IRIS PASSCAL INSTRUMENT CENTER

GENERATING A PH5 ARCHIVE FOR ACTIVE SOURCE AND MIXED MODE EXPERIMENTS

For PASSCAL Platforms: Mac OSX & Linux
Version 2017:342

This document was written and produced by the
PASSCAL Data Group

Last revision 12/08/2017 IRIS PASSCAL Data Group

Please send your feedback, questions, and comments to: data_group@passcal.nmt.edu

Table of Contents

Introduction

PH5 is the PASSCAL recommended archival format for active, passive, and mixed source data sets. This program uses a hierarchal data format, version 5 (HDF5) data model, which has several strengths including:

- Extensible and portable, allowing it to evolve and operate on a variety of platforms through a number of programming interfaces
- Self-describing and allows direct access to parts of the file without the need to parse the entire file.
- Has been in used for two decades and is utilized in a variety of disciplines, from aerospace to the medical field.

PH5 is a unified archiving format designed to make archival and retrieval of data as simple as possible. Because the metadata and data are kept separate, the architecture allows for quick and easy updating of the metadata without having to reprocess the entire data set.

PH5 is built using multiple HDF5 files. The complete set contains a master.ph5 file and a number of mini.ph5 files. The master contains much of the metadata as well as links to the minis, which contain the actual trace data.

The archiving of your data fulfills the principle investigator (PI) responsibilities defined in the PASSCAL Data Delivery Policy (https://www.passcal.nmt.edu/content/general-information/policy/data-delivery-policy).

Additional documentation can be found on our website: https://www.passcal.nmt.edu/content/data-archiving/documentation/active-source

Please take a moment to thoroughly review this guide before you start. If you have any questions please contact: data_group@passcal.nmt.edu.

Within this document:

Headers, general scripts and commands are in bold.
Command-line usage is highlighted yellow
GUI options and menus are highlighted gray
Standard output is italicized.
URLs and email addresses are blue.

Pre-archiving steps and installing PH5

Prior to starting this archival process you should contact the Data Group and acquire, complete, and/or review:

1. On the PASSCAL website (http://www.passcal.nmt.edu) log in to your account. Once you have logged in, go to the PI home page (http://www.passcal.nmt.edu/pihomepage), select your current experiment and then click on the “Mobilize Project” button. Completion of this form sends a request to the DMC for an assembled ID and network code for your experiment.

Note: This form alerts the IRIS Data Management Center (DMC) of your temporary network and sets up the infrastructure needed for the DMC to accept your data.

2. Review the PASSCAL Data Delivery Policy: http://www.passcal.nmt.edu/content/general-information/policy/data-delivery-policy

3. The installation process and required files to set up PH5 can be found at: https://github.com/PIC-IRIS/PH5/wiki

If you encounter problems with the PH5 software, please contact data_group@passcal.nmt.edu for assistance.

Steps in brief

- Create an organized directory structure for your data
- Load raw data into PH5
- Create an experiment table
- Create/modify metadata tables
- Load experiment and metadata tables
- Check geometry using Google Earth
- Create and load other necessary tables
- Test PH5 by outputting data

First steps: activating PH5 and organizing your data

Activating the PH5 environment

PH5 is run from an Anaconda environment that you created when you installed PH5. You need to be in bash in order to activate the PH5 environment. In order to run PH5 commands, you need to activate the PH5 environment:

```
>> source activate ph5
```

*Note: If you are on a shared server, you might need to load Anaconda before activating the PH5 environment:

```
>> module load anaconda
```

After activating the PH5 environment, you should have (ph5) before your bash prompt in your terminal window.

In order to deactivate the PH5 environment, you need to run a similar command:

```
>> source deactivate ph5
```

Creating an organized directory structure for your data

PASSCAL suggests using the following directory structure. Start by creating a main directory for the project. Create a subdirectory for your raw data and copy the raw data files into it. If you have multiple types of instrumentation, create a separate raw directory for each instrument:

```
>> mkdir RAW-Texans – put all Texan TRD files here
```

```
>> mkdir RAW-RT130 – put all RT130 ZIP files here
```

Next, create a PH5 directory:

```
>> mkdir PH5 – this is where your PH5 master and mini files will be created
```

Also, create a metadata directory. All of the metadata files for your data will be placed in this directory.

```
>> mkdir METADATA
```

Loading raw data into PH5

Make a list of all the raw data files and their full path. This can be done from the command line from you main project directory by running the following command:

```
>> ls –d $PWD/RAW-Texans/*.TRD > raw_list
```

This command will create a text file called raw_list in your main project directory. Open this file and make sure that only the TRD files are listed. If you have multiple types of instrumentation (ie., Texans and RT130s), add the paths to the raw data for each additional instrumentation type to this list as needed.

If you have 1C or 3C nodes, additional steps are required to load your data. Please read the "Preparing to load node data with pforma" section. If you do not have any node data, skip to the "Loading data with pforma" section.

Preparing to load node data with pforma

Processing node data with **pforma** requires a few additional steps.

Create more descriptive file names:
The default file names created by the node server are functional but are not descriptive enough for effective troubleshooting. **pforma** requires more descriptive file names that include the array, station, and serial number of the node in the file name.

Original node file names:

```
        /storage/SanBernardinoBasin/SanBernardinoBasin201710/RAW-node/10.fcnt
        /storage/SanBernardinoBasin/SanBernardinoBasin201710/RAW-node/11.fcnt
        /storage/SanBernardinoBasin/SanBernardinoBasin201710/RAW-node/12.fcnt
```

Create a list of the original file names in the main project directory with the absolute path to each file:

```
>> ls –d $PWD/RAW-node/*.fcnt > node-list.txt
```

Run **unsimpleton** to create soft links with more descriptive names:

```
>> unsimpleton –f node-list.txt –d Data-nodes
```

Where:
node-list.txt is the list of original node file names with their absolute path
Data-nodes is the directory where the soft links will be located

After running **unsimpleton**, you should have a new directory in your main directory named *Data-nodes*:

```
>> ls Data-nodes
        PIC_1_101_1897.0.0.rg16
        PIC_1_102_1896.0.0.rg16
        PIC_1_103_1893.0.0.rg16
```

After running **unsimpleton**, make a new list of raw file names using the soft links created in the Data-nodes folder. In **pforma**, select the new list with more descriptive names for the 'RAW list file'. If you have multiple instrument types (ie RT130s and nodes) you can append the list with the new more descriptive file names for the node data to the list of all other raw files.

Combine SEG-D traces for faster processing
Nodes can write data to little tiny windows (30 sec) in the raw data, which makes processing in PH5 very slow. You will need to determine what the event windows are in the raw SEG-D data and set **pforma** to combine traces to get ~30 minute windows. If you already know how long the SEG-D traces in your data files are, you can skip this step and use that information to determine how many traces you will combine to get 30 minute windows.

You can determine how many samples are in each SEG-D trace using **dumpfair**. The number of samples is in the Trace Header Extension 1 as samples_per_trace.

To see the number of samples:

```
>> dumpfair 1.fcnt | grep samples_per_trace | head -1
```

Once you have identified the number of samples per trace, you can use this combined with the sample rate to determine the length of each trace in minutes.

For example, if each trace has 75,000 samples at a sample rate of 250sps:

75,000 samples/250 sample per sec=300 sec/60 sec per min=5 minutes

Calculate # of windows need to combine to get ~30 minute window:

- with 5 minutes windows, need to combine 6 traces to get 30 minute windows

In **pforma**, you will need to specify both how many SEG-D traces to combine in the PH5 as well as the UTM Zone. You can set these by selecting the appropriate option in the File dropdown menu in **pforma**.

Number of traces to combine: File > Combine # of SEGD traces in ph5

Set UTM zone: File > UTM Zone

If you have node data from multiple UTM zones, you will need to run **pforma** for the data for each UTM zone separately in order to correctly map the node locations into the PH5 archive.

Proceed with the steps outlined in the “Loading data with pforma” section.

Loading data with pforma

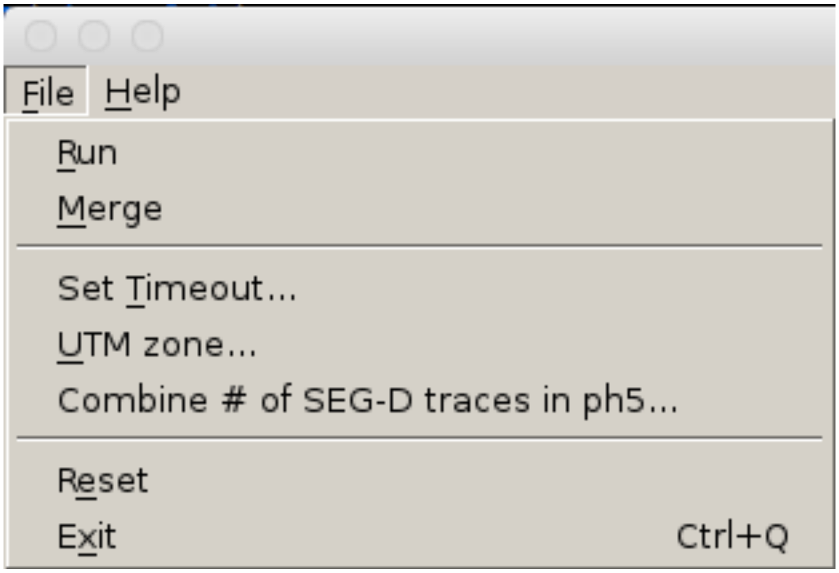
Data is loaded into PH5 using a GUI program called **pforma**. **pforma** takes all of the raw data files referenced in the list you created earlier and divides them up to multi-thread the loading process to make it faster. **pforma** will use all available cores on your system. Currently, this is not configurable.

pforma creates multiple subdirectories in the PH5 directory. After loading all of the data using **pforma**, you will merge the resulting subdirectories into a complete PH5.

