linux as well as 32-bit and 64-bit Windows operating systems machines. The main difference between *aaspi_io* and SEP i/o is that *aaspi_io* will *not* support Linux pipes, since pipes is not supported by Windows. All input and output seismic data and attribute files will be in AASPI format, which at the end of the day, need to be converted back to SEG-Y format (the 2nd and 3rd tabs near the top of the panel).

We will always wish to define our vertical and horizontal measurement units. In (1I) use the pull-down arrow to select the vertical units. The default for SEG-Y time data will be 's' (for seconds), with a 2 ms sample increment stored as 2000 μs (microseconds). Depth data are more challenging. If the physical sample increment is 5 m (or 15 ft), and if we store them as 5000 mm, (or 15000 mft) then you should enter the unit 'km' (or 'kft') as the vertical unit. You would type 'm' ('ft') only if the physical 5 m (or 15 ft) sample increment is stored as 5000000 μm (15000000μft).

In the United States, almost all surveys define horizontal distances in ft. In almost every other place in the world, we will use m. However, be forewarned that you may obtain trade data from a European operator working the Gulf of Mexico with survey coordinates converted to m. There is a SEG-Y flag for this (1 for m, 2 for ft) but it is often set to 0 during processing and import/export from various packages. For this reason, the AASPI software allows you to (1J) explicitly choose the 'Horizontal units' using the drop down arrow selector.

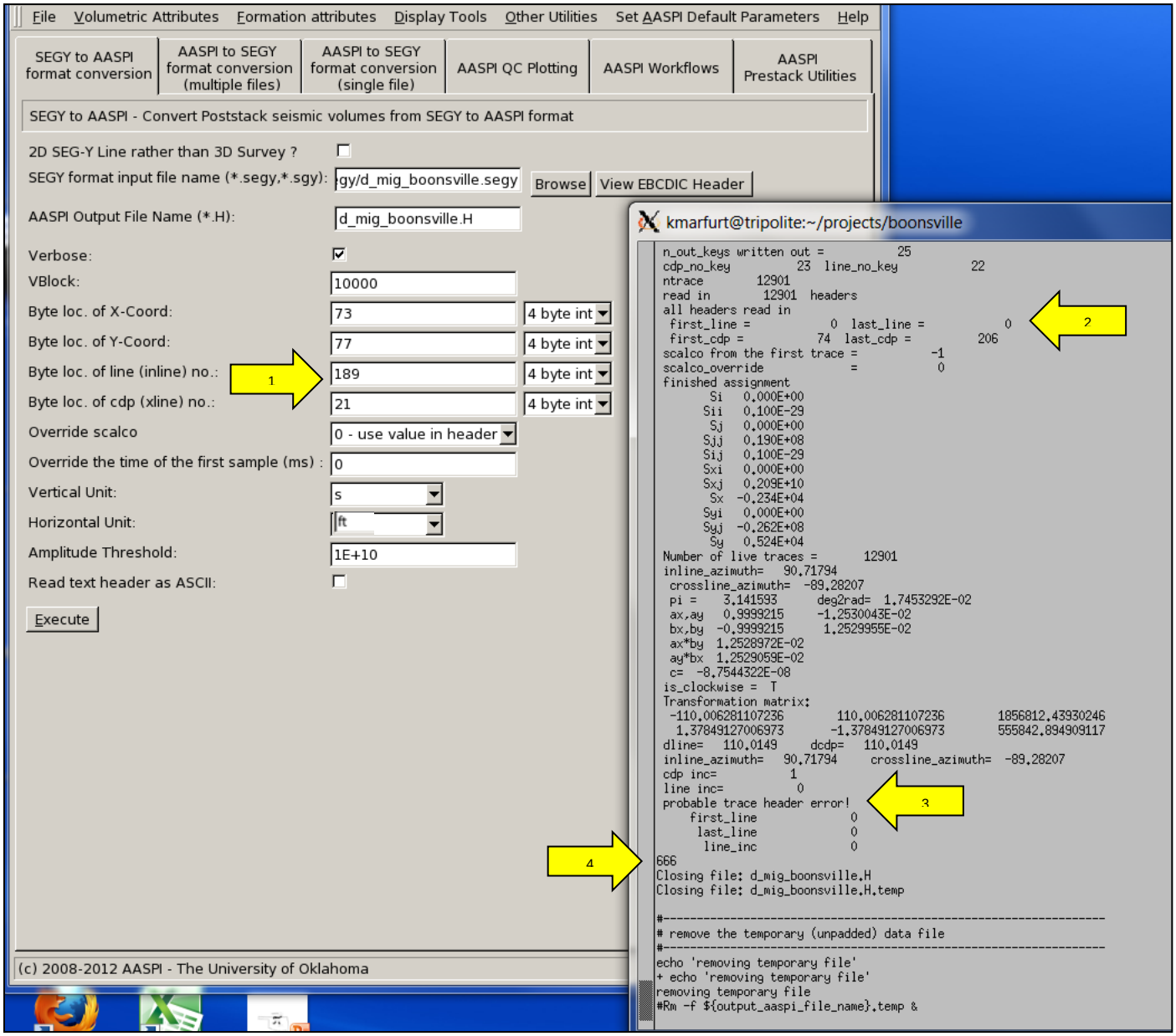
Finally, from time to time you may obtain a data set with 'glitches' in it. Ideally, such numbers are flagged somewhere along the way as 'NaN's, which stands for 'Not a Number', but it can be a cold, cruel world out there. Tape transcription, faulty disk drive controllers, or errors in seismic

