```
#comment: This is a batch file example.

net PI Pier database at PASSCAL

sta WH05 34.0740 -106.9188 1.43 Socorro, NM, USA
time 08/13/2009 00:00:00
datalogger  q330_PIC-3chan 0984
sensor trillium_120 0 T4E06
axis Z 0 0 - 1 1
axis N 0 90 - 2 1
axis E 90 90 - 3 1
samplerate 40sps
channel Z BHZ
channel N BH1
channel E BH2
samplerate 1sps
channel Z LHZ
channel N LH1
channel E LH2
add

close WH05 04/25/2010 23:59:59

sta WH06 34.6577 -106.9244 1.43 Socorro, NM, USA
time 08/27/2009 00:00:00
datalogger q330_PIC-3chan 0988
sensor cmg3t 0 T4E07
axis Z 0 0 - 1 1
axis N 0 90 - 2 1
axis E 90 90 - 3 1
samplerate 40sps
channel Z BHZ
channel N BH1
channel E BH2
samplerate 1sps
channel Z LHZ
channel N LH1
channel E LH2
add

close WH06 04/25/2010 23:59:59
```


Q330 Data Processing In a Nutshell

You've offloaded a service run and have stacks of multiplexed files. Now what to do with them? Start with organization and quality control (steps 1-5). Build the database (6-10) and the dataless (11-12). Then send the day volumes and dataless to PASSCAL (13). **Unix commands (bold print)** and any command line arguments are highlighted in yellow. Input files are denoted by <*filename*>.

1. Start by creating a main directory for the project. Once the project directory is made, create a subdirectory within it for the raw data from your service run. For example: **mkdir** service1_RAW

If you used B14 balers copy the multiplexed (*.ALL) files to this directory. If you used B44 balers, copy the BALER44 directory on the USB drive to your raw directory and rename it to reflect your station.

cp –rp /Volumes/BALER44 service1_RAW/<*stationName*>

(The –p preserves the timestamp on the files). If your data used more than one USB drive you could call the second one <*stationName*>2.

2. Use a text editor to create a batch file in your project directory describing every station in your network. See the template on page 4 to get started. Be very accurate with your entries – small typos now can cause big headaches later.

3. Split the multiplexed files into station subdirectories. The (-w) flag specifies an organized directory path and the required filename structure that packs the files into station-channel-day-volumes. Use the following command line for B14:

miniseed2days -f –w "day_volumes/%{sta}/%{sta}.%{net}.%{loc}.%{chan}.%Y.%j" <service1_RAW>/*.ALL >& miniseed2days.out

Use this command for a B44:

miniseed2days -f –w "day_volumes/%{sta}/%{sta}.%{net}.%{loc}.%{chan}.%Y.%j" <service1_RAW>/<*stationName*>*/data >& miniseed2days.out

(the '*' after station-name is a wildcard to grab both USB drives' data and the (–f) flag cleans up the data if necessary).

4. Verify the data quality by reviewing the traces and log files (with **qpeek and pql**). (see 'Q330 State of Health (SOH) Channels' at the web link on page 3 for more information). Obvious signs of trouble include loss of GPS timing, overlaps, gaps, corrupted files, etc. Make a note of any problems. Use **fixhdr** to correct any problem headers, mark timing issues, and/or to convert the files to big endianess if they aren't already. For more information on how to use these tools and the Q330 State of Health (SOH) channels, refer to the Appendices at:
http://www.passcal.nmt.edu/content/data-archiving/documentation/passive-source and
http://www.passcal.nmt.edu/content/pql-II-program-viewing-data

5. If you had to alter any of the header information with **fixhdr**, you'll need to re-run **miniseed2days.** This will correctly rename those files before adding them to the database in Step 8. To do this you should make a new output directory and keep the revised files separate from the originals; here we call it day_volumes2

miniseed2days –w "day_volumes2/%{sta}/%{sta}.%{net}.%{loc}.%{chan}.%Y.%j" day_volumes >&miniseed2days2.out

6. Build the Antelope database in your project directory using the batch file you built in Step 2:
dbbuild -b <*dbname*> <*batchfile*> >& dbbuild.out

7. View your database: **dbe** <*dbname*>. You might want to take a quick look at the site table for location inaccuracies and the sitechan table to check that all of your channels and on/off dates are correct. If you find errors or inaccuracies, correct the batch file and repeat Steps 6 & 7. At this point you have a descriptive framework (metadata only) - the next step is to attach the waveforms.

8. When you're certain the database is error-free, link in the waveforms:
miniseed2db <*day_volumes*>/* <*dbname*> >& miniseed2db.out

Note: The day_volumes directory you use in **miniseed2db** depends on whether or not you had to make changes to your headers. Use day_volumes if the headers were correct but use day_volumes2 if you made corrections to your headers.

9. Correlate the channel ids between tables by running: **dbfixchanids** <*dbname*>

10. Verify the correlation of your data and database: **dbversdwf** –tu <*dbname*> >& dbversdwf.out
This checks that the times in the wfdisc agree with the mseed times.
Also run: **dbverify** –tj <*dbname*> >& dbverify.out
This checks only for the consistency of 2-table joins on all possible combinations of database tables. Check the resulting dbverify.out file for errors. If necessary, fix the batch file and repeat Step 6.

11. Create the dataless SEED volume (a.k.a the dataless) with the following naming convention:
mk_dataless_seed –o NN.YY.dbname.YYYYDOYHHMM.dataless <*dbname* >
Where: **NN** is your network code, **YY** is the year of your data, and **DOYHHMM** is the approximate **current** day-of-year-hour-minute. The dataless is a type of index of the metadata that allows you and future users to see what data are available. If any station or time range is missing from the dataless, the corresponding data are orphaned and totally inaccessible by anyone.

12. Verify the dataless.
Run **seed2db** –v NN.YY.dbname.YYYYDOYHHMM.dataless
The program lists all the stations and errors (if any) to standard output.

13. Last step: Please drop a note to *data_group@passcal.nmt.edu* before sending the data to PASSCAL so that we can set up a receiving area. Attach your latest dataless to this email unless it is larger than 5Mb. You can use our tool **data2passcal** to automatically send the data.

A few tips…

Many database errors can be avoided by rounding the start & close times in the batch to 00:00:00 and 23:59:59, respectively. It's better to start early and close late (even by a day or two) to be certain that all data & log files are described in the dataless. To avoid tears when you move a DAS to a new station, be sure that the close date on the first station is *before* the open date on the new station. Station changes, such as a new datalogger, sensor or sample rate, **must** be documented in the batch file as shown in the "Building a Batch File for dbbuild" document found in the Appendices section on our website.

For a much more detailed processing guide, please see the PASSCAL document: "Generating SEED From Q330 Raw Data", available from our web page:
http://www.passcal.nmt.edu/content/data-archiving/documentation/passive-source