To launch the **pforma** GUI, type into the command line:

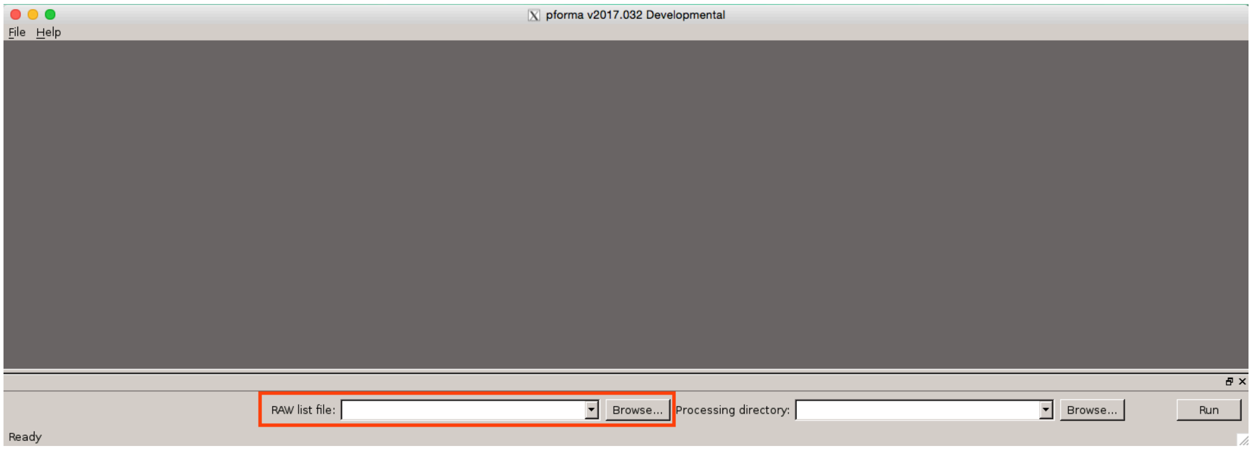
```
>> pforma
```

The GUI that opens will allow you to load data into a PH5 archive using the raw file list that you created earlier. In the main **pforma** window you can select the list of raw data files that you created and the processing directory where the PH5 archive will be created. The main **pforma** menu (under File) has several useful options.

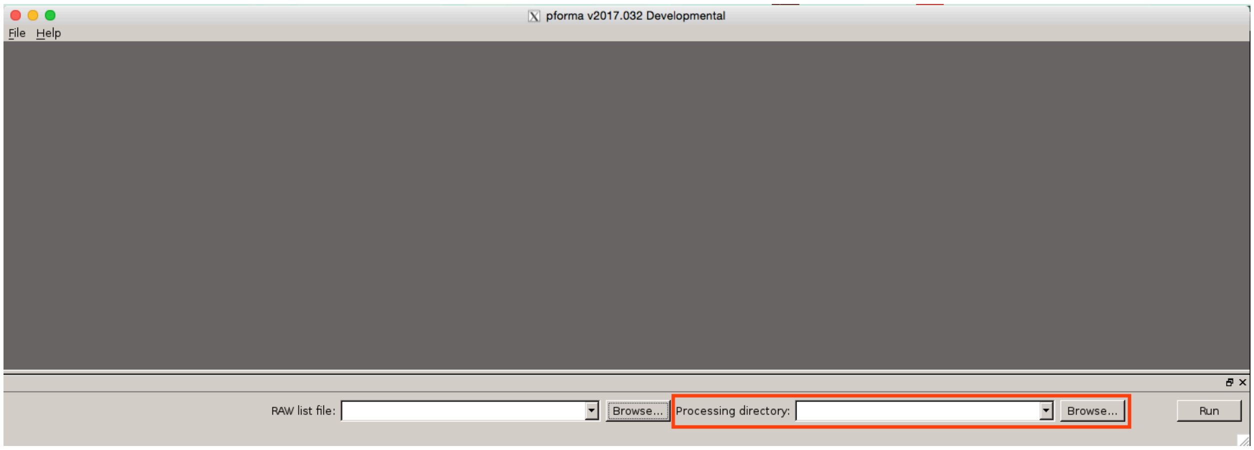


Run – Run the load process after selecting the raw file list and the processing directory
Merge – Combine the data from the sub-directories created by loading data with **pforma** into a final, complete PH5 archive in the Sigma directory
Set Timeout – Change the timeout from the default (default is 2000 seconds)
UTM Zone – Set the appropriate UTM zone for processing node data
Combine # of SEG-D traces in ph5 – Sets the number of SEG-D traces that will be combined into a single trace in PH5 for node data. Nodes can write out the data in very short traces which increases processing time when loading data and retrieving data from the PH5 archive
Reset – Reset settings to default
Exit – Close **pforma**

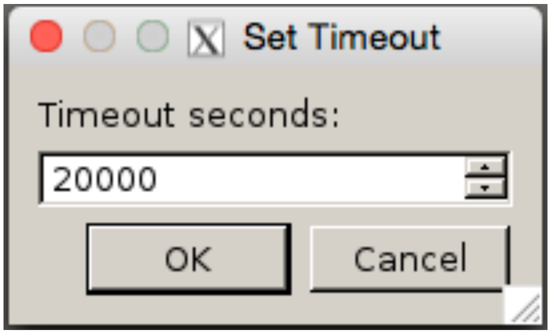
Select the Browse button to the right of the “RAW list file” field and browse for the file list you created in your main project directory. This list should include all raw data files that will be included in your PH5 archive.



Select the Browse button to the right of the “Processing directory” field and browse for the PH5 directory you created earlier.

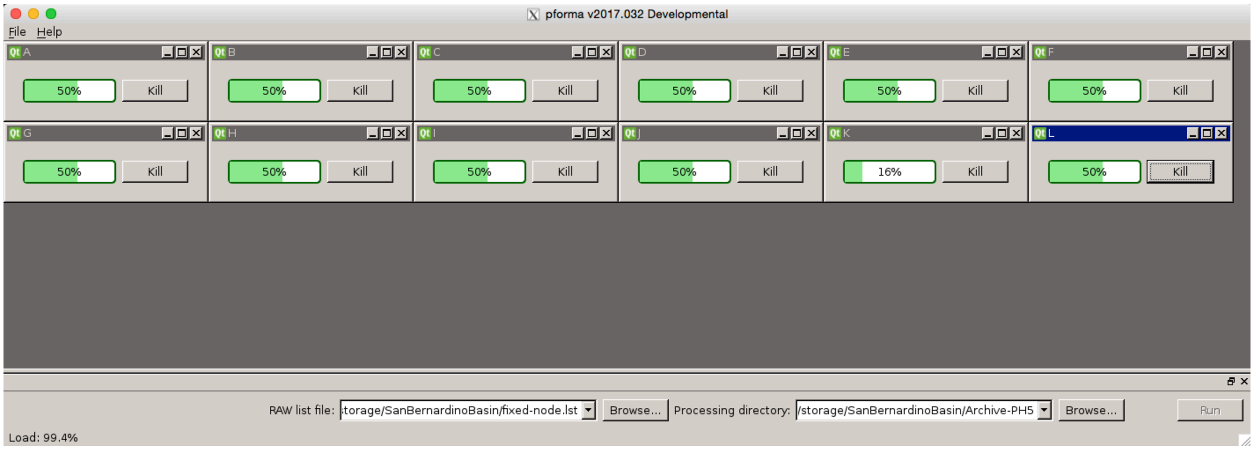


Increase timeout: File > Set Timeout. For Texans, the default timeout (2000 seconds) should be sufficient but when processing data from RT130s, increase the timeout by factor of 10.



Timeout seconds: 20000

Click Run and wait for all the bars to show 100%.

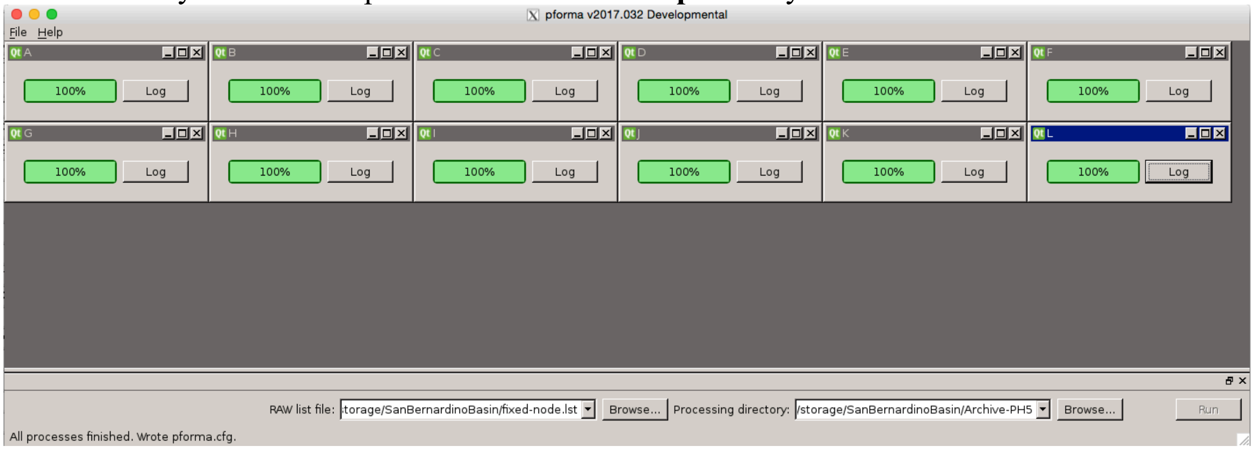


RAW list file: :torage/SanBernardinoBasin/fixed-node.lst

Processing directory: /storage/SanBernardinoBasin/Archive-PH5

Load: 99.4%

When the processes are complete, the Kill button will change to Log. You can select the Log button to see a summary of what was processed. Do not close **pforma** yet.