processing with insufficient internal error checking can introduce bad numbers. If you have NaNs, you will need to clip your data by (1K) typing in an appropriate number (say 10 standard deviations away from 0.0 for data that have had the outliers removed). The AASPI software will detect the outliers and attempt to interpolate their values from adjacent time or depth samples.

In general, if you work as an interpreter, it is good practice to QC your data using your workstation interpretation software of choice and then to export the (potentially cropped) seismic amplitude data volume in 32-bit IBM floating point format using what your workstation defines as the SEG-Y standard. Your workstation will always store the inline no., crossline no, x, and y locations in the exact same place, such that once you have figured it out, you will have few further problems. In addition, most interpretation software will generate a histogram. You can clip the data 32-bit data to remove any glitches before exporting as SEG-Y. If you are working for a processing company, all of these features described above are normal operating procedure and fade into the background of more significant data format problems that you deal with daily.

QC and Troubleshooting Header Byte Locations

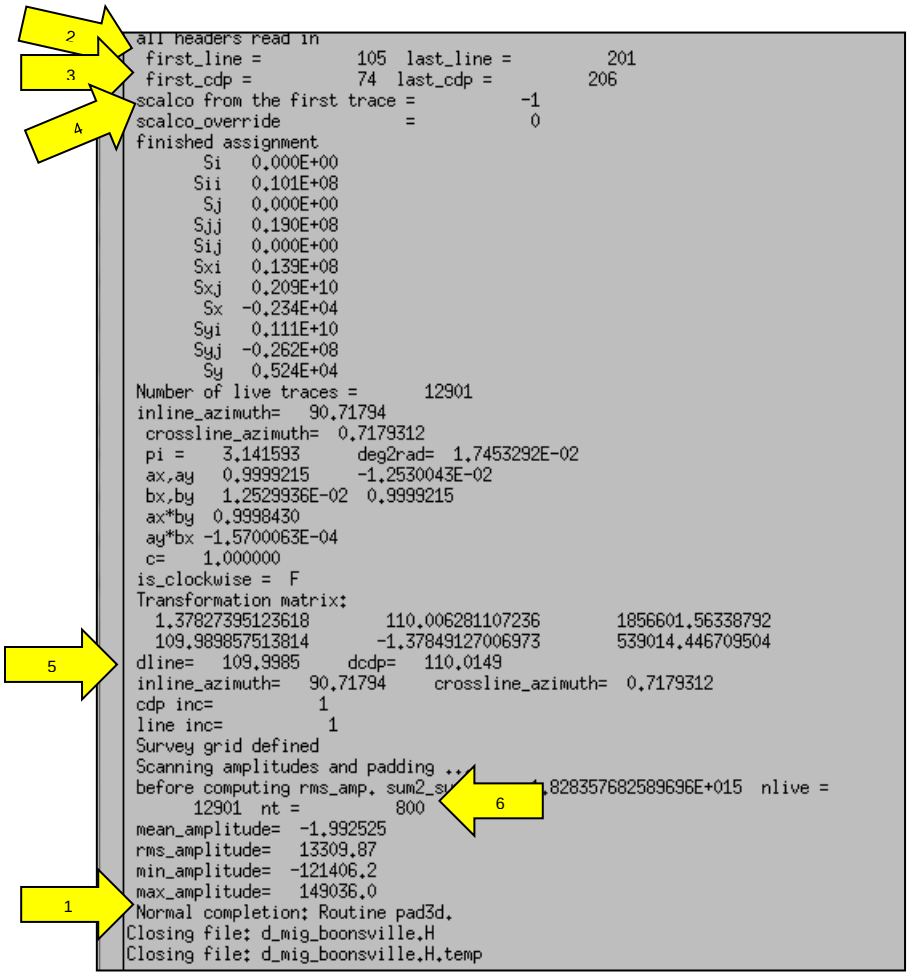
Let us first look at an example of a conversion gone badly. In this example below, we have (1) left in the SEG-Y 2002 value of 189 for *Byte loc. of line (inline) no.* and clicked *Execute*.



