

# **Troubleshooting & identifying errors in *basic data processing with antelope* – Archiving oriented**

Version: 2012.286

Category: Passive source, Antelope processing for SEED data

Objective: User tips for troubleshooting Antelope database tools while processing data for archiving.

POSSIBLE PROBLEM	SOLUTION/APPROACH
Did not find files	Verify path where your files, waveforms, pf files are located. Please check paths where your mseed data are, and the new wfname file-naming convention specified with the <code>-w</code> option.
Cannot find miniseed2days or any other antelope tool	<p><b>miniseed2days</b> is part of the antelope distribution so if antelope is installed it should be available.</p> <p>Check if you can a do man on miniseed2days or any other tool [weird syntax, think about changing]</p> <p>Verify that antelope is sourced, if not source it (source /opt/antelope/current-version/setup.csh or .sh) depending on your preference or platform</p> <p>Verify antelope installation (run check_antelope_installation from the command line) if ok, check path or source</p> <p>Verify your license (run check_license from command line)</p>
Bad blockettes	<p>Some miniseed files may have an error/warning about bad blockettes or out of order blockettes. When this happens, sometimes, by using miniseed2days with the <code>-DU</code> option (please see man page for detail) it will let you pass over this warning.</p> <p>Feel free to contact us if you have more questions at: <a href="mailto:data_group@passcal.nmt.edu">data_group@passcal.nmt.edu</a>.</p>
Bad Quality	<p>QT1395090109.ALL miniseed2days: bad clock quality = 0 : dropping blockette #1 for QT_1395_BHZ_0 QT1395090109.ALL</p> <p><b>SOLUTIONS</b></p> <p>1)Using the <code>-k</code> option (retain all input data blocks, regardless of clock timing quality values) will allow you to generate mseed day volumes.</p> <p>miniseed2days -k -d my_db -w "day_volumes/{sta}/{sta}.{net}.{loc}.{chan}.{Y.%j}" my_raw/</p>

## Out of order block

The error may look like:

```
miniseed2days: appending to ./day_volumes/1395/1395.QT..OCF.2009.009
miniseed2days *fatal*: block #6 (QT_1395_OCF_, t= 1/09/2009 (009)
21:05:59.000) is at or before the previous block for this channel.
```

### Solution:

Using the `-DU` or `-Du` (depending on how strict you want to run this option) option will keep a record of the previously written time data block and will not write a second block with the same or previous time.

The command line should look like :

```
miniseed2days -k -Du -d my_db -w
"day_volumes/{sta}/{sta}.{net}.{loc}.{chan}.{Y}.{j}" my_raw/
```

for more details please read man page on miniseed2days

*What do to when would like to avoid the long command line when running miniseed2days*

```
miniseed2days -d my_db -w "day_volumes/{sta}/{sta}.{net}.{loc}.{chan}.{Y}.{j}" my_raw/
```

**If you have root permission**, under `/opt/antelope/current-version/data/pf` you can simply edit the `pf` file for `miniseed2days` and change the default for `wfname`

from:

```
wfname %Y/%j/{net}_{sta}_{chan}_{loc}.msd
minimum_timing_quality 10
```

```
pf_revision_time 1237332028
```

to:

```
wfname mseed_dayv_s1/{sta}/{sta}.{net}.{loc}.{chan}.{Y}.{j}
minimum_timing_quality 10
```

By editing this file and changing the `wfname` default you will avoid having to use the `-w` option in the command line so now you can simply type in the command line:

```
miniseed2days -d my_db my_raw/
```

You may need to edit this file each time you have a new service so each set of day volumes for each service is named accordingly, for example for `service2` , you'll edit:

```
wfname mseed_dayv_s2/{sta}/{sta}.{net}.{loc}.{chan}.{Y}.{j}
minimum_timing_quality 10
pf_revision_time 1237332028
```

... and so on.

**If you don't have root permission to write under /opt/antelope/4.11/data/pf then proceed as follows:**

1) Copy default file in your local directory where you will run *miniseed2days*:

```
<my_cpu:my_experiment> cp /opt/antelope/current-version/data/pf/miniseed2days.pf .  
Please notice the dot at the end of the command (for copying locally)
```

2) Edit the *miniseed2days.pf* file as above

3) Set the *PFPATH* environment (path for parameter files in antelope)

For *tsch/csh* type:

```
<my_cpu:my_experiment> setenv PFPATH $ANTELOPE/data/pf:.
```

For *bash*, type:

```
<ny_cpu:my_experiment> export PFPATH= $ANTELOPE/data/pf:. For bash
```

**Please notice the “ :. ” used to read defaults parameter files (under /opt/antelope/current-version/data/pf) PLUS customized file (your *miniseed2days.pf* file in local directory)**

4) Run ***miniseed2days*** as:

```
miniseed2days -d my_db my_raw/
```

Table 7. Possible errors when evaluating traces with *dbe/dbpick*

Check	Issues on traces	Comments
dbe (db editor)	Coverage	Helps to visualize completeness per station/channel/network in the <i>wfdisc</i> table
	Overlaps	Usually generated when traces added more than once to the db. Highlighted in orange when viewing waveforms ( <i>dbpick</i> )
	Gaps	Helps to identify data missing from the db. These can either be overlooked traces or real gaps.

Note: you can use *pql* also to identify coverage, overlaps or gap issues. You can find *pql* as part of the software release by PASSCAL (<http://www.passcal.nmt.edu/content/software-resources>)

Table 8. Possible errors when running dbverify – consistency and checks of integrity of information in database and traces

Check	Issues on traces & meta-data	Comment
dbverify	1) Non-described channels/stations	Comparing cmd output & batch file to identify the reason for this warning
	2) Multiple configurations for same time frame	Mainly due to bad closing times in the batch file or multiple configurations for the same station in one day without proper closing
	3) Removed files	Missing files in original path
	4) Duplicate record	Same record in multiple wfdisc entries

### Examples of common issues:

#### 1) Problem: Non-described channels/stations

##### Example of output:

The following records of AGU\_PIC.wfdisc did not have corresponding rows in sensor:  
Record #149 : sta = BHM8 chan = BHZ time = 8/12/2009 1:16:34.000 endtime = 8/13/2009 1:50:34.000  
Record #150 : sta = BHM8 chan = BHZ time = 8/15/2009 3:04:24.000 endtime = 8/16/2009 3:39:44.000

##### Possible solutions:

- Evaluate your batch file and the configuration described for this station & check:
  - a. station configuration
  - b. start/end times
  - c. channel description
  - d. location code if use in batch or trace
  - e. sample rate (s)
  - f. Once mismatching information is identified, edit batch file & rebuild db

#### 2) Problem: multiple configurations for same station/day

##### Example of output:

Comment check that **calib doesn't change in the middle of a waveform**

dbopen calibration dbjoin wfdisc sta chan time

#time::endtime dbverify 0 sta chan time wfdisc.time wfdisc.endtime

Record # 0 : sta = BARR chan = BHE time = 4/12/2009 0:00:00.000 endtime = 12/31/2009 23:59:58.999 0  
 BARR BHE 1239494400.00000 4/11/2009 16:40:45.525 4/12/2009 0:00:27.600  
 Record # 1 : sta = BARR chan = BHN time = 4/12/2009 0:00:00.000 endtime = 12/31/2009 23:59:58.999 0  
 BARR BHN 1239494400.00000 4/11/2009 16:40:45.525 4/12/2009 0:00:32.500  
 Record # 2 : sta = BARR chan = BHZ time = 4/12/2009 0:00:00.000 endtime = 12/31/2009 23:59:58.999 0  
 BARR BHZ 1239494400.00000 4/11/2009 16:40:45.525 4/12/2009 0:00:06.475

### **Possible Solutions:**

- Read output and try to identify the records or field it refers to, you can use mseedhdr or mseedpeek, these two tools will help to identify possible changes on the configuration of the station (for example if sample rate or instrumentation was modified during the day while servicing, antelope will have trouble adding multiple instrument response on a day volume file)
- Verify time when configuration changed for station (i.e. BARR), in this case you may need to go back to your field notes and identify the times when changes were done so they can be properly described in the batchfile (and therefore dataless)
- Check/add closing statements for station configurations.
- Rebuild your database – this should be done only once all previous checks have been done, if something is wrong on the batch file or data, it should be properly described before rebuilding, otherwise it will keep getting the same errors.

### **3) Duplicated wfdisc records**

#### **Example of output:**

Keys for records #274072 and #274073 in table wfdisc match:  
 sta ABRA | ABRA chan BHE | BHE time 1/01/2009 0:00:41.050 | 1/01/2009 0:01:06.525  
 endtime  
 1/01/2009 0:01:07.500 | 1/01/2009 0:01:07.500

### **Possible Solutions:**

- Open the table in question with dbf (usually wfdisc)
- Identify reported records (i.e. 274072 & 274073)
- Plot and identify if overlaps are simply due to buffer packing
- If significant overlap, remove duplicated records
- If resulting from buffer packing - ignore

### **4) Dbverify :missing wfdisc files - Files that were added to the database cannot be found in their original location**

#### **Example of output:**

Waveform file 'MSEED/SC44.XR..BHE.2008.220' does not exist.

Waveform file 'MSEED/SC44.XR..BHE.2008.221' does not exist.  
Waveform file 'MSEED/SC44.XR..BHE.2008.222' does not exist

#### Possible solutions:

- Move the files back
- Fix the chanids (dbfixchanids)
- Remove the wfdisc table and add the traces again if needed
- Do nothing - sometimes traces are moved on purpose to open space.

Table 9. Possible errors when verifying the database with dbversdwf

Check	Issues	Comments
dbversdwf	BAD records	<p>Command line: dbversdwf -dtu your_db &gt; dbversdwf.out</p> <p>These warnings are usually associated to:</p> <ul style="list-style-type: none"> <li>• Bad endianness</li> <li>• Bad logical record</li> <li>• Stein compression issues</li> <li>• Corrupted blockette</li> </ul>

#### Possible Solutions:

- Look at the traces with any viewer (e.g. pql) to identify issues
- Move bad records aside-remove from db
- Run ckMseed to identify bad traces/errors
- Use mseedhdr to look at order of the trace
- Use fixhdr if needed, to change endianness
- Once fixed (if possible) add back to your database

Table 10. Possible errors when generating database from dataless or when generating dataless

error	output	solution
Missing originating_organization	<p>Command line:</p> <p>1) To generate database from dataless: db2sd -v YoseRocks_dataless X7.08.YoseRocks.20090281014.dataless</p> <p>db2sd *fatal*: Please fill in the 'originating_organization' in db2sd.pf</p>	<p>Under the directory: \$ANTELOPE/data/pf/ You may need to edit the file site.pf to add the originating_organization.</p> <p>An example of the site.pf file at PASSCAL looks like:</p>

	<p>or</p> <p>2) To generate dataless from database</p> <p>Command line:  mk_dataless_seed -v YoseRocks_dataless  X7.08.YoseRocks.20090281014.dataless</p> <p>db2sd *fatal*: Please fill in the  'originating_organization' in db2sd.pf</p>	<pre>default_seed_network  PI Institution           IRIS PASSCAL originating_organization IRIS PASSCAL mail_domain passcal.nmt.edu mailhost mail.passcal.nmt.edu</pre>
<b>Exact same locations for more than one station</b>	<p>db2sd: Offending wfdisc row is T2S6:VMW  12/16/2004 12:38:28.000</p> <p>db2sd: Multiple sensor/site/sitechan entries  for wfdisc record</p> <p>db2sd: Offending wfdisc row is T2S6:VPB  12/16/2004 12:38:28.000</p> <p>db2sd *fatal*: 240 problems -- won't create  SEED volume</p>	<p>Verify if station location is  correct.</p>