RAW list file: :torage/SanBernardinoBasin/fixed-node.lst

Processing directory: /storage/SanBernardinoBasin/Archive-PH5

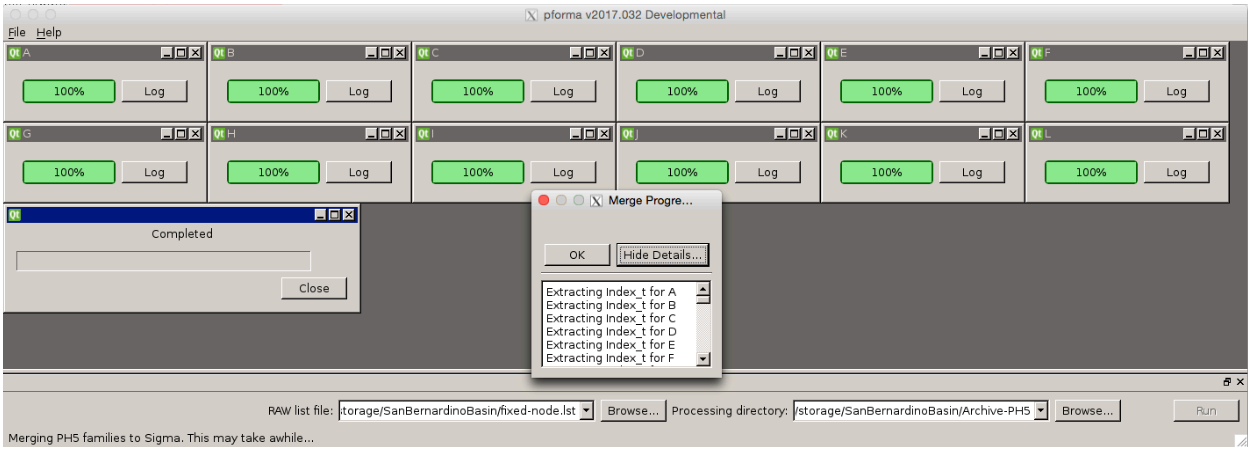
All processes finished. Wrote pforma.cfg.

If **pforma** fails to process a data file and stops, the progress bar for that process will turn orange (indicating there was a problem, but **pforma** finished running) or red (**pforma** stopped running without finishing processing the data). In each case, click on the Log button to the right of the progress bar for that process to determine what went wrong. Please send the information in this log file to data_group@passcal.nmt.edu if you are not able to resolve the problem.

Common problems include:
Timeout: **pforma** times out when processing a specific data file. If this occurs, select File > Set Timeout and increase the timeout. Delete all of the subdirectories in the PH5 processing directory and run **pforma** again.
Corrupted RT130 data: **pforma** stops running on a specific ZIP file due to the data file being corrupted. If this occurs, check the log for that process to identify the problem data. Move that zip file to another folder and remove the entry from the raw file list. Delete all of the subdirectories in the PH5 processing directory and run **pforma** again. Email data_group@passcal.nmt.edu for recommendations for dealing with the corrupted data.

Click File on the main GUI, and select Merge, and wait for this to complete as well. This creates combined PH5 master and mini files that can be found in a subdirectory called "Sigma" within the PH5 processing directory you specified. Once the merge is complete, a window will pop up to let you know it is finished. You can close **pforma**.

If you close **pforma** before merging the PH5, restart **pforma** from the command line. You will need to re-identify the raw file list (even though the data are already processed) and processing directory. Select the Browse button to the right of the "RAW list file" field and browse for the file list you created in your main project directory. Select the Browse button to the right of the "Processing directory" field and browse for the PH5 directory that you selected for **pforma** to use earlier. Click File on the main GUI, and select Merge, and wait for the merge to complete.

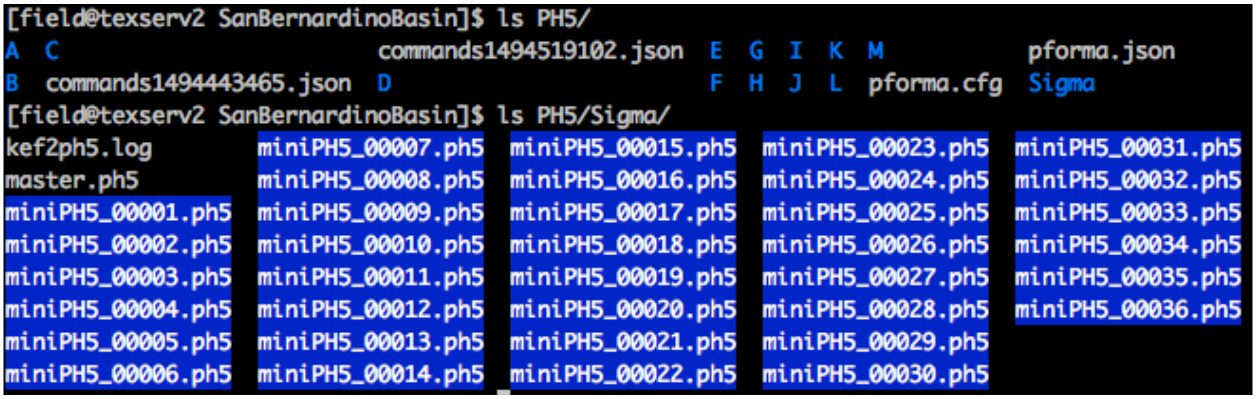


```
Extracting Index_t for A
Extracting Index_t for B
Extracting Index_t for C
Extracting Index_t for D
Extracting Index_t for E
Extracting Index_t for F
```

RAW list file: :torage/SanBernardinoBasin/fixed-node.lst Browse... Processing directory: /storage/SanBernardinoBasin/Archive-PH5 Browse...

Merging PH5 families to Sigma. This may take awhile...

pforma creates a directory for each subset of data that it loads. In the example above, **pforma** created 13 subdirectories in the processing directory named A through M. After merging the PH5 archive, the Sigma directory where the final PH5 archive is located is created.



```
[field@texserv2 SanBernardinoBasin]$ ls PH5/
A  C                        commands1494519102.json  E  G  I  K  M        pforma.json
B  commands1494443465.json  D                        F  H  J  L  pforma.cfg  Sigma
[field@texserv2 SanBernardinoBasin]$ ls PH5/Sigma/
kef2ph5.log        miniPH5_00007.ph5  miniPH5_00015.ph5  miniPH5_00023.ph5  miniPH5_00031.ph5
master.ph5         miniPH5_00008.ph5  miniPH5_00016.ph5  miniPH5_00024.ph5  miniPH5_00032.ph5
miniPH5_00001.ph5  miniPH5_00009.ph5  miniPH5_00017.ph5  miniPH5_00025.ph5  miniPH5_00033.ph5
miniPH5_00002.ph5  miniPH5_00010.ph5  miniPH5_00018.ph5  miniPH5_00026.ph5  miniPH5_00034.ph5
miniPH5_00003.ph5  miniPH5_00011.ph5  miniPH5_00019.ph5  miniPH5_00027.ph5  miniPH5_00035.ph5
miniPH5_00004.ph5  miniPH5_00012.ph5  miniPH5_00020.ph5  miniPH5_00028.ph5  miniPH5_00036.ph5
miniPH5_00005.ph5  miniPH5_00013.ph5  miniPH5_00021.ph5  miniPH5_00029.ph5
miniPH5_00006.ph5  miniPH5_00014.ph5  miniPH5_00022.ph5  miniPH5_00030.ph5
```

If you need to add more files you can add them by repeating the above steps, then delete or move the Sigma directory and re-merge the PH5.

Adding more data using the command line

If you have already built your PH5 archive, but need to add additional raw data, you can use the command line from within the Sigma folder:
125atoph5 –f <file_list> -n master.ph5 >& 125a2ph5.out

The file list should contain the full path for the additional raw data files. Use **125atoph5** to load Texan raw data, **130toph5** for RT130 data, and **segdtoph5** for nodes.