The output of the job will be displayed in the original (in this case, gray) *xterm* from which you launched **aaspi_util**. Note that the first and last cdp numbers are reasonable with values of 74 and 206, but that (2) the first line and last line numbers are both 0. These incorrect values result in (3) an error message. For those of you who read the Book of Revelations every night before going to bed, you will recognize that (4) a stop code of 666 is not a good thing to have, since it is the 'sign of the beast'. Whenever you see a stop code of 666, you need to look through the

output to determine if an error message has been printed out. In this case the error message reads 'probable trace header error!'

Let us go back, type in the correct byte location ('5') for the Byte loc. of line (inline) no., and click *Execute* again. The **aaspi_util** GUI sends output to the original window from which it was executed (displayed in the lower right). The GUI runs a shell script, which can be found (and modified if you wish) under the AASPI directory as *${AASPIHOME}/scripts/aaspi_segy2aaspi.sh*. For large data volumes (or for data sitting over a slow NFS connection), this program may take a while to write the results to your local disk.



The above image is a screen capture of the bottom of the dialogue that pops up in your original xterm window. Be sure you have your IT folks set up your environment so that you can monitor the progress of your jobs. (Marfurt has visited one location where this is not the case, which makes error tracking somewhat more tedious).

First look for (1) the normal completion of AASPI program **pad3d**. So far so good. Next, quality control (2) the range of lines (3) and the range of cdps. Note that (4) the data had a scalco value of -1, which using the SEGY standard definition, indicates that the data were stored in tenths of a foot. The (5) line increment (*dline*) and cdp increment (*dcdp*)are very close to the nominal 110 ft

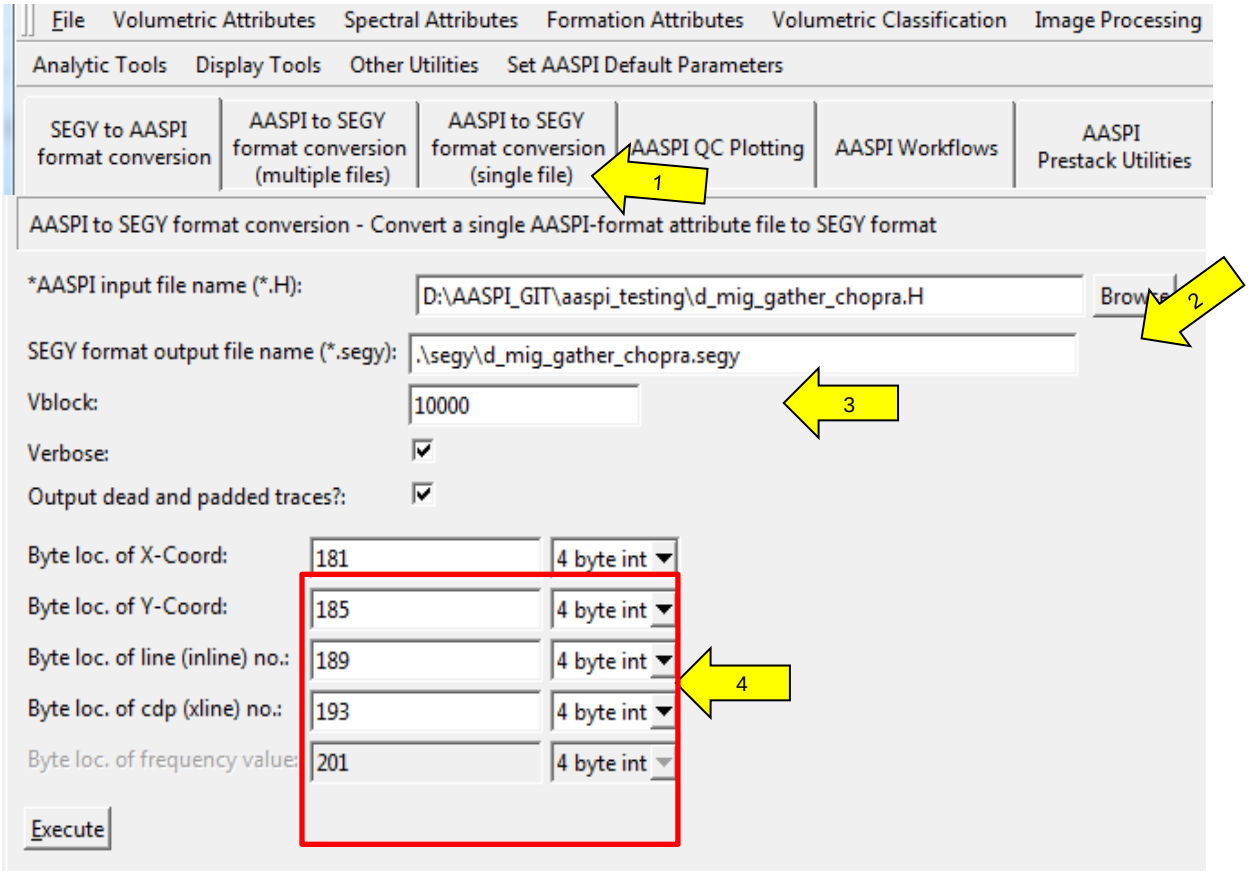
by 110 ft bin definition, so everything is as it should be The inline and crossline azimuths (directions of increasing indices) will allow us to define dip azimuth and curvature strike. If any of these values are incorrect, you will be wasting your time to do anything further since you probably will not be able to load any results into your workstation for interpretation. . The number of samples (6) is 800. Some summary statistics of the range of amplitudes follows. Finally, some simple statistics of the seismic data volume itself (min, max, and RMS amplitudes) will be used later to avoid truncation errors in our add-drop semblance and covariance matrix calculations. If any of these numbers is a NaN (Not a Number), your conversion did not occur properly and you should check the input file formats.

