Introduction to Active Source Data Archiving Utilizing PH5 as the Archive Format

Data Group
data_group@passcal.nmt.edu
IRIS/PASSCAL Instrument Center
Socorro, NM
Version: 2012336

Table of Contents

1 Conventions used in this Document

Below are conventions utilized throughout this document:

Bold-only indicates something of importance or a new section heading.

Italicized-only indicates the name of software within the body of the document so as to allow the software to stand-out in the text. Software utilized in examples or usage statements will not be italicized.

Bold italicized text indicate user input either during software operation or within a file.

2 PH5 Archive Format: PASSCAL's Implementation of the HDF5 Model

2.1 HDF5/HDF5 Overview

HDF5 (Hierarchical Data Format) is an open-source data model with more than a 20 year development history. It is capable of storing large, complex data sets and is utilized in a variety of science and engineering fields. Data and meta-data are organized in several groups, which are further subdivided into subgroups, arrays and tables. The principle strength of HDF5 as a data archiving model for active source data sets is the separation of the meta-data from the time series. This allows the meta-data to be entered, stored, and manipulated independent of the data. Unlike SEG-Y data sets, with HDF5 one may modify the meta-data, such as shot times for example, without reprocessing and re-archiving the data. This strength is analogous to a principle advantage of the SEED data format in that the meta-data, stored as a dataless SEED volume, may be modified after data are archived without the need to re-archive data.

We refer to the PASSCAL implementation of HDF5 as a PH5 file. PASSCAL-supported software exists to populate, view and modify PH5 file contents, and to convert a PH5 file to SEG-Y format. The PH5-formatted data are stored at the IRIS DMC and, at the time of a user request, data are cut, with the start-time and duration set by the user (within the limits of SEG-Y), and written in SEG-Y gathers or PASSCAL SEG-Y trace files.

2.2 File Organization

The data and meta-data are organized in several groups, tables, and arrays within the PH5 file. The Experiment Group is the primary group. Within the Experiment Group the data and meta-data are further organized into four main groups:

- Receivers
- Reports,
- Responses
- Sorts

In addition to these groups, the Experiment Group also contains the Experiment Table which

is comprised of general information, such as a summary of the experiment, the general geographic location, and principal investigator names and institutions (Figure 1).

Various naming conventions are used within a PH5 file and in the archiving process:

- groups – all groups end with the suffix "_g" (e.g. Sorts_g),
- tables – all tables end with the suffix "_t" (e.g. Experiment_t),
- arrays – all arrays end with the suffix "_a" (e.g. SOH_a).

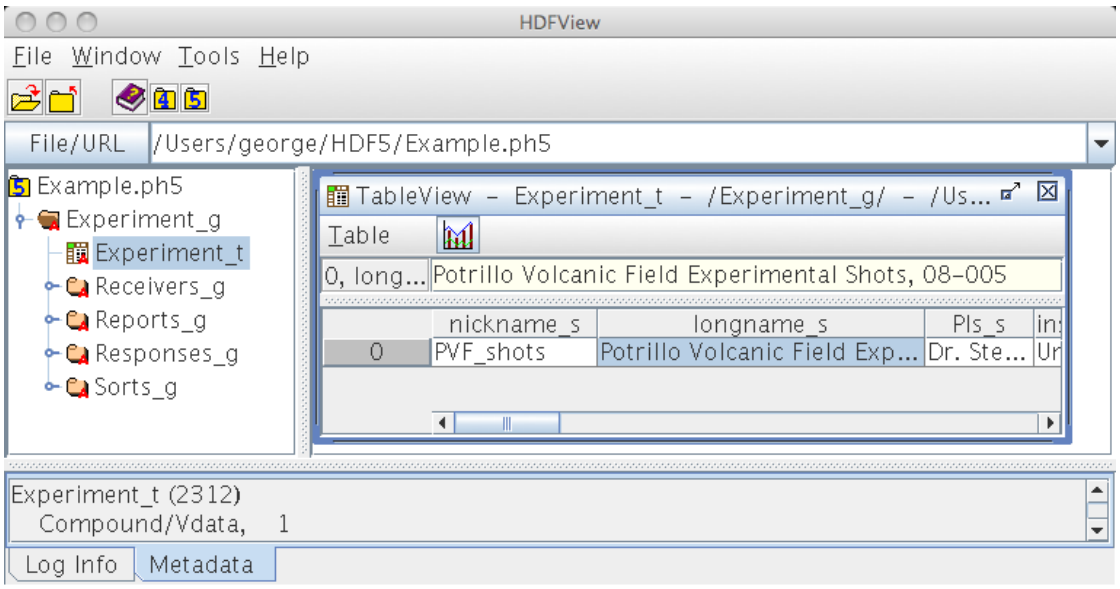

Figure 1. A snapshot of a PH5 file when viewed with the HDF community-supported GUI, *hdfview*. The hierarchical data structure of PH5 is reinforced with the outline form of organization on the left side of the window. The Experiment Table (Experiment_t) has been opened and is shown on the right side of the viewer. The Experiment Group contains all of the data and meta-data. Beneath Experiment Group the data and meta-data are organized further into the Receivers, Reports, Responses, and Sorts Groups and the Experiment Table. The GUI *hdfview* is discussed later in this document.

The Receiver Group (Figure 2a) serves to organize the data and data-logger-specific information. Within the Receiver Group each data logger has its own Das Group where its data, state-of-health (SOH) and recording window information are stored. The Time and Receiver Tables within the Receivers Group store internal clock drift correction details (if applicable) and sensor orientation information, respectively.

The Reports Group (Figure 2b) includes information such as your experiment report, associated documents and information utilized by the IRIS DMC to generate web forms for the data set.

The Response Group (Figure 2b) is populated with data logger gains and bit weights and, when applicable, response files as generated by *rdseed* (for more information about *rdseed* see www.iris.edu/manuals/rdseed.htm).