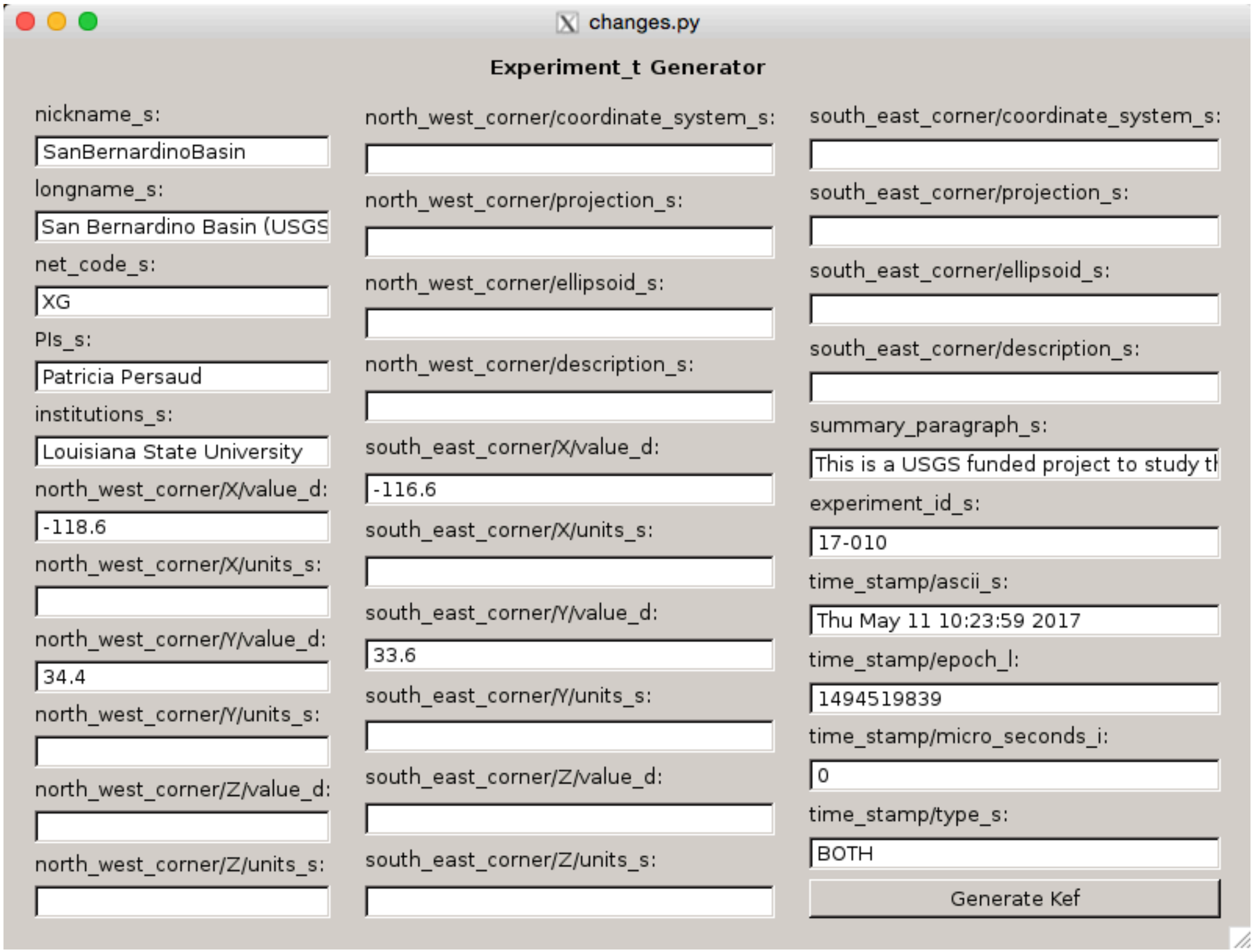
Creating an experiment table

Kitchen exchange format files, or kefs, are used to import metadata into the PH5 file. In the metadata directory, build the experiment summary kef with the GUI:

>> experiment_t_gen

Required fields include nickname, long name, net code, PIs, institutions, coordinates for the bounding box, summary paragraph, and experiment ID (assembled ID).

If you have not already submitted a mobilization form to the DMC and received an assembled ID and network code for your experiment, please visit the PASSCAL PI Home Page (http://www.passcal.nmt.edu/pihomepage) and select the "Mobilize Project" button for your experiment. An assembled ID and network code are required for the experiment summary kef.



nickname_s: SanBernardinoBasin
longname_s: San Bernardino Basin (USGS
net_code_s: XG
PIs_s: Patricia Persaud
institutions_s: Louisiana State University
north_west_corner/X/value_d: -118.6
north_west_corner/X/units_s:
north_west_corner/Y/value_d: 34.4
north_west_corner/Y/units_s:
north_west_corner/Z/value_d:
north_west_corner/Z/units_s:

north_west_corner/coordinate_system_s:
north_west_corner/projection_s:
north_west_corner/ellipsoid_s:
north_west_corner/description_s:
south_east_corner/X/value_d: -116.6
south_east_corner/X/units_s:
south_east_corner/Y/value_d: 33.6
south_east_corner/Y/units_s:
south_east_corner/Z/value_d:
south_east_corner/Z/units_s:

south_east_corner/coordinate_system_s:
south_east_corner/projection_s:
south_east_corner/ellipsoid_s:
south_east_corner/description_s:
summary_paragraph_s: This is a USGS funded project to study t
experiment_id_s: 17-010
time_stamp/ascii_s: Thu May 11 10:23:59 2017
time_stamp/epoch_l: 1494519839
time_stamp/micro_seconds_i: 0
time_stamp/type_s: BOTH

nickname_s: Experiment nickname. This is used to name SEG-Y files generated by ph5toevt. We recommend using an experiment nickname without any spaces.
longname_s: The long name of the experiment. It is recommended that this match the experiment long name in the PASSCAL experiment database and on the DMC mobilization form.
net_code_s: The 2-letter FDSN network code for the experiment. If you do not have a network code for your experiment, please confirm that you have filled out the mobilization form for the experiment on the PASSCAL PI Home Page (http://www.passcal.nmt.edu/pihomepage) or contact the PASSCAL Data

Group (data_group@passcal.nmt.edu) if you did not receive a network code when you filled out the mobilization form.
PIs_s: A list of the experiment Principal Investigators.
institutions_s: The institutions with which the experiment PIs are affiliated. It is recommended that the institutions be listed in the same order as the PIs.
north_west_corner/X/value_d: Longitude in decimal degrees of the northwest corner of a bounding box for the experiment.
north_west_corner/Y/value_d: Latitude in decimal degrees of the northwest corner of a bounding box for the experiment.
south_east_corner/X/value_d: Longitude in decimal degrees of the southeast corner of a bounding box for the experiment.
south_east_corner/Y/value_d: Latitude in decimal degrees of the southeast corner of a bounding box for the experiment.
summary_paragraph_s: A brief paragraph summarizing the experiment.
experiment_id_s: The DMC assigned assembled ID for the experiment. If you do not have an assembled ID for your experiment, please confirm that you have filled out the mobilization form for the experiment on the PASSCAL PI Home Page (http://www.passcal.nmt.edu/pihomepage) or contact the PASSCAL Data Group (data_group@passcal.nmt.edu) if you did not receive an assembled ID when you filled out the mobilization form.

After filling out all of the necessary fields, select Generate Kef and save the experiment kef in the METADATA folder.

Below is a sample of a kef created by **experiment_t_gen**:

```
#Thu May 11 10:27:42 2017      2017.128
#
#   Table row 1
/Experiment_g/Experiment_t
      nickname_s = SanBernardinoBasin
      longname_s = San Bernardino Basin (USGS) - Jan 2017 nodal deployment
      net_code_s = XG
      PIs_s = Patricia Persaud
      institutions_s = Louisiana State University
      north_west_corner/X/value_d = -118.6
      north_west_corner/X/units_s =
      north_west_corner/Y/value_d = 34.4
      north_west_corner/Y/units_s =
      north_west_corner/Z/value_d =
      north_west_corner/Z/units_s =
      north_west_corner/coordinate_system_s =
      north_west_corner/projection_s =
      north_west_corner/ellipsoid_s =
      north_west_corner/description_s =
      south_east_corner/X/value_d = -116.6
      south_east_corner/X/units_s =
      south_east_corner/Y/value_d = 33.6
      south_east_corner/Y/units_s =
      south_east_corner/Z/value_d =
      south_east_corner/Z/units_s =
      south_east_corner/coordinate_system_s =
      south_east_corner/projection_s =
      south_east_corner/ellipsoid_s =
      south_east_corner/description_s =
      summary_paragraph_s = This is a USGS funded project to study the structure of the San Bernardino Basin in Southern California using a nodal deployment.
      experiment_id_s = 17-010
      time_stamp/ascii_s = Thu May 11 10:23:59 2017
      time_stamp/epoch_l = 1494519839
      time_stamp/micro_seconds_i = 0
      time_stamp/type_s = BOTH
```

Afterwards, navigate to the Sigma directory and load the resulting kef (in this example it was saved as experiment_t.kef) into the PH5 file:

```
>> keftoph5  –n master.ph5  –k  ../../METADATA/experiment_t.kef
```

*Note: Every time you load a kef using **keftoph5**, an entry is made in a kef2ph5.log file. This can be useful for checking the progress of your steps.

Receiver and event metadata tables

Receiver and shot metadata is loaded into PH5 by first creating a series of CSV (comma-separate variable) files for the receiver and shot arrays and then converting those files to kef files using the GUI program **noven**. While the columns of the CSV file may be in different orders, the format used in each field has very specific requirements.

Receiver metadata

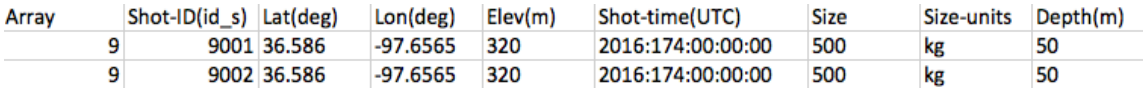
Required columns: Array, Station number(id_s), Seed station name, DAS serial number, Sensor serial number, Channel number, Sample rate, Seed channel, Latitude, Longitude, Elevation, Deploy time, Pickup time, DAS manufacturer, DAS model, Sensor manufacturer, and Sensor model.

In addition, for channels with sample rates less than 1sps (ie mass positions recorded at 0.1sps, or UH? or VH? seismic data (0.01sps and 0.1sps respectively)), a 'sample rate multiplier' column is required. For all sample rates 1 or greater, the sample rate multiplier will be set to 1. For sample rates less than 1, the sample rate column should be set to 1 and the sample rate multiplier should be 10 (for 0.1sps) or 100 (for 0.01sps).

Shot/Event metadata

Shot/event metadata is not required. If you have shot metadata, then you will also need to create a shot array table in addition to the receiver array tables you have already created. The shot table has fewer required fields than the array table.

Required columns: shot array, shot ID, latitude, longitude, elevation, shot time. Additional columns for shot size, shot size units (kg, mb, etc) and depth can be added as needed.



Array	Shot-ID(id_s)	Lat(deg)	Lon(deg)	Elev(m)	Shot-time(UTC)	Size	Size-units	Depth(m)
9	9001	36.586	-97.6565	320	2016:174:00:00:00	500	kg	50
9	9002	36.586	-97.6565	320	2016:174:00:00:00	500	kg	50

An example file for a shot array table can be found on the PASSCAL website: https://www.passcal.nmt.edu/content/data-archiving/documentation/active-source

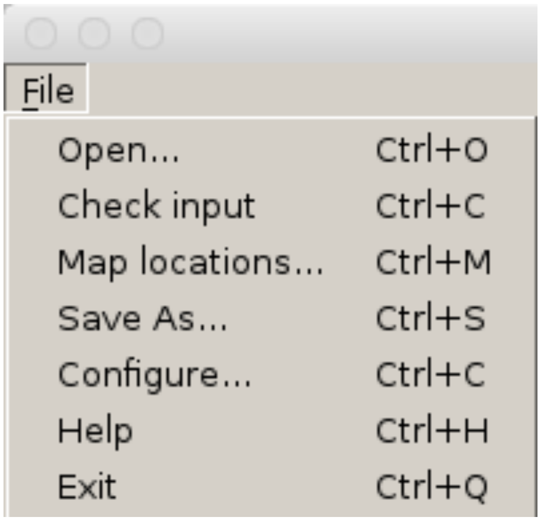
Creating kef files from metadata tables with noven

Array and event information is loaded into PH5 using a series of kef files. You can create these kef files by loading the CSV files that you just created into a GUI program called **noven**. This program loads CSV files for both arrays and events, checks the formatting of the required columns, and then outputs a kef file that can then be loaded into the master.ph5 file using **keftoph5**.