The data conversion is a two-step process. The first step converts the 2D SEG-Y data format to a 2D AASPI data format using program **aaspi_segy_read**. Then program **pad3d** reads in the data and headers of the 2D AASPI data format and determines where the data lie on a grid. After considerable work by Tim Kwiatkowski, we feel we can now read in data that have been stored with increasing or decreasing line and cdp increments, and data that are either padded or unpadded with dead traces to generate a rectangular grid.

Program **pad3d** also searches for dead traces. We have found that many data volumes do not follow the 2002 SEG-Y standard for the trace-id, which should read 1 for a live trace, 2 for a dead trace, and 3 for a padded trace. Many times we will find the number 0. Program **pad3d** scans each trace to determine if it is live, dead, or has been padded by the program, assigning appropriate x, y locations for padded data. The program also scans each trace from the top and bottom and detects the location of the first non-zero value, allowing subsequent programs to preserve mutes. (Certain Canadian data volumes required muting data below a given time level for resource ownership reasons).

Poststack Data Conversion from AASPI to SEGY format (single file)

To convert an AASPI-formatted data into SEGY, go to *AASPI to SEGY format conversion (single file)* tab (1) on the aaspi_util GUI. Simply browse (2) and select an AASPI formatted file to be converted (in this case, d_mig_gather_chopra.H). The output file name (3) is automatically set to have the same name with the input file but with extension ".segy" instead of ".H". The file name also indicate that the output would be located in the "segy" directory within the current working directory. Specify header byte locations (4) of the output if needed, and click Execute.