Finally, the Sorts Group (Figure 2c) includes a number of tables including the Array, Event,

Offset, and Sorts Tables. The Event Table contains source details, such as size, time, and location while the Array Table(s) store receiver information including latitude, longitude and elevation based on archiver-defined station arrays. The Offset Table stores source-to-receiver offsets in terms of azimuth and distance from source to receiver. Finally, the contents of the Sorts Table relate the archiver-defined station arrays and recording windows to optimize the data retrieval of during the construction of SEG-Y gathers or trace files.

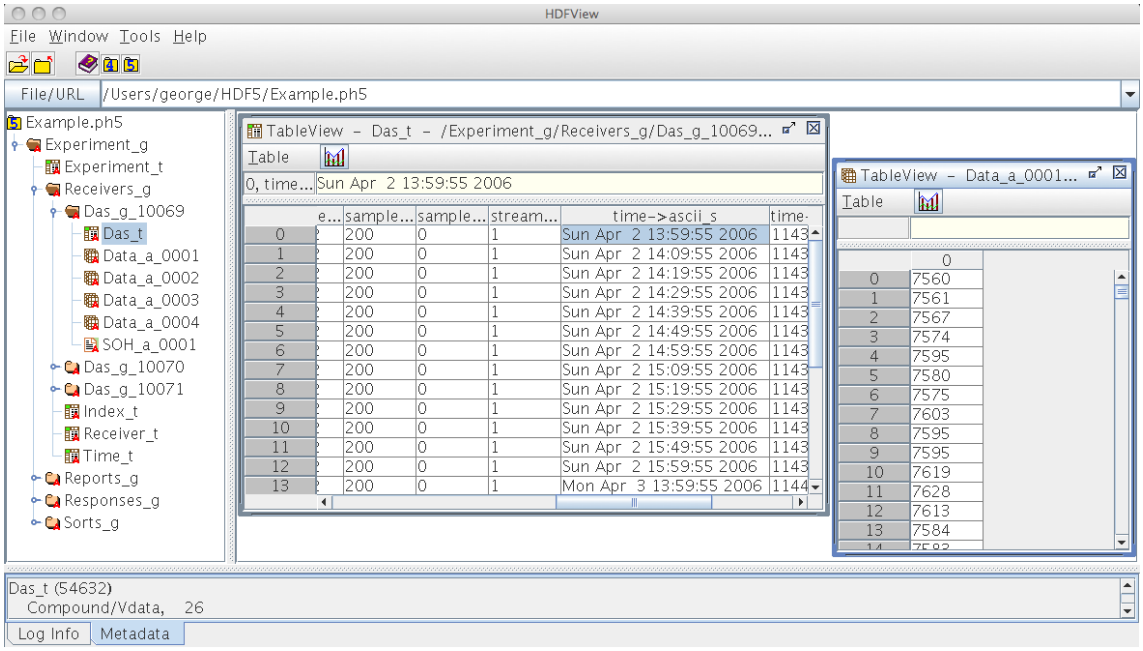

Figure 2a. Zooming into the Das Group for data logger serial number 10069. Within the Receiver Group, a Das Group, for each data logger, contains the data and state-of-health (SOH) information for a given data logger in the form of arrays. Selected, in the center, is the Das Table (Das_t), highlighting an recording window start time, and, to the right, a Data Array (Data_a_0001) listing the data point values corresponding to the highlighted recording window start time. Also found in the Receiver Group are the Receiver Table (Receiver_t), containing sensor orientation information, and the Time Table (Time_t), storing internal oscillator drift correction factors. The Index Table is not presently used.

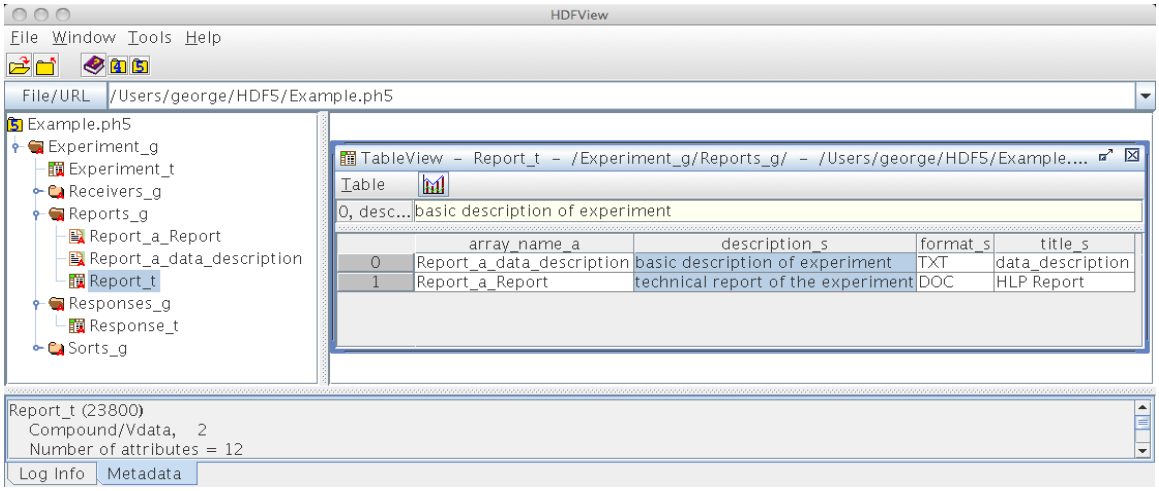

Figure 2b. An overview of the Reports and Responses Group. Within the Reports Group a variety of information may be stored: reports, papers, maps and plots in a variety of formats doc, png, pdf - to name a few. Here we show the Report Table (report_t) which catalogs the contents of the Reports Group. The Response Group in this example is not populated with responses. However, if responses were stored, the Responses Table would catalog the responses stored within this group.