To launch the **noven** GUI, type into the command line:

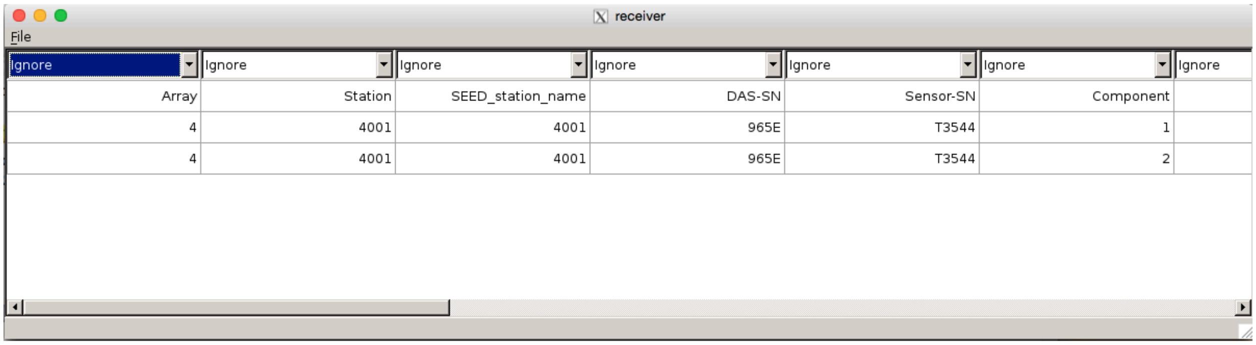
```
>> noven
```

The GUI that opens will allow you to load the CSV format metadata files that you created for the receiver and shot arrays, specify the columns, and then save the array tables as PH5 compatible kef files that can be loaded into the master.ph5 file. The main **noven** menu (under File) has several useful options.



Open – Open a receiver or event CSV file
Check input – verify that the input for each column is correctly formatted and that all required metadata are present
Map locations – create a Google Earth kml file from the currently loaded metadata table
Save As – save input out to a kef file after the input has been checked
Configure – change settings for **noven** including specifying if file is receiver or event metadata
Help – displays **noven's** help for the specified file type (receiver or event)
Exit – close **noven**

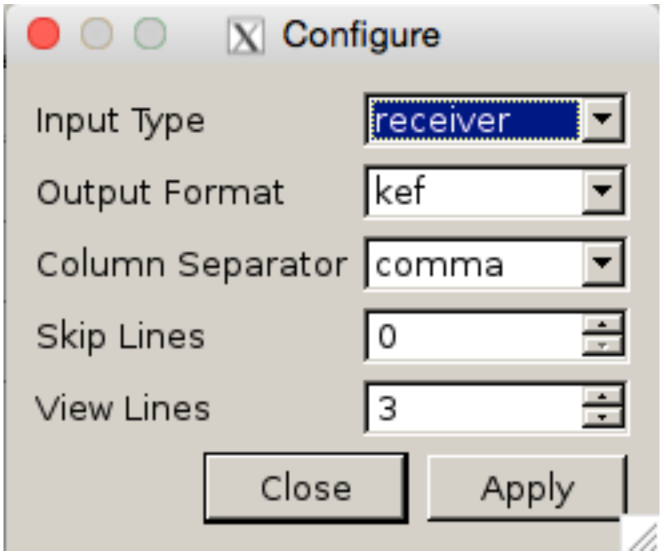
Select File > Open and navigate to the METADATA folder and select either the receiver or event CSV file that you created earlier. When you initially open a file, **noven** defaults to displaying the top 3 rows of the file. You can modify this in the Configure menu.



Ignore	Ignore	Ignore	Ignore	Ignore	Ignore	Ignore
Array	Station	SEED_station_name	DAS-SN	Sensor-SN	Component	
4	4001	4001	965E	T3544	1	
4	4001	4001	965E	T3544	2	

The 'receiver' header on the GUI window indicates that **noven** is interpreting the currently open file as a receiver table.

You can configure several of **noven**'s setting by opening the File > Configure window:

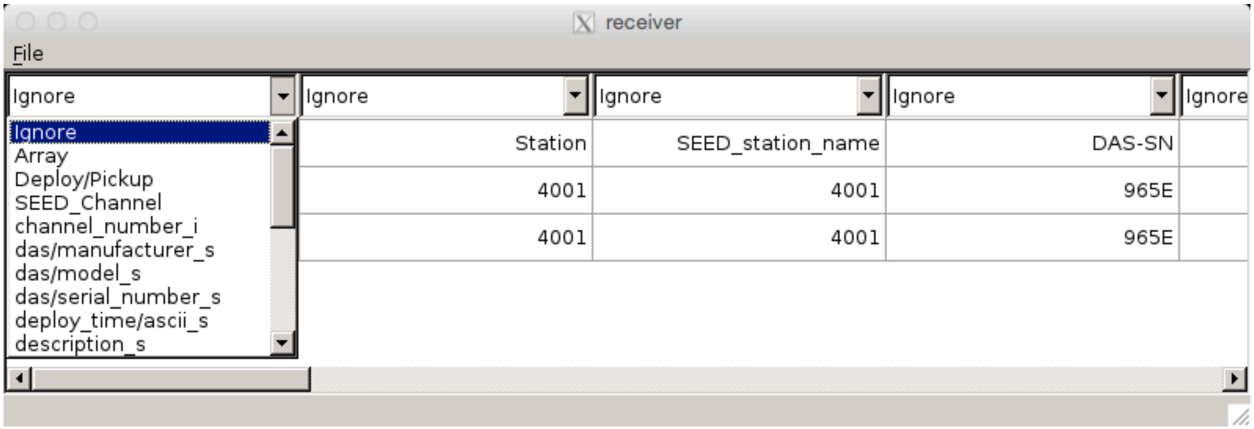


Input Type: receiver
Output Format: kef
Column Separator: comma
Skip Lines: 0
View Lines: 3

Input Type – Specify if the file you are currently working with is for a receiver or an event table.
Output Format – output metadata in a kef file.
Column Separator – Select the appropriate delimitor for your input file. The default is comma-delimited.
Skip Lines – Number of lines at the top of the file to skip. This allows you to load a file with a header field for each column, and then skip that line when checking the input and saving out the kef file.
View Lines – Number of lines to show. **noven** defaults to showing the top 3 lines of a file, but the number of visible lines can be increased if desired.

After setting the appropriate input type in the Configure window (receiver or event), you can specify which columns in your file match the required input columns. While the format of each column is specified, the order of those columns in your file is flexible. When you initially open a file in **noven** the "Ignore" at the top of each column indicates that that column will not be included in the output kef file.

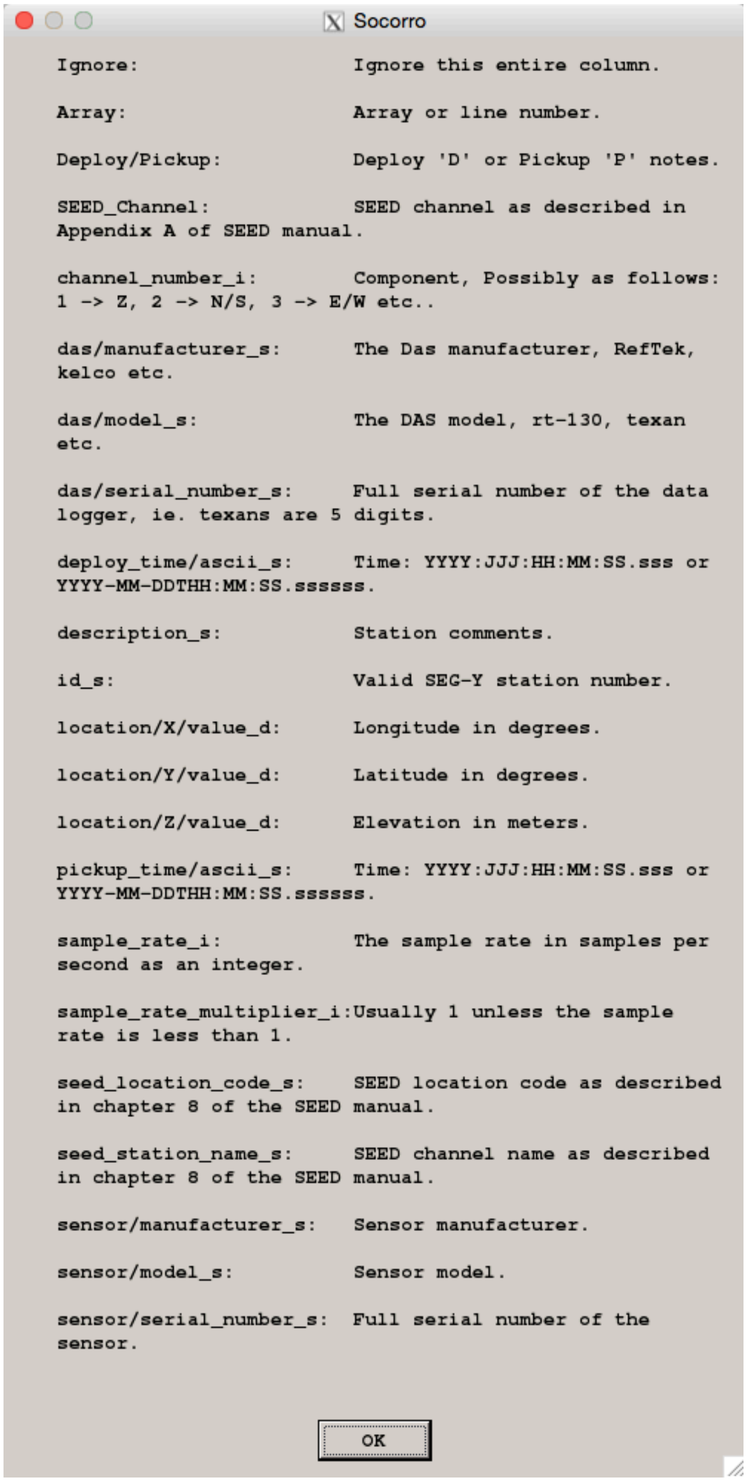
You can select the dropdown menu at the top of each column and change the “Ignore” to the appropriate column identifier.