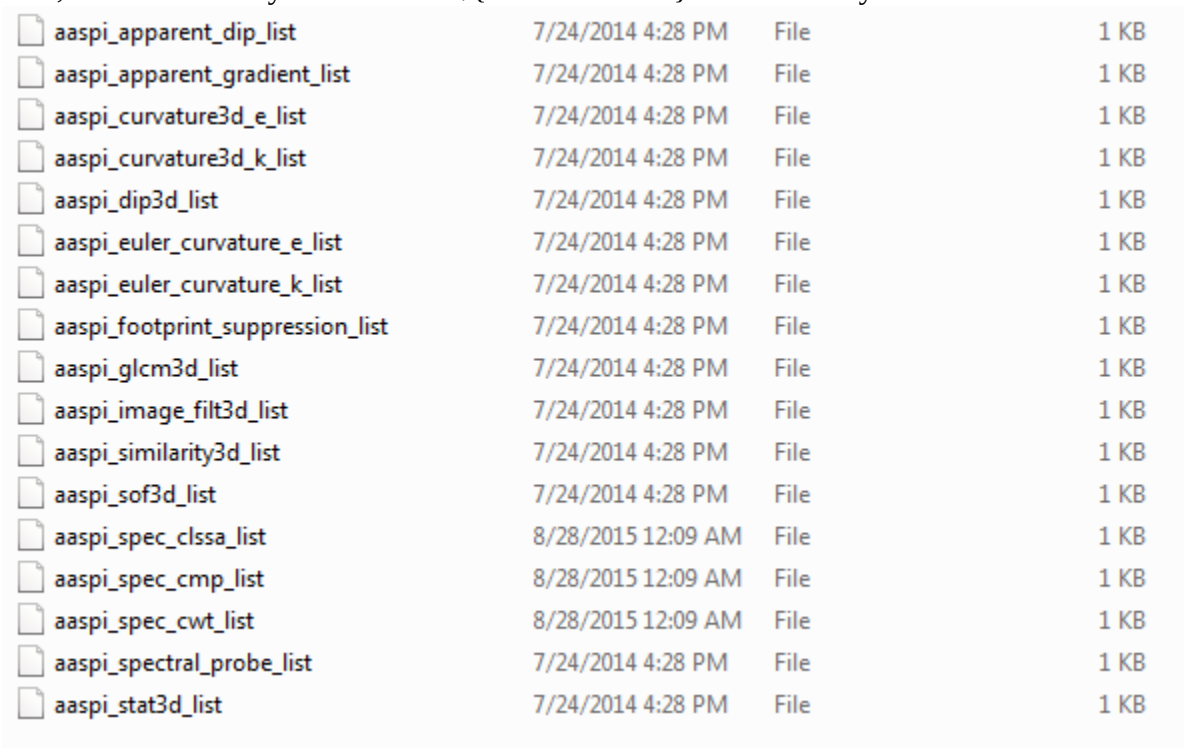


Poststack Data Conversion from AASPI to SEGY format (multiple files)

While this conversion can be done for any file, you will see as you read the documentation that there are a great many attributes. Converting files one at a time is not only tedious, but can introduce naming or typing errors. For this reason, we provide a multiple file conversion utility. Within the AASPI software, the output file names are not arbitrary, but rather controlled by a combination of GUIs and shell scripts. Typical attribute files will have the format of "attribute name" followed by an underscore, followed by "unique project name" followed by another hyphen, followed by a "suffix" followed by ".H". Thus, my inline reflector dip attribute for the Boonsville data volume will be named (in dip3d documentation) *inline_dip_boonsville_0.H* where 'boonsville' is my unique project name and the "0" indicates that I considered this to be my 0^{th} or baseline computation. Previous releases of AASPI would simply convert this file to SEGY with the name *inline_dip_boonsville_0.segy*. Such long file names caused data base headaches with some of sponsor installations.

Our September 2012 AASPI release provides the ability to rename the attribute files in manner more consistent with your environment. Many of our sponsors use oracle and other data bases with a predefined naming convention. Some of the older interpretation software (such as Geoframe) may be relatively limited in the number of characters a file name can have. To address these issues, we have constructed a GUI that facilitates this naming strategy.

As with the default parameters defined above, and indeed with the interface between the GUIs and the python scripts, everything is controlled by intermediate files. The use of files (rather than command line arguments) facilitates moving our software across the Linux/Windows OS. In this case, the files actually reside in the ${AASPIHOME}/lists directory and have the form *.list



aaspi_apparent_dip_list	7/24/2014 4:28 PM	File	1 KB
aaspi_apparent_gradient_list	7/24/2014 4:28 PM	File	1 KB
aaspi_curvature3d_e_list	7/24/2014 4:28 PM	File	1 KB
aaspi_curvature3d_k_list	7/24/2014 4:28 PM	File	1 KB
aaspi_dip3d_list	7/24/2014 4:28 PM	File	1 KB
aaspi_euler_curvature_e_list	7/24/2014 4:28 PM	File	1 KB
aaspi_euler_curvature_k_list	7/24/2014 4:28 PM	File	1 KB
aaspi_footprint_suppression_list	7/24/2014 4:28 PM	File	1 KB
aaspi_glcm3d_list	7/24/2014 4:28 PM	File	1 KB
aaspi_image_filt3d_list	7/24/2014 4:28 PM	File	1 KB
aaspi_similarity3d_list	7/24/2014 4:28 PM	File	1 KB
aaspi_sof3d_list	7/24/2014 4:28 PM	File	1 KB
aaspi_spec_clssa_list	8/28/2015 12:09 AM	File	1 KB
aaspi_spec_cmp_list	8/28/2015 12:09 AM	File	1 KB
aaspi_spec_cwt_list	8/28/2015 12:09 AM	File	1 KB
aaspi_spectral_probe_list	7/24/2014 4:28 PM	File	1 KB
aaspi_stat3d_list	7/24/2014 4:28 PM	File	1 KB

If I *edit* one of *the aaspi_dip3d_list* I note that it consists of two identical columns:

```
inline_dip          inline_dip
crossline_dip           crossline_dip
dip_magnitude           dip_magnitude
dip_azimuth         dip_azimuth
```

The column on the left will not be changed by the GUI and will form the root word if the AASPI format files, which typically have the form

