

Seg2Mat: SEG-2 to MATLAB file converter

Stewart A. Levin

ABSTRACT

To support SEG-2 data import into MATLAB[®], I have modified a SEG-2 converter from the Colorado School of Mines Center for Wave Phenomena Seismic Unix package to convert SEG-2 files to corresponding MATLAB files.

INTRODUCTION

Multiple groups in the Stanford School of Earth Sciences, and the Department of Geophysics in particular, are interested in processing and analyzing data acquired in SEG-2 format (Pullan, 1990). Such data may be acquired in shallow engineering studies or microseismic monitoring of hydraulic fracturing. So when a Ph.D. graduate student in Mark Zoback's geomechanics group needed help reading and understanding a large microseismic dataset in SEG-2 format, I cobbled together a converter based on a modified Seismic Unix (SU) code that I had previously used to convert SEG-2 to an internal SEP format.

The MATLAB *.mat* format is well-suited to the task. The format is painstakingly documented by The MathWorks, Inc. (2013) and is limited by its use of 32 bit offsets to about 2 GB (or 4 GB at a pinch), the same limitation as in the SEG-2 format. The baseline SU *seg2segy* utility is somewhat less well documented both because the output *segy* has a rigid structure and naming convention and because the method for mapping SEG-2 headers, which can be almost arbitrarily named and defined, is difficult for many users to set up. (I will explicate this later.) For MATLAB conversion, however, the output is not a rigid format but instead is a near one-to-one mapping of trace header names and values.

SEG2MAT BASICS

The input to Seg2Mat is a set of SEG-2 files with names ending in the suffix *.dat* which are described by an initial filename following a general pattern $[alpha]+[0-9]+.dat$ and optionally prefixed with a directory name to override the default current directory, for example *FILE045.dat* or */data1/survey/REC0001.dat*, and a count of the maximum number of such files to process with the filename numeric field incrementing by one. The output of Seg2Mat will be a series of new files with the *.dat* suffix changed to *.mat*. So to convert *FFID001.dat* through *FFID100.dat* to *FFID001.mat* through *FFID100.mat* you would invoke the command line:

Seg2Mat FFID001.dat 100

Each output *AAAnnn.mat* file contains a MATLAB structure

```
AAAnnn =
FileHeader: [112xK char]
TraceHeaderNames: [32xL char]
TraceHeaderValues: [LxM double]
TraceDataValues: [NxM double]
```

consisting of a copy of the SEG-2 File Header keyword entries, an array of header names prefixed with 001, 002, 003, ... in order to make indexing the following 2D trace header array convenient, and last the array of data traces. Here *K* is the number of keyword lines in the File Header, *L* the number of trace header entries, *M* the number of traces, and *N* the number of samples per trace. (For readability, you might want to display `transp(AAAnnn.FileHeader)` and `transp(AAAnnn.TraceHeaderNames)`.)

SEG2MAT FINE POINTS

The mapping of input trace headers to MATLAB trace headers is supplied in the precompiled table:

```
ACQUISITION_DATE 6 81
ACQUISITION_SECOND_FRACTION 1 54 1000
ACQUISITION_TIME 7 84
ALIAS_FILTER 5 0 1 71 72
CDP_NUMBER 5 1 1 12
CDP_TRACE 5 1 1 14
CHANNEL_NUMBER 5 1 1 8
CLIENT 3 1
COMPANY 3 2
...
NOTE 0 0
POLARITY 0 0
PROCESSING_DATE 3 9
PROCESSING_TIME 3 10
RECEIVER_GEOMETRY 0 0
RECEIVER_SPECS 0 0
RECORD_INDEX 0 0
SKEW 0 0
SOURCE 0 0
```

```
SOURCE_GEOMETRY 0 0
SOURCE_LINE_ID 0 0
```

which takes some explanation. The first two tokens in each line are the SEG-2 trace header name followed by an action code ranging from 0 to 8. The meaning of the action codes is as follows:

```
0 = copy to output FileHeader (or ignore)
1 = multiply the header value by the 3rd parameter and store as a
    16 bit integer in the 2nd parameter header location
2 = interpret the header as a sort code or units code
3 = copy to line number in reel header specified by 2nd parameter
4 = interpret the header as a trace type code
5 = like code 1, but the 2nd parameter is a data type 0=short, 1=int,
    2=floating point, and the remaining parms are header locations
6 = convert to a year and day of year in two consecutive header locations
    ending with the location of the 2nd parameter
7 = convert to an hour, minute and second in three consecutive header
    locations ending with the location of the 2nd parameter
8 = interpret the header as a geophone type code
```

Most of the output trace header names will match the names used by the SU package to refer to places in their SEG-Y-ish trace header.

EXAMPLE

Converting the test (and debug) record, randi001.dat, provided by Randi Walters produced randi001.mat. A transcript of my loading it into MATLAB follows.

```
7 vostok: matlab
Warning: No display specified. You will not be able to display graphics
```

```
< M A T L A B (R) >
Copyright 1984-2011 The MathWorks, Inc.
R2011b (7.13.0.564) 64-bit (glnxa64)
August 13, 2011
```

To get started, type one of these: helpwin, helpdesk, or demo.
For product information, visit www.mathworks.com.

```
>> load randi001.mat
```

```
>> randi001
```

```
randi001 =
```

```
      FileHeader: [112x13 char]  
      TraceHeaderNames: [32x87 char]  
      TraceHeaderValues: [87x108 double]  
      TraceDataValues: [1400x108 double]
```

```
>> transp(randi001.FileHeader)
```

```
ans =
```

```
ACQUISITION_DATE 27/11/2008  
PinnTech_SeqId 215  
CLOCK_SYNC GPS LOCKED 0  
INSTRUMENT GEOSPACE GEORES 53  
ACQUISITION_SECOND_FRACTION 0.222499000000000  
UNITS METERS  
ACQUISITION_TIME 18:42:15  
JOB_ID [Redacted]  
CLIENT [Redacted]  
TRACE_SORT AS_ACQUIRED  
COMPANY [Redacted]  
OBSERVER [Redacted]  
NOTE
```

```
>> transp(randi001.TraceHeaderNames)
```

```
ans =
```

```
001 tracl  
002 tracr  
003 fldr  
004 tracf  
005 ep  
006 cdp  
007 cdpt  
008 trid  
009 nvs  
010 nhs  
...
```

```
075 syflt
076 gelevflt
077 selevflt
078 sdepthflt
079 gdelflt
080 sdelflt
081 swdepflt
082 gwdepflt
083 modsernum
084 senscalibdb
085 sensorientvertical
086 sensorientinline
087 sensorientcrossline
```

DISCUSSION

The Seg2Mat utility is not fully general purpose, requiring source code editing to overcome some hardwired array dimensions. In addition, providing the file header and trace header keywords in transposed array format may be more desirable. Nevertheless, it is sufficiently useful in its present form that I now advertise it to SEP sponsors. Source code download is at <http://sepwww.stanford.edu/oldsep/stew/Seg2Mat.c> and compiled with `gcc -g Seg2Mat.c -DMATOUTPUT -lm -o Seg2Mat`. (Use `-DSEPOUTPUT` for SEP format output instead. In that setting, set the environment variable `DATAPATH` to control where the trace headers and trace data are stored instead of in the current directory.)

REFERENCES

- Pullan, S. E., 1990, Recommended standard for seismic (/radar) files in the personal computer environment: *Geophysics*, **55**, 1260–1271.
- The MathWorks, Inc., 2013, MATLAB[®] MAT-File Format, **r2013b**: http://www.mathworks.com/help/pdf_doc/matlab/matfile_format.pdf.