Station	SEED_station_name	DAS-SN
4001	4001	965E
4001	4001	965E

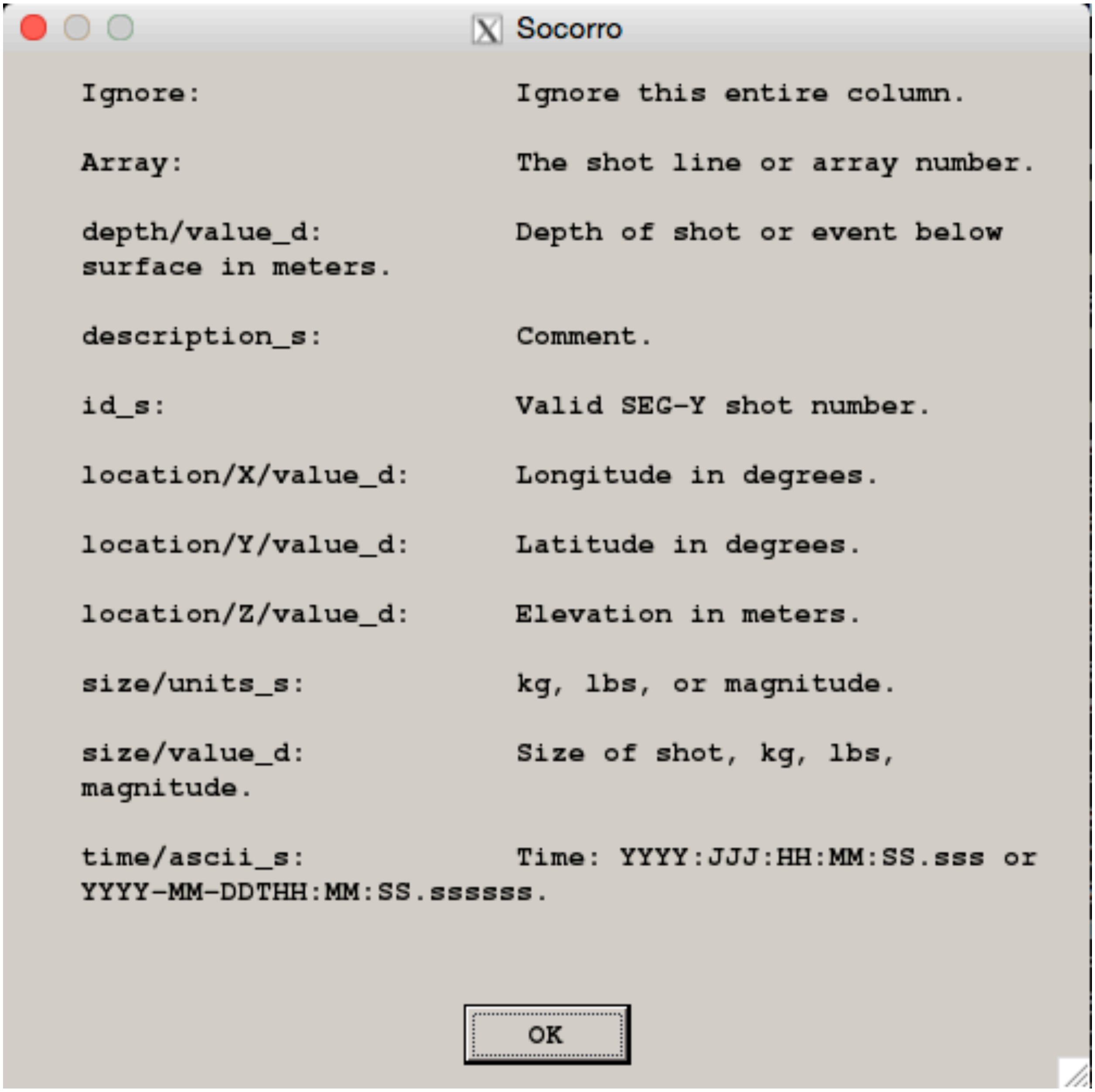
noven’s Help menu has a complete list of the possible columns for the specified file type (receiver or event) and the required format for each column. Select File > Help to view the help for that file type.

For receivers:



```
Ignore:                  Ignore this entire column.

Array:                   Array or line number.

Deploy/Pickup:           Deploy 'D' or Pickup 'P' notes.

SEED_Channel:            SEED channel as described in
Appendix A of SEED manual.

channel_number_i:        Component, Possibly as follows:
1 -> Z, 2 -> N/S, 3 -> E/W etc..

das/manufacturer_s:      The Das manufacturer, RefTek,
kelco etc.

das/model_s:             The DAS model, rt-130, texan
etc.

das/serial_number_s:     Full serial number of the data
logger, ie. texans are 5 digits.

deploy_time/ascii_s:     Time: YYYY:JJJ:HH:MM:SS.sss or
YYYY-MM-DDTHH:MM:SS.ssssss.

description_s:           Station comments.

id_s:                    Valid SEG-Y station number.

location/X/value_d:      Longitude in degrees.

location/Y/value_d:      Latitude in degrees.

location/Z/value_d:      Elevation in meters.

pickup_time/ascii_s:     Time: YYYY:JJJ:HH:MM:SS.sss or
YYYY-MM-DDTHH:MM:SS.ssssss.

sample_rate_i:           The sample rate in samples per
second as an integer.

sample_rate_multiplier_i:Usually 1 unless the sample
rate is less than 1.

seed_location_code_s:    SEED location code as described
in chapter 8 of the SEED manual.

seed_station_name_s:     SEED channel name as described
in chapter 8 of the SEED manual.

sensor/manufacturer_s:   Sensor manufacturer.

sensor/model_s:          Sensor model.

sensor/serial_number_s:  Full serial number of the
sensor.
```

For events:

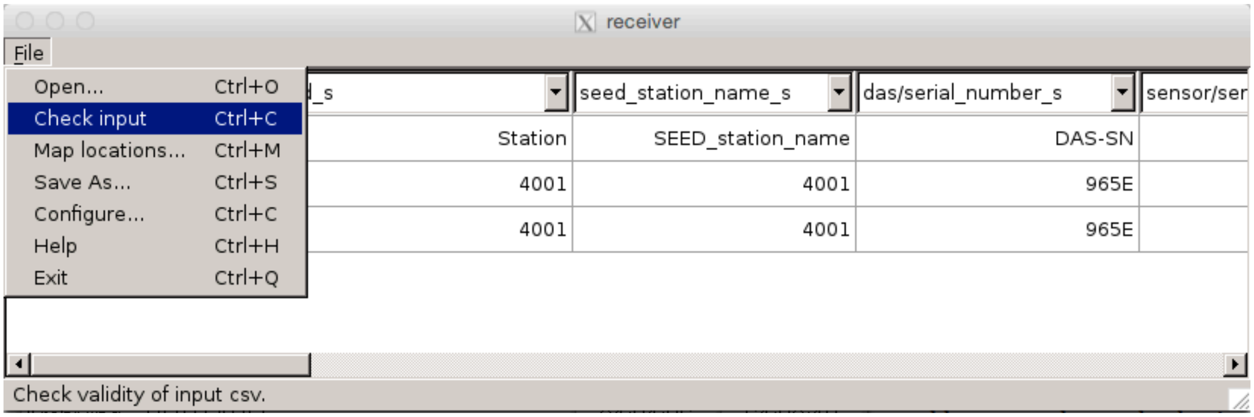


```
Ignore:                  Ignore this entire column.

Array:                   The shot line or array number.

depth/value_d:           Depth of shot or event below
surface in meters.

description_s:           Comment.

id_s:                    Valid SEG-Y shot number.

location/X/value_d:      Longitude in degrees.

location/Y/value_d:      Latitude in degrees.

location/Z/value_d:      Elevation in meters.

size/units_s:            kg, lbs, or magnitude.

size/value_d:            Size of shot, kg, lbs,
magnitude.

time/ascii_s:            Time: YYYY:JJJ:HH:MM:SS.sss or
YYYY-MM-DDTHH:MM:SS.ssssss.
```

If there are additional columns in the file that are not required for PH5 (notes, contact information for landowners, etc), leave those columns specified as “Ignore” and they will not be included in the kef file.

Once you have identified all of the columns in your file using the dropdown menus, if your file has a header line, return to the File > Configure window and change the number of lines to skip so that all header lines are skipped and the only lines displayed are the desired metadata.

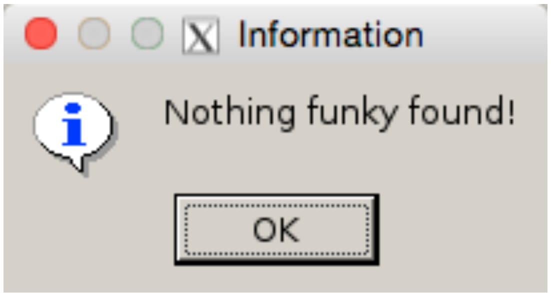
You can then check the file input.