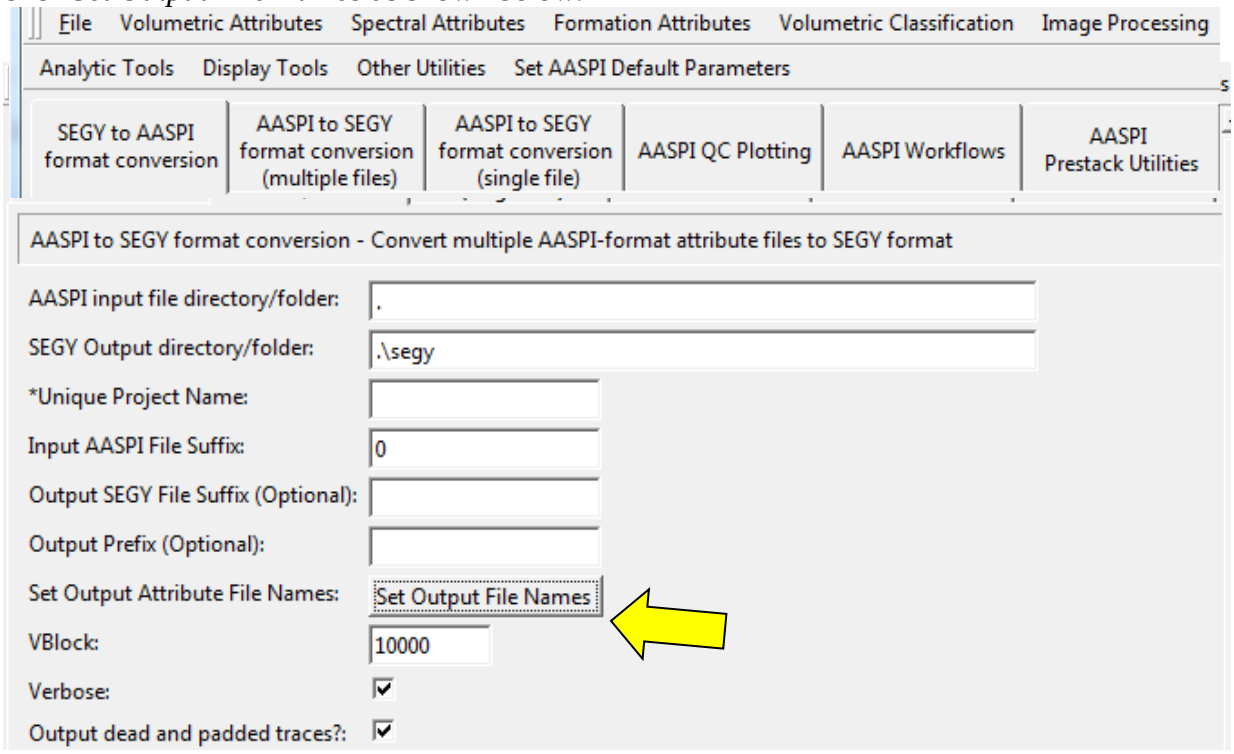
${root_left}_${unique_project_name}_${suffix}.H .