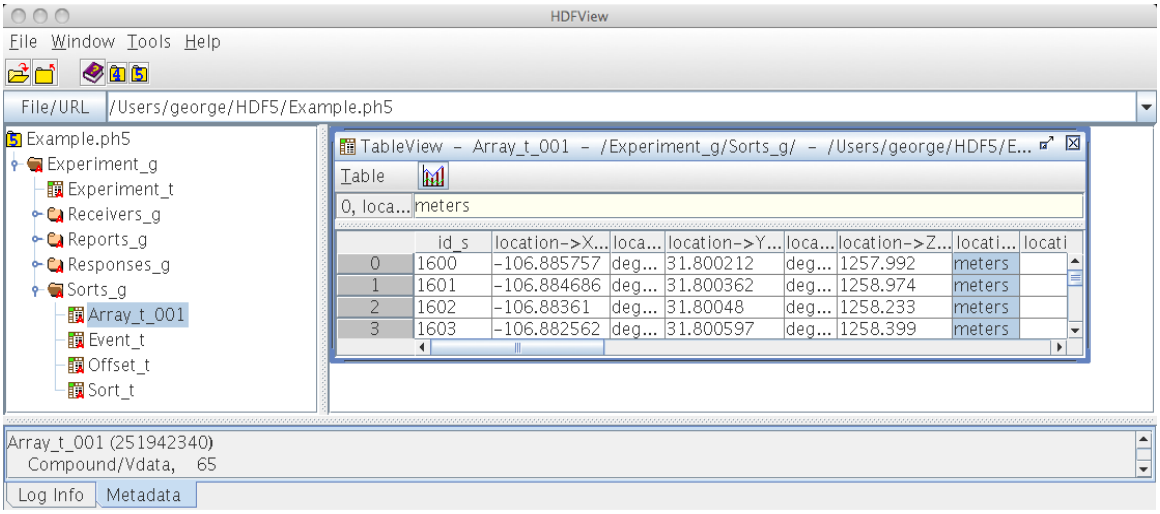

Figure 2c. A view of the Sorts Group expanded with the array table opened. The array table within the Sorts Group stores station location information based on archiver-defined receiver arrays. Array_t_001 is the only receiver array described for this experiment; multiple arrays may be used to organize you experiment's geometry. In the figure the array window (Array_t_001) has been opened and shows the first four stations and their latitude, longitude and elevation. Also stored within the Sorts Group are tables containing: source details (in the Event_t table), source to receiver offsets (in the Offset_t table), and recording window information (in the Sorts_t table).

3 Getting Started

3.1 Assembled data set ID

Active source data sets are normally archived as assembled data sets. An ID number serves as a reference for data requests and submissions. Please request an assembled data set ID number via the PI Home Page on the IRIS/PASSCAL website at:

www.passcal.nmt.edu/pihomepage

Note an automated email will be sent by IRIS DMC staff suggesting the archiver should send the data directly to the DMC. Please disregard this point, as all PASSCAL-supported experiments should have their data sets sent to PASSCAL for review and archiving.

3.2 Processing Hardware and Software

Hardware and software needs: a computer with a Linux OS with the appropriate PH5 processing tools will satisfy these requirements. All of the necessary software is available on the “field” user account of a PASSCAL-supplied computer.

Data requirements: at this point the data should be off-loaded from your data loggers on to a common hard-drive. If the data are not stored on the internal drive of a PASSCAL-supplied field laptop, ensure the data are written to a hard-drive utilizing a file system compatible with Linux (ext3). At a minimum you will need slightly more than double the raw data set size

available on the drive to store the raw data, the PH5 file and other related files. Organize your data and meta-data in a logical manner, such that paths and file names are easily found and referenced during the archiving process.

A checklist of meta-data required for active source experiments processed as PH5 data sets is available in **Appendix A,** Meta-Data Checklist - PH5 Data Sets**.** This meta-data should be available in deployment file, or dep file. See **Appendix B** Dep File Construction regarding the construction of a dep file. PASSCAL recommends the use of spreadsheets to initially store meta-data, such as the PH5-template spreadsheet, which is introduced in **Appendix B.**

Finally you will need a short report, in pdf, Word, or text format, describing the following:

purpose and general overview of the experiment,
sensors used (response information if applicable),
source type, size, depth (vibroseis specifics such as duration, frequency, wave type, if applicable), and
principle investigator names, institutions and contact information.

Recall various naming conventions are used within a PH5 file and in the archiving process:

groups – all groups end with the suffix "_g" (e.g. Sorts_g),
tables – all tables end with the suffix "_t" (e.g. Time_t),
arrays – all arrays end with the suffix "_a" (e.g. SOH_a).

You will find it helpful to use the same naming conventions outlined here (as we do in this document) for various input and output files during the archiving process.

3.3 Working with the PH5 Tools

The PH5 software tools are maintained by PASSCAL and are installed on PASSCAL field computers. All software tools described in this document have usage statements available through the "-h" command line option.

4 Processing your Data Set

4.1 The kef File: Providing a Means to add or edit Meta-Data

We input and modify the meta-data contents of PH5 files by way of formatted text files called **kitchen exchange format** files, or **kef** files. A kef file is a text file with specific formatting serving to represent the rows and columns in the tables described in a PH5 file. For example, the experiment_t.kef file contains summary experiment information. See **Appendix C**, Example Experiment_t.kef File, to view the contents. An example experiment.kef file is available on your PASSCAL computer from which you may construct your own.

4.2 Initializing a PH5 File

Most software tools developed for PH5 operate on existing PH5 files. Therefore, we need to build an empty PH5 file, which will provide the framework into which we load the data and meta-data. The program *initialize-ph5* provides this mechanism. While *initialize-ph5* will load the experiment.kef file it best to check its contents and load it by the same means as other kef files, which is discussed in a subsequent section.

Example: initialize-ph5 -n 08-021

initialize-ph5 help:

```
Usage: Version: 2011.187.2 initialize_ph5 [--help]--kef kef_file --output output_file

Program to initialize PH5 file at start of experiment. The kef file should
contain information for experiment table /Experiment_g/Experiment_t.

Options:
  -h, --help            show this help message and exit
  -n output_file, --nickname=output_file
                        Experiment nickname.
  -k kef_file, --kef=kef_file
                        Kitchen Exchange Format file containing experiment
                        info.
```

4.3 Loading the Data

With a PH5 file in hand we are ready to add the raw data to the PH5 file. Texan data are loaded with *125a2ph5;* for RT-130 data use *1302ph5.* In either case the raw data are most easily loaded by defining and referring to a list of data files. List the full path to the files or the path relative to the current working directory. Consider writing the standard output and standard error to a file (as shown in the example below) so you have a record of the output and errors, if any, which may occur.

After the data have been loaded, space permitting, make a backup copy of the ph5 file prior to loading the meta-data.

Below are the first few lines of an example raw file list to be used as input to *125a2ph5*:

```
../../HLP/Keller/RAW/I0025RAW.TRD
../../HLP/Keller/RAW/I0026RAW.TRD
../../HLP/Keller/RAW/I0027RAW.TRD
```

The first example shows how to load Texan data; the subsequent example shows how to load RT-130 data.

Example: 125a2ph5 -f trd_file_list -n 08-021 >& 125a2ph5_08021.out

125a2ph5 help:

```
Usage: Version 2010.315 125a2ph5 [--help][--dep dep_file][--kef kef_file][--raw raw_file
| --file file_list_file] --nickname output_file_prefix

Read a raw texan files and optionally a kef file into ph5 format.

Options:
  -h, --help            show this help message and exit
  -r raw_file, --raw=raw_file
                        RT-125(a) texan raw file
  -f file_list_file, --file=file_list_file
                        File containing list of RT-125(a) raw file names.
  -n output_file_prefix, --nickname=output_file_prefix
                        The ph5 file prefix (experiment nick name).
  -k kef_file, --kef=kef_file
                        Kitchen Exchange Format file.
  -d dep_file, --dep=dep_file
                        Rawmeet dep file.
  -s samplerate, --samplerate=samplerate
                        Extract only data at given sample rate.
  -p
```

Here's how to load RT-130 data.

Example: 1302ph5 -n08-021 -f rt130_file_list >& 1302ph5_08021.out

1302ph5 help:

```
Usage: Version: 2010.270 initialize_ph5 [--help]--kef kef_file --output output_file

Program to initialize PH5 file at start of experiment. The kef file should
contain information for experiment table /Experiment_g/Experiment_t.

Options:
  -h, --help            show this help message and exit
  -n output_file, --nickname=output_file
                        Experiment nickname.
  -k kef_file, --kef=kef_file
                        Kitchen Exchange Format file containing experiment info.
yellowstone:HDF5 george$ 1302ph5 -h
Psyco not available. Will run slowly.
Usage: 1302ph5 [--help][--dep dep_file][--kef kef_file][--raw raw_file | --file file_list_file]
```

```
--nickname output_file_prefix

Read a raw rt-130 files and optionally a kef and/or dep file into ph5 format.
v2011.005 Developers version

Options:
  -h, --help            show this help message and exit
  -r raw_file, --raw=raw_file
                        RT-130 raw file
  -f file_list_file, --file=file_list_file
                        File containing list of RT-130 raw file names.
  -n output_file_prefix, --nickname=output_file_prefix
                        The ph5 file prefix (experiment nick name).
  -k kef_file, --kef=kef_file
                        Kitchen Exchange Format file.
  -d dep_file, --dep=dep_file
                        Rawmeet dep file.
  -s samplerate, --samplerate=samplerate
                        Extract only data at given sample rate.
  -w windows_file, --windows_file=windows_file
                        File containing list of time windows to process.
                        Window start time    Window lentgh, seconds
                        -----------------    ----
                        YYYY:JJJ:HH:MM:SS    SSSS
  -p par_file, --parfile=par_file
                        [Used to set sample rate and gain in the case of a
                        missing event header.]
                        Parameter file used to set samplerate, gain.
                        The file contains colon separated lists.
                        The first line describes the
                        order and the first char must be '#'.
                        As example the first four lines could be:
                        #das;refchan;refstrm;samplerate;gain
                        9882; 1; 1; 40; x1
                        9882; 2; 1; 40; x1
                        9882; 3; 1; 40; x1
                        9882; 1; 2; 1; x32
                        9882; 2; 2; 1; x32
                        9882; 3; 2; 1; x32
                        Allowed fields:
                        das;station;refchan;refstrm;samplerate;gain
  -P
```

4.4 Construction of a Dep File

A deployment file, or dep file, is a text file containing a conglomeration of source and/or

receiver information (meta-data) in a specific semi-colon-delimited format. Here we describe the information contained within a dep file and how one would create the dep file.

See **Appendix B,** Dep File Construction, regarding the construction of a dep file. PASSCAL recommends the use of spreadsheets to initially store meta-data, such as the PH5-template tab-delimited spreadsheet discussed in **Appendix B**.

The receiver and shot dep files typically contain the items below. Note the list below documents required and recommended fields to include in each dep file (which are included in **Appendix B,** Dep File Construction, procedures).

Shot lines:

- "SHOT" (a descriptor for the line)
- shot ID (where first digit is the the array number)
- flag (same as shot ID)
- line (or array, optional)
- latitude (N or S or "-" for south latitudes, in decimal degrees (WGS84 system), optional)
- longitude (E or W, or "-" for west longitudes, in decimal degrees, optional)
- depth (in meters, optional)
- size (optional, in kilograms or magnitude equivalent; may be added to PH5 file later)

Receiver lines:

- "RECV" (a descriptor for the line)
- DAS serial number: for Texans, add 10000 to number as **all Texan serial numbers must be 5 digits long**
- receiver or flag number (where first digit is the the array number)
- Line (or array)
- DAS type
- channel (1, 2 or 3)
- sensor ID (optional)
- latitude (N or S or "-" for south latitudes, in decimal degrees (WGS84 system))
- longitude (E or W, or "-" for west longitudes, in decimal degrees)
- elevation (meters)
- deployment time*
- pick up time*

* If deployment and pick up times are not known see **Appendix D**, Determining Deployment and Pick up Times from Data Loaded in a PH5 file.

Lines from both the receiver and shot dep files examples are shown below. Notice in these examples:

- two different dataloggers are defined (Texan and Reftek (RT-130))
- 5 digit Texan serial numbers
- each station defined has 3 components (simply omit the lines defining the horizontal components (2 and 3) if only vertical data exist)

- inclusion of receiver deployment and pick up times

Receiver dep file example

```
RECV;10083;2001;2;texan;1;;;N43.872292;W120.879075;1302.896400;;2008:252:18::00;2008:255:08:10:00;;
RECV;10083;2001;2;texan;2;;;N43.872292;W120.879075;1302.896400;;2008:252:18::00;2008:255:08:10:00;;
RECV;10083;2001;2;texan;3;;;N43.872292;W120.879075;1302.896400;;2008:252:18::00;2008:255:08:10:00;;
RECV;9AC6;3135;3;reftek;1;;;N42.791510;W118.801380;1578.900000;;2008:252:18::00;2008:255:08:10:00;;
RECV;9AC6;3135;3;reftek;2;;;N42.791510;W118.801380;1578.900000;;2008:252:18::00;2008:255:08:10:00;;
RECV;9AC6;3135;3;reftek;3;;;N42.791510;W118.801380;1578.900000;;2008:252:18::00;2008:255:08:10:00;;
```

Shot dep file example

```
SHOT;9012;9012;9;N43.635933;W120.186240;1347.000000;2008:253:00:10:00;;;;;;;;
```

4.5 Reviewing the experiment's geometry

Before loading the ph5 file take a moment to plot and review the experiment's geometry, using *kmz-builder* to construct a kmz file (compressed Google Earth kml file and related files) for each dep file. Plot both the receiver and shot dep files. Be sure to rename the output file produced by *kmz-builder* to ensure the previously produced kmz file is not over-written by subsequent runs of *kmz-builder*. Load the resulting kmz file(s) into Google Earth and review your geometry.

Example: kmz-builder -d receivers.dep

kmz-builder help:

```
Usage: kmz-builder.py --nickname PH5-file-prefix || -d dep-file [--path path-to-PH5-files]

Version: 2011.021. Generates a GoogleEarth kmz file of receivers and sources.

Options:
  -h, --help            show this help message and exit
  -n PH5_file_prefix, --nickname=PH5_file_prefix
                        The PH5 file prefix .
  -d dep_file           the dep file to plot
  -p PH5_path, --path=PH5_path
                        Path to PH5 files. Defaults to current directory.
  -b
```

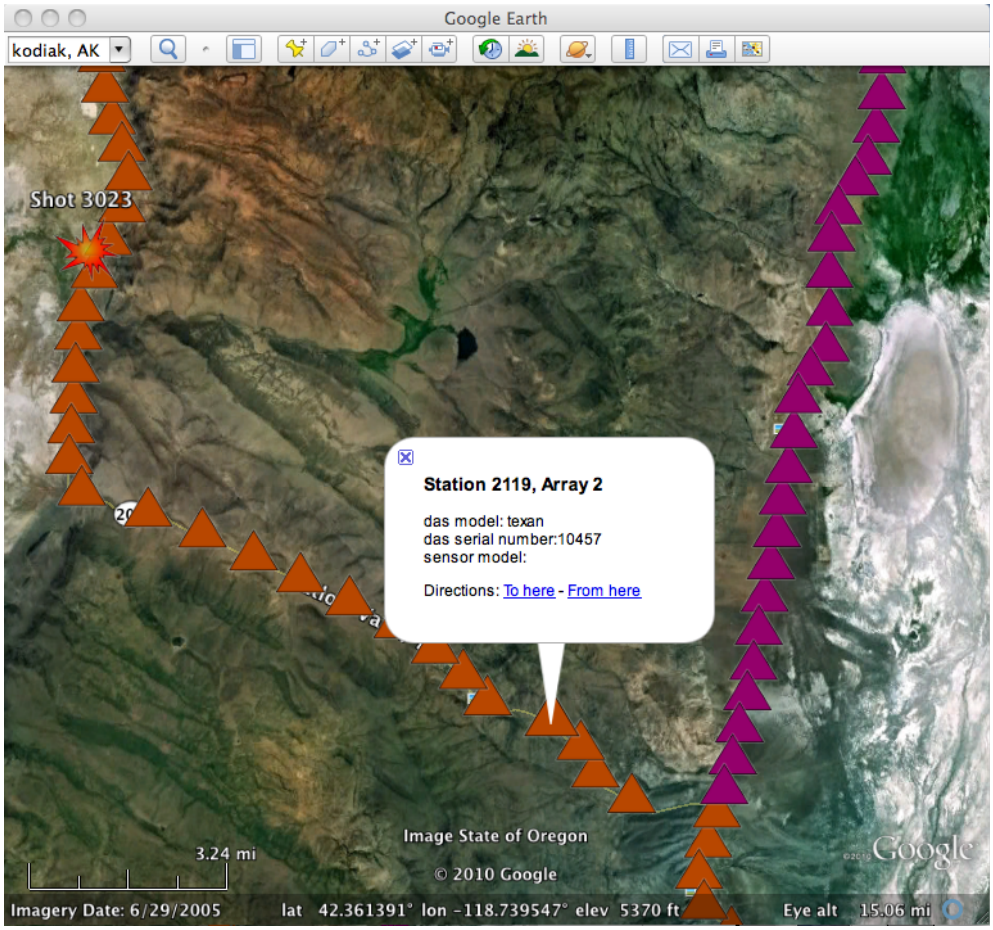

Figure 3. A close-up view of a receivers in a KMZ file built by *kmz-builder* loaded into GoogleEarth. Notice the color-coded receiver arrays and meta-data available. While the sensor model is not a variable stored within a dep file, the sensor model may be populated with by modifying the appropriate KEF file, see Section 4.13.

Presently up to 20 different receiver arrays may be plotted in addition to the sources. Each receiver is plotted as a triangle and organized into color-coded arrays; sources are depicted as explosions. Clicking on receivers and sources provides information such as parent array, DAS serial number, sensor type and shot time. Any errors found in the geometry should be corrected before uploading the dep file into the PH5 file.

4.6 Loading the dep file

The dep file may be loaded into the PH5 file with *dep2ph5,* which populated the appropriate tables with the meta-data contained within the dep files. **Before loading the receiver dep file however, be sure to sort its contains with *sort-recv-dep* has shown below**:

Example: sort-recv-dep receivers.dep > receivers-sorted.dep

sort-recv-dep help:

Usage: sort-recv-dep recv_only_depfile.dep > sorted_depfile.dep

Now load your sorted receivers dep file with *dep2ph5* as shown below – don't forget to also load you shot dep file following the same example as below, remembering to call your shot dep file on the command line in place of the sorted receiver dep file. (**Note the shot dep file does not require sorting**.)

Example: dep2ph5 -n 08-021 -d receivers-sorted.dep >& dep2ph5.out

dep2ph5 help:

```
Usage: dep2ph5 [--help][--dep dep_file] --nickname output_file_prefix

Read a dep file into PH5 format.

Options:
  -h, --help            show this help message and exit
  -n output_file_prefix, --nickname=output_file_prefix
                        The PH5 file prefix (experiment nick name).
  -d dep_file, --dep=dep_file
```

4.7 Viewing the Contents of a PH5 File: hdfview

The contents of a PH5 file may be viewed directly with *hdfview* (may be called with *hdfview.sh* on some installations). Click on the Experiment Group, for example, and you find the four primary groups, described earlier, within your PH5 file. See **Appendix E**, Hdfview, for tips regarding its usage. Considering reviewing the contents of the Sorts group, where receiver and shot geometry information are stored. Please give it a try as you will likely use it later during processing. **Note: open all PH5 files in read-only mode (refer to the File drop-down menu). Editing a PH5 file with hdfview may result in the corruption of the PH5 file and may necessitate the reprocessing of your data set.**

4.8 Calculating the Source-to-Receiver Offsets for each Station

We are now ready to calculate the receiver offsets as we have loaded the necessary shot and receiver meta-data. The software *geod2kef* calculates receiver offsets and produces a kef file containing the calculated offsets which we load into our PH5 file. Note the offsets produced by *geod2kef* are described in a radial coordinate system. Offsets as defined in the SEG-Y shot gather (and output by our SEG-Y gather software) are simply offsets along a line fit to the receivers for each receiver array.

Example: geod2kef -n 08-021 > 08-021-Offset_t.kef

geod2kef help:

```
Usage: Version: 2009.322
```

```
geod2kef --nickname output_file_prefix [--path][-h][--listellipsoids][--listunits][-U units][-E
ellipsoid]

Read locations and calculate offsets from events to receivers. Produce kef
file to populate PH5 file.

Options:
  -h, --help            show this help message and exit
  -n output_file_prefix, --nickname=output_file_prefix
                        The PH5 file prefix (experiment nick name).
  -p output_file_path, --path=output_file_path
                        Path to directory containing PH5 files. Defaults to
                        current directory
  -U output_units       Units to output offsets in. (Use -u to get list of
                        acceptable units.) Default == 'm' (meters)
  -E calculation_ellipsoid
                        Ellipsoid to use. (Use -e to get a list of acceptable
                        ellipsoids.) Default == 'WGS84'
  -e, --listellipsoids  List available ellipsoids.
  -u, --listunits       List all available output units.
```

Below is the first entry in the offset_t kef file produced by our example usage of *geod2kef*:

```
#   Wed Nov 18 17:25:21 2009   geod2kef version: 2009.322   PH5 version: 2008.346 Alpha
/Experiment_g/Sorts_g/Offset_t
        event_id_s = 3016
        receiver_id_s = 1000
        azimuth/value_f = -67.703303
        azimuth/units_s = degrees
        offset/value_d = 305.315
        offset/units_s = km
```

4.9 Timing-drift Correction Information

The texan data logger has a stable oscillator, however it still has a non-zero drift. By time-stamping the data logger with GPS time before and after deployment we are able calculate the total drift and a correction factor, and incorporate these into a kef file with *time-kef-gen*. The timing corrections may be applied to the data at the requestor's discretion during data requests, which will be addressed in a subsequent section of this document.

Note: RT-130 data loggers regularly sync to GPS time during deployment and therefore do not require use of *time-kef-gen*.

Example: time-kef-gen -n 08-021 > 08-021-Time_t.kef

time-kef-gen help:

```
Usage: time-kef-gen --nickname PH5-file-prefix [-p path]

Generates a kef file to populate Time_t from SOH_a_.

Options:
  -h, --help            show this help message and exit
  -n PH5_file_prefix, --nickname=PH5_file_prefix
                        The PH5 file prefix (experiment nickname).
  -p PH5_path, --path=PH5_path
                        Path to PH5 files. Defaults to current directory.
```

4.10 Sensor Orientation Information

Sensor orientations are not defined in the dep file, therefore it is necessary to populate the PH5 file with this information. To do so we first create a receiver_t.kef file which defines the orientations for the sensor used on the experiment. An example receiver_t.kef file is available in $KITCHEN/examples on your PASSAL laptop from which you may make yourself a copy to edit and load. The example receiver kef file describes three components; **remember to remove the horizontal components' table rows for vertical-only stations.** Note sensor orientation follow that of geologic azimuth and dip and are relative to true North.

4.11 Describing the Event Windows

With the meta-data in place we need to provide an efficient means of referencing the station arrays and recording windows for the data request process; *sort-kef-gen* provides this mechanism. All data windows should start at the same time when utilizing the “-a” option of *sort-kef-gen.* Contact PASSCAL regarding situations where data windows may start at different times.

Run *sort-kef-gen* to build a kef file. This kef file will be loaded into the PH5 in the following section.

Example: sort-kef-gen -n 08-021 -a > 08-021-sort_t.kef

sort-kef-gen help:

```
Usage: sort-kef-gen --nickname PH5-file-prefix --serial-number DAS-SN | --auto [--path
path-to-PH5-files]

Generate a kef file to populate Sort_t.
```

```
Options:
  -h, --help            show this help message and exit
  -n PH5_file_prefix, --nickname=PH5_file_prefix
                        The PH5 file prefix (experiment nickname).
  -p PH5_path, --path=PH5_path
                        Path to PH5 files. Defaults to current directory.
  -s sn, --serial-number=sn
                        DAS to use to get windows.
  -a, –auto             Attempt to auto detect windows.  Windows should start
                      at the same time on all DASs.
-d
```

4.12 Inserting kef Files

All kef files are loaded into the PH5 file by *kef2ph5*. Syntax and format within the kef file are important, and, if incorrect, may introduce errors which could prove challenging to address. For this reason a check or “dry-run” option (“-c”) is available with *kef2ph5* which goes through the entire process necessary to add the kef file contents to the PH5 file, except actually writing to it. **The “-c” usage is strongly encouraged prior to importing user-edited kef files to the PH5 file as the option provides validation of the kef file format and syntax.** Consider writing the standard output and standard error (as shown in the example below) to a file so you will have a record of the dry-run for later review. Review the output file for warnings, errors, or failures. Running egrep -i “err|prob|warn|fatal|fail/kill” on the output file from the dry-run is a useful means of quickly detect problems encountered with kef2ph5.

In the examples below we load the experiment kef file. **Be certain to also test receiver_t.kef file, or any other kef file which has ben hand-edited, prior to loading.**

Example: kef2ph5 -n 08-021 -k experiment_t.kef -c > & experiment_kef2ph5.out

kef2ph5 help:

```
Usage: kef2ph5 --kef kef_file --nickname ph5_file_prefix [--path path]
Version: 2010.200

Update a ph5 file from a kef file.

Options:
  -h, --help            show this help message and exit
  -n OUTFILE, --nickname=OUTFILE
                        The ph5 file prefix (experiment nickname).
  -k KEFFILE, --kef=KEFFILE
                        Kitchen Exchange Format file.
  -p PATH, --path=PATH  Path to directory where ph5 files are stored.
  -c, --check           Show what will be done but don't do it!
```

If the *kef2ph5 -c* (dry-run) command processed without error then we may update the PH5 file.

Example: kef2ph5 -n 08-021 -k experiment_t.kef

Now load the software-produced kef files as shown in the experiment_t.kef immediately above, as checking with kef2ph5's -c option is unnecessary.

The software-produced kef files include:

- 08-021-offset_t.kef
- 08-021-time_t.kef
- 08-021-sort_t.kef

4.13 Altering the Contents of PH5 Tables

There will be occasions to change or better-define existing meta-data within a PH5 file. The process for altering meta-data in a PH5 file is a four step procedure:

1. download the desired table to a kef file via *tabletokef*
2. edit the contents as necessary
3. confirm the kef file syntax and formatting with *kef2ph5 -c*
4. modify PH5 file with *kef2ph5*.

Below is an example of extracting the contents of the array table within the Sorts Group using *table2kef*.

Step 1: download the Array_t_001 from the Sorts group.

Example: tabletokef -n 08-021 -A 1 > 08-021-Array_t.kef

tabletokef help:

```
Usage: tabletokef --nickname ph5-file-prefix options

Dump a table to a kef file.

Options:
  -h, --help            show this help message and exit
  -n ph5_file_prefix, --nickname=ph5_file_prefix
                        The ph5 file prefix (experiment nickname).
  -p ph5_path, --path=ph5_path
                        Path to ph5 files. Defaults to current directory.
  -d
  -E, --Experiment_t    Dump /Experiment_g/Experiment_t to a kef file.
  -S, --Sort_t          Dump /Experiment_g/Sorts_g/Sort_t to a kef file.
```

-O, --Offset_t Dump /Experiment_g/Sort_g/Offset_t to a kef file.
-V, --Event_t Dump /Experiment_g/Sorts_g/Event_t to a kef file.
-A n, --Array_t_=n Dump /Experiment_g/Sorts_g/Array_t_[n] to a kef file.
-R, --Response_t Dump /Experiment_g/Responses_g/Response_t to a kef file.
-P, --Report_t Dump /Experiment_g/Reports_g/Report_t to a kef file.
-C, --Receiver_t Dump /Experiment_g/Receivers_g/Receiver_t to a kef file.
-D das, --Das_t=das Dump /Experiment_g/Receivers_g/Das_g_[das]/Das_t to a kef file.
-T, --Time_t Dump /Experiment_g/Receivers_g/Time_t to a kef file.

Here are the first few lines representing the first table row of output from our example usage of *tabletokef*.

```
#
#       Tue Feb 24 16:24:52 2009        PH5 version: 2008.346 Alpha
#
#   Table row 1
/Experiment_g/Sorts_g/Array_t_001
        id_s = 1000
        location/X/value_d = -120.897075
        location/X/units_s = degrees
        location/Y/value_d = 43.875092
        location/Y/units_s = degrees
        location/Z/value_d = 1308.2317
        location/Z/units_s = meters
        location/coordinate_system_s =
        location/projection_s =
        location/ellipsoid_s =
        location/description_s =
        deploy_time/ascii_s =
        deploy_time/epoch_l = -1
        deploy_time/micro_seconds_i = 0
        deploy_time/type_s =
        pickup_time/ascii_s =
        pickup_time/epoch_l = -1
        pickup_time/micro_seconds_i = 0
        pickup_time/type_s =
        das/serial_number_s = 10026
        das/model_s = texan
        das/manufacturer_s =
        das/notes_s =
        sensor/serial_number_s =
        sensor/model_s =
        sensor/manufacturer_s =
        sensor/notes_s =
        description_s =
        channel_number_i = 1
```

Step 2: Editing the kef file

A kef file may append, update, or delete contents of a table based on the specific syntax in the group-table description. (the line: /Experiment_g/Sorts_g/Array_t_001 in our example kef file). The append option is the default action if nothing follows the group-table description. Notice in earlier described kef files, such as those generated by *geod2kef* or t*ime-gen-kef*, nothing follows the group-table description as these kef files serve to append their contents to the empty tables in the PH5 file created during the initializing of the PH5 file.

To update or delete rows of a table, an ":Update:column_name" or ":Delete:column_name" must be added to the end of the group-table description. Be sure to choose a column which is populated with a unique value (e.g. id_s as shown in the example below).

For example, to change the sensor type from a blank entry to "4.5 Hz Vertical" for row 1 of the Sorts-Array table 1, add the text ":Update:id_s" shown in italicized bold below for clarity, to row 1 and change the sensor type to "4.5 Hz Vertical". If another field is to be used to uniquely identify the table column be sure to use the entire column name, not just part of it (e.g. use "das/serial_number", not just "serial_number") in the update or delete.

```
#
#       Tue Feb 24 16:24:52 2009        PH5 version: 2008.346 Alpha
#
#   Table row 1
/Experiment_g/Sorts_g/Array_t_001:Update:id_s
        id_s = 1000
        location/X/value_d = -120.897075
        location/X/units_s = degrees
        location/Y/value_d = 43.875092
        location/Y/units_s = degrees
        location/Z/value_d = 1308.2317
        location/Z/units_s = meters
        location/coordinate_system_s =
        location/projection_s =
        location/ellipsoid_s =
        location/description_s =
        deploy_time/ascii_s =
        deploy_time/epoch_l = -1
        deploy_time/micro_seconds_i = 0
        deploy_time/type_s =
        pickup_time/ascii_s =
        pickup_time/epoch_l = -1
        pickup_time/micro_seconds_i = 0
        pickup_time/type_s =
        das/serial_number_s = 10026
        das/model_s = texan
        das/manufacturer_s =
        das/notes_s =
        sensor/serial_number_s =
        sensor/model_s = 4.5 Hz Vertical
        sensor/manufacturer_s =
        sensor/notes_s =
        description_s =
        channel_number_i = 1
```

Note, every row requiring an update will require a similar entry in the kef file. **Table rows which do not require edits must be deleted from the kef file - recall the default action of**

a table row entry is to append, so these unedited rows would be inadvertently duplicated in the table if not deleted from the kef file prior to loading with kef2ph5.

Step 3: Testing of the kef file: the “dry-run”.

Test the modified kef file as you would any other kef file prior to populating the PH5 file (remember ? - *kef2ph5 -c*). Search the output for errors.

Example: kef2ph5 -n 08-021 -k 08021-Sorts_Array_t.kef -c > & 08021-Offset_kef2ph5.out

Step 4: Implement the updates of the kef file.

The *kef2ph5* dry-run gave good results; now we are ready to update the PH5 file.

Example: kef2ph5 -n 08-021 -k 08021-Offset_t.kef

4.14 Incorporating Additional Source Information

The size and depth of the shots should be stored in the event table of the sorts group. Follow the same four steps outlined in the previous example regarding the sensor model to update the source information. You will use the “-V” option of *tabletokef* to download the event table contents to a kef file for modification.

4.15 Populating the PH5 file with Keys, Descriptions and Reports

Other documents, papers, reports may also be loaded into the PH5 file if desired, utilizing *report2ph5*. Word documents, ps and text files are acceptable formats; PDFs are encouraged. **Note *report2ph5* also generates a kef file during uploading of the files which is not necessary to load.**

Input will be requested by *report2ph5*. At the minimum the file description field requires user input (bold and italicized below).

Examples: report2ph5 -f data_description.txt -n 08-021
Report title [data_description]:
File suffix [TXT]:
Internal array name [Report_a_data_description]:
File description []: ***DMC data description***
Writing: Report_a_data_description.kef

Help output:
Usage: load-report --file report-file --kef kef-file --nickname experiment-nickname [--path path-to-kef-file]

Load a report (pdf) into a ph5 file.

Options:
-h, --help show this help message and exit
-f REPORT_FILE, --file=REPORT_FILE
The file containing the report, (pdf, doc, ps, etc.).
-k KEF_FILE, --kef=KEF_FILE
Kef file describing row in Report_t for the report.
-n NICKNAME, --nickname=NICKNAME
Experiment nickname.
-p PATH, --path=PATH Path to where ph5 files are stored

Your experiment report, or other documents (maps, tables, etc) maybe loaded into the PH5 file with report2ph5 in the same manner as in the previous example.

5 Sending your Data Set to PASSCAL

PASSCAL staff invite you to send your data by a variety of means, including: FTP, DVD, and USB/fire-wire hard-drive. Hard-drives sent to PASSCAL will be returned to the archiver or PI after the data have been confirmed as archived at the IRIS DMC. DVDs will be recycled or disposed of unless you specifically request they be returned.

Prior to sending or transferring any data sets please contact the data group (data_group@passcal.nmt.edu) so the appropriate staff are aware your data are on the way.

Limit FTP transfers to files less than 2 GB in size, as this is a limit of the protocol. If you choose to ftp your data follow the steps below.

1. Login to qc.passcal.nmt.edu. Use “anonymous” as the user and your email address as the password.
2. Change directories to datadrop/assembled.
3. Transfer your PH5, report(s), data description, and data request key files.

For data sets larger than 2 GB please send your data set by DVD (limited to 4.5 GB) or USB/fire-wire hard drive. Be sure to include a short note listing the DVD or drive contents.

PASSCAL data group staff will review the contents of your data set by:

1. building and reviewing several SEG-Y gathers to ensure the essential meta-data are in place. If you would like to build a SEGY file or gather on your own see **Appendix F,** Construction of Preliminary SEG-Y Gathers.
2. reviewing the report(s) to ensure its contents include: an overview of the experiment, a description of the geographical layout, sensors and sources details
3. reviewing the data description and data request keys

4. produce a kmz file and review a Google Earth map of the source and receivers.

After PASSCAL data group staff have satisfactorily reviewed the data set, it will be sent to the DMC. You will be notified after PASSCAL staff have confirmed the data set has been archived.

Author: George W. Slad, george@passcal.nmt.edu