_s	seed_station_name_s	das/serial_number_s	sensor/ser
Station	SEED_station_name	DAS-SN	
4001	4001	965E	
4001	4001	965E	

Check validity of input csv.

If all required columns are defined, and the contents of all columns are formatted correctly, a go ahead message will be displayed.



Nothing funky found!

Once you have received this confirmation that the file is formatted properly, click File > Save As… to save a .kef file of the information. This file should be saved into the METADATA directory.

Once you have checked your .csv and exported it as a .kef file, you can also export the .csv into a .kml file to look at in Google Earth. With **noven** still open, select File > Map Locations, and save the file as example.kml. You can then open this file on Google Earth to ensure that your receivers and sources are in their proper locations. If necessary, correct any errors in your metadata and rebuild the .kef file.

Loading metadata tables

Once your .kef file is created for your receiver and/or shot file, you may then load them into PH5. From the PH5 directory, use the command:

```
>>keftoph5 –n master.ph5 –k ../../METADATA/array_1.kef
>>keftpph5 –n master.ph5 –k ../../METADATA/shot_1.kef
```

In these two examples, the receiver data was named array_1.kef, and the shot metadata was named shot_1.kef. You can load multiple records if you wish to separate out the arrays into multiple .kef files.

If you discover errors in the metadata after loading the source or receiver kef files, you can remove the source or receiver metadata from the PH5 by using the **delete_table** command. See "Deleting Tables" for instructions on removing tables fro the PH5 archive.

Adding response information

PH5 requires you to specify the response information for your instrumentation. There are two steps to this process: updating the response table references and merging the response information. You can skip the Update the response table references section if you only have one type of instrumentation.

Updating the response table references for multiple instrument types

Skip this step if you only have one type of instrumentation. In your PH5 directory containing the sub families and Sigma directory run:

```
>> set_n_i_response
```

This will create a new directory called RESPONSE_T_N_I. Move into this directory.

From the RESPONSE_T_N_I directory run:

```
>> load_das_t --path=../Sigma --onlysave
 >> loadd_das_t --path=../Sigma --onlyload
```

Delete the old response table and load the new Response_t_cor.kef file found in the RESPONSE_T_N_I directory.

```
>> delete_table –n ../Sigma/master.ph5 –R
>> keftoph5 –n ../Sigma/master.ph5 –k Response_t_cor.kef
```

Merging responses

Response files for individual dataloggers and sensors can be downloaded from the IRIS DMC Nominal Response Library (NRL): http://ds.iris.edu/NRL/

Create a RESPONSE folder in your main project directory:

```
>> mkdir RESPONSE
```

From the NRL , save the RESP files for all of your sensors and dataloggers in the response directory.

resp_load is run in the Sigma directory where your combined PH5 archive is located.

First run:

```
resp_load -n master.ph5  -a <comma separated list of arrays you want to load resps for>
```

To generate a template csv file for array 1:

```
>> resp_load -n master.ph5 -a 1
```

This will generate a csv file that contains every combination of das and sensor used in the specified arrays and save it as input.csv

Example of file contents:

```
rt130,cmg3t,100,1,1,,
```

Edit this template file with the path to the RESP files for the DAS and sensor.

Example:
rt130,cmg3t,100,1,1,/path/to/RESP/RT130_100SPS.RESP,/path/to/RESP/CMG3T.RESP

Next you run:

```
>> resp_load -n master.ph5 -a <comma separated list of arrays> -i input.csv
```

This will generate a new response.kef and new array_t_XXX.kef files for each array you specified and replace the existing tables with the updated response and array tables.

Creating and loading derived tables

geo_kef_gen
geo_kef_gen calculates the source-to-receiver offset for each station and shot, and then loads the resulting tables back into ph5 with the following commands. If you do not have any events/sources, you do not need to run **geo_kef_gen**.

```
>> geod_kef_gen –n master.ph5 > offset_t.kef
```

followed by

```
>> keftoph5 –n master.ph5 –k offset_t.kef
```

time_kef_gen
You only need to perform this step if you have Texan data. If you are using RT130s and/or nodes, skip to sort_kef_gen.
Texan data may need timing-drift corrections, which can be easily done with the command

```
>> time_kef_gen –n master.ph5 > time_t.kef
```

followed by

```
>> keftoph5 –n master.ph5 –k time_t.kef
```

sort_kef_gen
The **sort_kef_gen** command produces a .kef containing the information that will help optimize data searches, which is done with the following commands

```
>> sort_kef_gen –n master.ph5 –a >sort_t.kef
```

followed by

```
>> keftoph5 –n master.ph5 –k sort_t.kef
```

Deleting Tables

If you have loaded any metadata that is later found to be incorrect, you can delete the associated tables containing the metadata or tables with calculations based upon the metadata with **delete_table**. **delete_table** will create a backup of the table you are deleting before removing it from the PH5. After rebuilding the .csv into an appropriate .kef file, you can use the following commands to remove any old metadata. For example, if a receiver location is in error after loading the array_1.kef into the PH5 file:

Delete affected receiver tables and any related tables:

```
>> delete_table –n master.ph5 –A 1
```

If you have already created and loaded the source-to-receiver offset table (created using **geo_kef_gen**), you will also need to remove that table when updating the array tables:

```
>> delete_table –n master.ph5 –O
```

This removes the source-receiver offset information. If all receivers/arrays are contained in one csv, you must remove all array tables.

Viewing and confirming your archive

PH5 has several tools to confirm the accuracy of your archive and view your metadata and data before extracting your data in the desired format.

Validate PH5 archive

ph5validate will output a text file of any warnings about your archive and can automatically fix some errors.

```
>> ph5_validate –n master.ph5
```

Check the PH5Validate.log file for any errors. Correct those errors before running **ph5_validate** again.

Checking metadata by outputting stationXML

To look at the stationXML of your ph5 archive, you can use the **ph5tostationxml** command. ph5tostationxml generates stationXML format metadata at the station or response level. To generate stationXML run:

```
>> ph5tostationxml  –n master.ph5  –p /path/to/your/PH5  –o exp-sta.xml  --level=station
>> ph5tostationxml  –n master.ph5  –p /path/to/your/PH5  –o exp-resp.xml  --level=response
```

ph5tostationxml will also generate a Google Earth KML file for all arrays and shot lines:

```
>> ph5tostationxml  –n master.ph5  –p /path/to/PH5  –o exp-geo.kml  –f KML
```

View waveform data

You can view shot and receiver gathers of your data directly from the PH5 using **ph5view**. Start **ph5view** from the command line. You can select your master.ph5 from the File > Open File menu. You can find instructions for using **ph5view** under the Help > Manual menu.

Testing PH5 by outputting data

Once your PH5 database has been completely built and validated, you can test the quality of the data by outputting it in various formats. In the following steps, you can either output to SEG-Y gathers with the **ph5toevt** command, or into miniseed or SAC using the **ph5toms** command.

Build and view SEG-Y gathers

PH5 can create event or receiver gathers via command line. Here is a gather-building example:

```
>> ph5toevt  -n master.ph5  --use_deploy_pickup  -o Gathers  -N  -l 10  –A 1  -x U -e 5012 --shot_line 1
```

The SEG-Y gather in this example is written to the Gathers directory and contains 10 second long un-time-corrected traces from array 1 receivers (all components) starting at the time of event 5012. A log of **ph5toevt** activity is also saved in the specified output directory. If you are ready to produce gathers for all events, you can run **ph5toevt** with the '-E' flag to create gathers for all of the events. Run **ph5toevt** –h for more options.

Output SAC or miniSEED data

PH5 also supports writing data out as SAC or miniSEED. You can write out all of the data from the PH5 archive or just specified time windows using the **ph5toms** command. **ph5toms** defaults to miniSEED for the output format unless the –F flag is specified. Try **ph5toms –h** for all options.

To output miniSEED for two days of data for array 1, run the following:

```
>> ph5toms -n master.ph5 -a 001 -s 2017:052:00:00:00 -t 2017:053:23:59:59 -o mseed -F MSEED
```

To output SAC files for all data contained in the PH5 archive, run the following:

```
>> ph5toms –n master.ph5 –F SAC –o sac_out/
```

Appendix

Updating the PH5 software

The PH5 software can be updated by pulling the most recent version from the PASSCAL PH5 GitHub.

To update the PH5 software, first locate the PH5 directory that was created when you initially pulled the PH5 package from GitHub.

From the PH5 directory, run:

```
>> git pull
```

This will go to the PH5 GitHub and pull the most recent version.

Now, activate PH5:

```
>> source activate ph5
```

Once PH5 is active, you can uninstall the current version and install the new version:

```
>> pip uninstall ph5
```

Select y when prompted. You should see a message saying that you "Successfully uninstalled ph5-????".

Now install the new version:

```
>> python setup.py install
```

Your PH5 software installation should now be up-to-date.

Loading all data using the command line

Using the command line to load data allows for precise loading of data into PH5, and is recommended only in specific situations. For the majority of experiments, loading data using **pforma** is recommended.

In the PH5 directory, initialize the PH5 file. Name it "master":

```
>> initialize_ph5 –n master.ph5
```

Add Texan raw data to the PH5 file with **125atoph5**. The "trd_list" should list the absolute or relative path and file names as shown to the right. Use the "M" option, specifying 1/6th the data volume to be loaded, in integer GB (you can use **du -sh** ../RAW to estimate data volume). The "M" option causes the data to be uploaded into a number of "data-only" ph5 files (called 'mini' files) linked to the master ph5 file, optimizing subsequent data processing and extraction.

Here we load ~100 GB of data:

```
>> 125atoph5 –f trd_list –n master.ph5 –M 17 >& 125a2ph5.out
```

Node data can also be loaded via command line using the **segdtoph5** command similar to the **125atoph5** example above.

Add RT130 data with **130toph5**, setting the "M" option to 1/2th the RT130 data volume to be loaded, in integer GB. A list of RT130 raw files may be referenced as input to **130toph5** similar to the **125atoph5** example above.

If Receivers do not Align True North or East

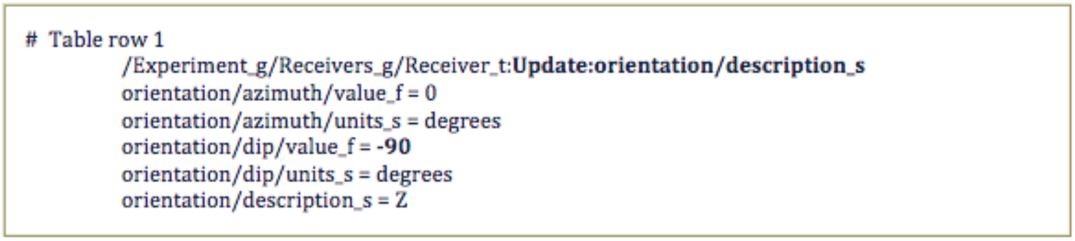
This step should only be followed in situation where a sensor's horizontal components are not oriented with N as True North and E as 90 East of True North, or where a sensor's vertical component provides positive voltage output with upward ground motion. Skip this step for standard installations of nodes, GS-11, L-4C, L-22, and L-28 sensors.

For the aforementioned situations:

a. Dump the receiver table to a .kef: **ph5tokef** –n master.ph5 –C > receiver_t.kef
b. Delete all lines from the .kef corresponding to rows in the table not requiring a change to the azimuth or dip, otherwise these rows will be duplicated in the table upon upload of the modified .kef.

c. Add the string ':Update:description_s' to the row description of the remaining rows (Remember the rows in need of revision should be the only ones left in the .kef).

In the example below we removed the lines corresponding to the N and E rows of the table, added ':Update:orientation/description_s' and changed the dip value from '90' to '-90'



```
# Table row 1
          /Experiment_g/Receivers_g/Receiver_t:Update:orientation/description_s
          orientation/azimuth/value_f = 0
          orientation/azimuth/units_s = degrees
          orientation/dip/value_f = -90
          orientation/dip/units_s = degrees
          orientation/description_s = Z
```

d. Load the receiver_t.kef into the PH5 file with
   >> **keftoph5** –n master.ph5 –k receiver_t.kef –c >& receiver_kef2ph5.out
   This command uses the '-c' option as a check of .kef syntax (necessary for hand-edited files).
e. Inspect the output file for errors. You can try egrep –i "err|warn|prob|fail|fatal" receiver_kef2ph5.out to search for the strings in the output file. Correct any issues and re-check.
f. Repeat the kef2ph5 command without the '-c' option for the changes to be written into the PH5 file.

Additional metadata guidelines

General guidelines

- Each array can only include one sample rate. If you recorded more than one sample rate, you must separate the different sample rates into separate arrays.
- Station ID (id_s) needs to be a valid SEG-Y station number, ie numeric only.
- For MSEED channel names, use SEED convention for your instrumentation and sample rate according to Appendix A of the SEED Manual. There is a lookup table for most of the PASSCAL instrument pool in this document.
- SEED station name is required; SEED station names are required to be 3 to 5 characters, alphanumeric, all capital letters. If your station IDs (id_s) fits these requirements, you can just repeat those values in this column.
- Separate files for node metadata are not required unless something in the original experiment setup needs to be changed (ie, changing the station names/numbers, etc).
- Once you have finished editing your receiver table, save the file out as a comma-separated variable file (.csv).

Instrument specific guidelines
Fairfield Nodal 1C nodes

- In most cases, PH5 takes node metadata from the SEG-D headers. Only create and load new node metadata if the SEG-D headers are incorrect.
- For 1C nodes, channel number 1 is Z.
- DAS-SN is made up of a combination of the Array/Line and Station, ie Line -> 321, Station -> 654, DAS-SN -> 321X654.
- For Sensor-SN, use the node serial number.

Example 1C node csv:

```
Array,Station(id_s),SEED_station_name,DAS-SN(LineXSta),Sensor-
SN,Channel_num(Z=1),SampleRate,SEED_Channel,Lat(deg),Lon(deg),Elev(m),Deploy_time,Pickup_t
ime,DASManufacturer,DASModel,SensorManufacturer,SensorModel
1,1001,1001,1X1001,12345,1,250,DPZ,36.586,-97.6565,
320,2016:173:00:00:00,2016:304:00:00:00,FairfieldNodal,ZLAND_1C,Geospace,GS30CT
1,1002,1002,1X1002,12346,1,250,DPZ,36.586,-97.6565,
320,2016:173:00:00:00,2016:304:00:00:00,FairfieldNodal,ZLAND_1C,Geospace,GS30CT
1,1003,1003,1X1003,12347,1,250,DPZ,36.586,-97.6565,
320,2016:173:00:00:00,2016:304:00:00:00,FairfieldNodal,ZLAND_1C,Geospace,GS30CT
```

Fairfield Nodal 3C nodes

- In most cases, PH5 takes node metadata from the SEG-D headers. Only create and load new node metadata if the SEG-D headers are incorrect.
- For 3C nodes, channel number 1 is N/S, 2 is E/W, and 3 is Z.
- DAS-SN is made up of a combination of the Array/Line and Station, ie Line -> 321, Station -> 654, DAS-SN -> 321X654.
- For Sensor-SN, use the node serial number.

Reftek RT125 (Texans)

- For Texans, channel number is 1 (vertical) for 1C geophones.
- For DAS/Texan serial numbers, add 10,000 to the serial number written on the Texan to get a 5-digit number.
- For sensor serial numbers, create unique dummy serial numbers for geophones if unique serial numbers do not exist.

Reftek RT130

- For RT130s, channel number 1 is Z, 2 is N/S, and 3 is E/W.
- Each recorded data stream/unique sample rate must be in its own array, ie 100sps in Array 1, 1sps in Array 2, and 0.1sps (ex. mass positions) in Array 3.
- For sample rates less than 1 (ie 0.1sps) the sample rate multiplier field (SR-mult) should be set appropriately. For 0.1 sps, SR-mult should be 10; for 0.01 sps, SR-mult should be 100.

Example RT130 csv:

```
Array,Station(id_s),SEED_station_name,DAS-SN,Sensor-SN,Channel_num(Z=1 N=2
E=3),SampleRate,SRmult,SEED_Channel,Lat(deg),Lon(deg),Elev(m),Deploy_time,Pickup_time,DAS
Manufacturer,DASModel,SensorManufacturer,SensorModel
1,1001,1001,A123,T1234,1,100,1,HHZ,36.586,-97.6565,
320,2016:173:00:00:00,2016:304:00:00:00,Reftek,rt130,Guralp,cmg3t
1,1001,1001,A123,T1234,2,100,1,HH1,36.586,-97.6565,
320,2016:173:00:00:00,2016:304:00:00:00,Reftek,rt130,Guralp,cmg3t
1,1001,1001,A123,T1234,3,100,1,HH2,36.586,-97.6565,
320,2016:173:00:00:00,2016:304:00:00:00,Reftek,rt130,Guralp,cmg3t
2,2001,2001,A123,T1234,1,1,1,LHZ,36.586,-97.6565,
320,2016:173:00:00:00,2016:304:00:00:00,Reftek,rt130,Guralp,cmg3t
2,2001,2001,A123,T1234,2,1,1,LH1,36.586,-97.6565,
320,2016:173:00:00:00,2016:304:00:00:00,Reftek,rt130,Guralp,cmg3t
2,2001,2001,A123,T1234,3,1,1,LH2,36.586,-97.6565,
320,2016:173:00:00:00,2016:304:00:00:00,Reftek,rt130,Guralp,cmg3t
```

Lookup table for equipment types/SEED channel names

Instrument	**Sample Rates (Hz)**							
	1000 ≤ SPS < 5000	250 ≤ SPS < 1000	80 ≤ SPS < 250	10 ≤ SPS < 80	1 < SPS < 10	1	0.1	0.01
STS-2	FH?	CH?	HH?	BH?	MH?	LH?	VH?	UH?
CMG-3T	FH?	CH?	HH?	BH?	MH?	LH?	VH?	UH?
CMG3ESP	FH?	CH?	HH?	BH?	MH?	LH?	VH?	UH?
TR-240	FH?	CH?	HH?	BH?	MH?	LH?	VH?	UH?
TR-120	FH?	CH?	HH?	BH?	MH?	LH?	VH?	UH?
TR-40	FH?	CH?	HH?	BH?	MH?	LH?	VH?	UH?
CMG-40T 30s	FH?	CH?	HH?	BH?	MH?	LH?	VH?	UH?
CMG-40T 1s	GH?	DH?	EH?	SH?	MH?	LH?	VH?	UH?
S-13	GH?	DH?	EH?	SH?	MH?	LH?	VH?	UH?
HS-10	GH?	DH?	EH?	SH?	MH?	LH?	VH?	UH?
L-4C*	GL?	DL?	EL?	SL?	ML?	LL?	VL?	UL?
L-22*	GL?	DL?	EL?	SL?	ML?	LL?	VL?	UL?
L-28	GL?	DL?	EL?	SL?	ML?	LL?	VL?	UL?
L-28LB (4.5Hz geophone)*	GL?	DL?	EL?	SL?	ML?	LL?	VL?	UL?
L-40A (40 Hz geophone)	GP?	DP?	EP?	SP?	MP?	LP?	VP?	UP?
Fairfield Node	GP?	DP?	EP?	SP?	MP?	LP?	VP?	UP?
FBA ES-T	FN?	CN?	HN?	BN?	MN?	LN?	VN?	UN?

Table 1. Recommended SEED channel names for many of the sensors available from PASSCAL

* The use of “H”, denoting high gain, assumes that the dataloggers are programmed using a gain of 32, which is the recommended gain setting for a typical PASSCAL experiment. In the event the gain is set to 1 at the datalogger, then the second character of the channel name should be set to “L”.