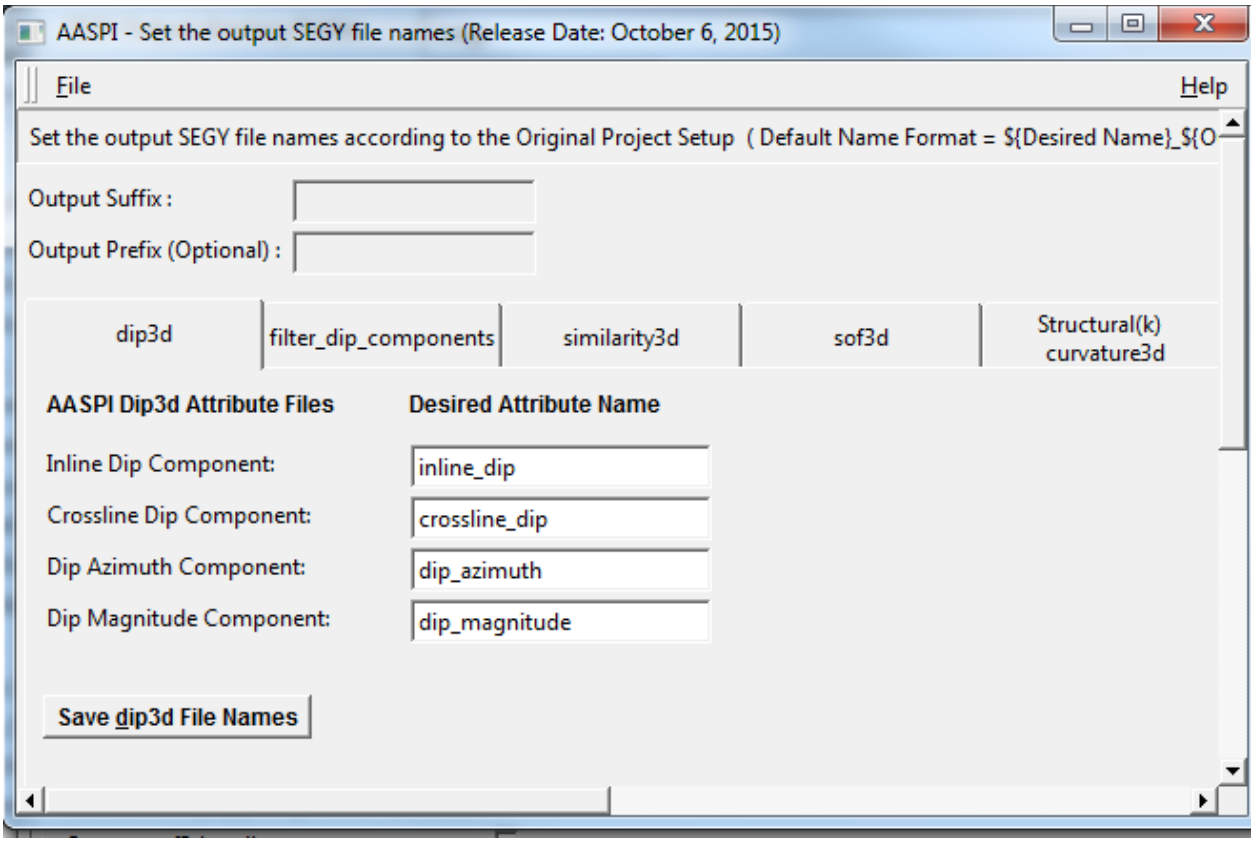
The column on the right *can* be changed. By default, the corresponding output file will have the form of previous AASPI releases:

${root_right}_${unique_project_name}_${suffix}.segy .

However, in the GUI, one can not only modify the right hand column, but also add a user-defined output prefix and output suffix (either of which may be blank). For instance, several Geoframe users require the jobname to be the leading characters in the file name. If there is a 16-character name limit, then the AASPI name needs to be shortened.

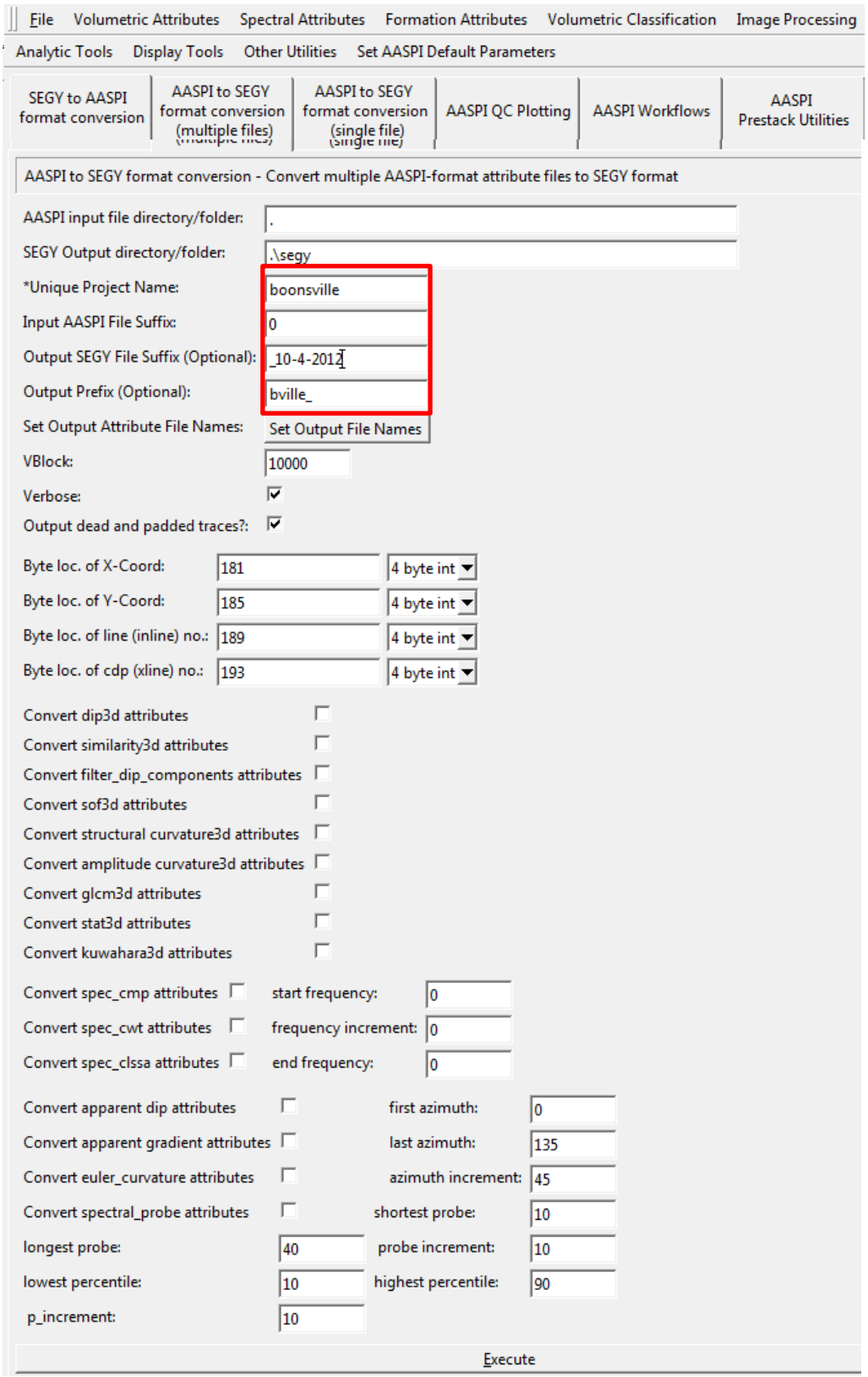
To set this up with GUI, choose the *AASPI to SEGY Conversion (Multiple Files) tab*, and the click *Set Output File Names* as shown below:





Click each of the programs for which you wish to change the names. Ideally, you will only want to do this once and the person doing it will place it in the ${AASPIHOME}/lists directory so that everyone uses the same convention. For this exercise, I did it in my home directory and generated the new *aaspi_dip3d_list* file:

```
[kmarfurt@tripolite ~]$ cat aaspi_dip3d_list
inline_dip=IL
crossline_dip                   XL
dip_magnitude                   DM
dip_azimuth                     DA
inline_dip_median_filt          IL_MED
crossline_dip_median_filt       XL_MED
dip_magnitude_median_filt       DM_MED
dip_azimuth_median_filt         DA_MED
inline_dip_lum_filt             IL_LUM
crossline_dip_lum_filt          XL_LUM
dip_magnitude_lum_filt          DM_LUM
dip_azimuth_lum_filt            DA_LUM
```


File
Volumetric Attributes
Spectral Attributes
Formation Attributes
Volumetric Classification
Image Processing
Analytic Tools
Display Tools
Other Utilities
Set AASPI Default Parameters
SEGY to AASPI format conversion
AASPI to SEGY format conversion (multiple files)
AASPI to SEGY format conversion (single file)
AASPI QC Plotting
AASPI Workflows
AASPI Prestack Utilities
AASPI to SEGY format conversion - Convert multiple AASPI-format attribute files to SEGY format
AASPI input file directory/folder: .
SEGY Output directory/folder: .\segy
*Unique Project Name: boonsville
Input AASPI File Suffix: 0
Output SEGY File Suffix (Optional): _10-4-2012
Output Prefix (Optional): bville_
Set Output Attribute File Names: Set Output File Names
VBlock: 10000
Verbose:
Output dead and padded traces?:
Byte loc. of X-Coord: 181 4 byte int
Byte loc. of Y-Coord: 185 4 byte int
Byte loc. of line (inline) no.: 189 4 byte int
Byte loc. of cdp (xline) no.: 193 4 byte int
Convert dip3d attributes
Convert similarity3d attributes
Convert filter_dip_components attributes
Convert sof3d attributes
Convert structural curvature3d attributes
Convert amplitude curvature3d attributes
Convert glcm3d attributes
Convert stat3d attributes
Convert kuwahara3d attributes
Convert spec_cmp attributes
Convert spec_cwt attributes
Convert spec_clssa attributes
start frequency: 0
frequency increment: 0
end frequency: 0
Convert apparent dip attributes
Convert apparent gradient attributes
Convert euler_curvature attributes
Convert spectral_probe attributes
first azimuth: 0
last azimuth: 135
azimuth increment: 45
shortest probe: 10
longest probe: 40
probe increment: 10
lowest percentile: 10
highest percentile: 90
p_increment: 10
Execute

I had run this job before with the *Unique Project Name* of *boonsville* and a *Suffix* of *0*. I have to make sure those are EXACTLY THE SAME with other programs I ran before. Otherwise it will not work. I set the Output *Prefix* to be "*bville_*" and the Output Suffix to be "*_10-4-12"* (project's date). I put a checkmark in next to *Convert dip3d attributes* and then click *Execut*e. The conversion completes and I obtain the following files in my segy subdirectory: