

# DBHELI IN RTSYSTEM CRONTAB

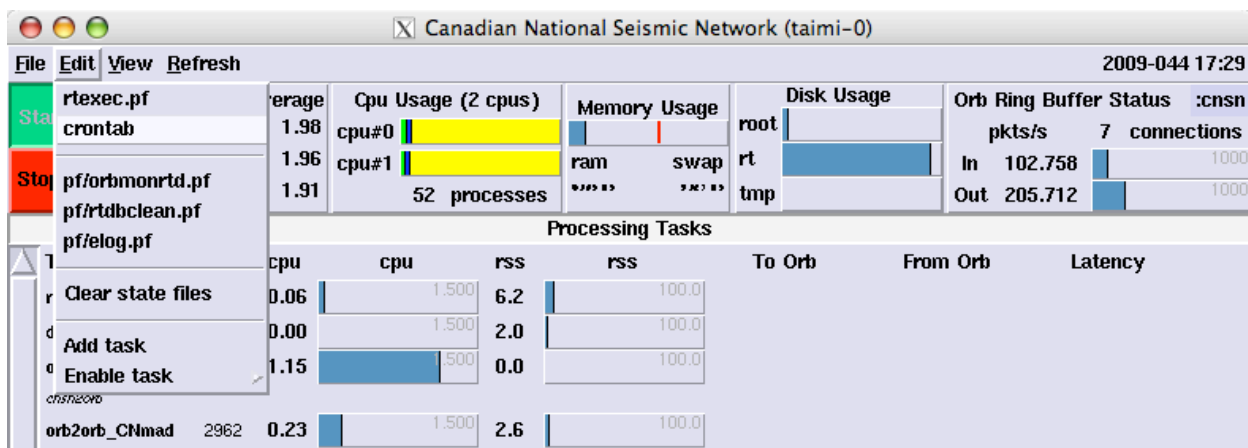
## 24-HOUR & EVENT DISPLAY

On the display computer, find the rtsystem from which the 24-hour plots are created. This is typically located in a directory such as: /home/rt/rtsystems/xxx, where xxx might be “display”, “web”, “cnsn”, “usarray”, etc.

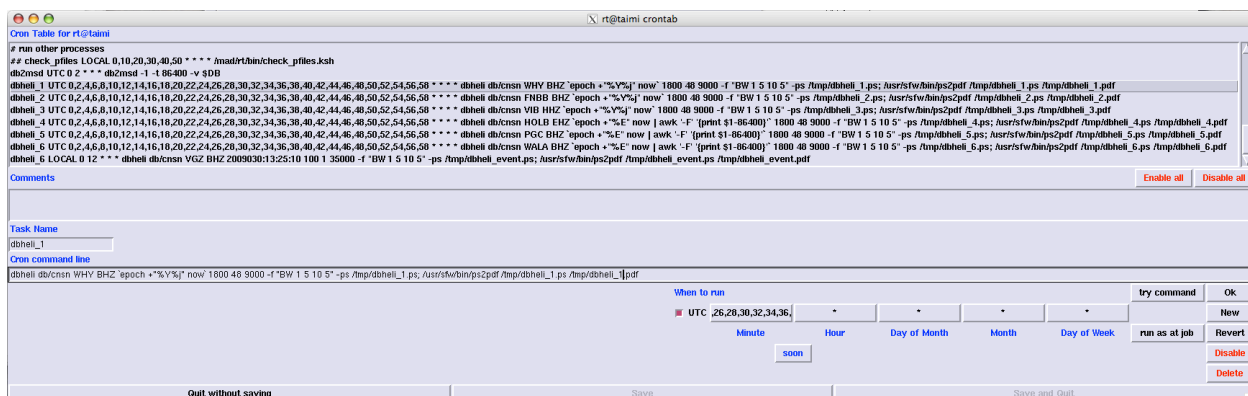
In this directory (this must be the same directory level in which rtexec.pf resides):

**% rtm &**

From the resulting GUI launch the crontab GUI (Edit → crontab):



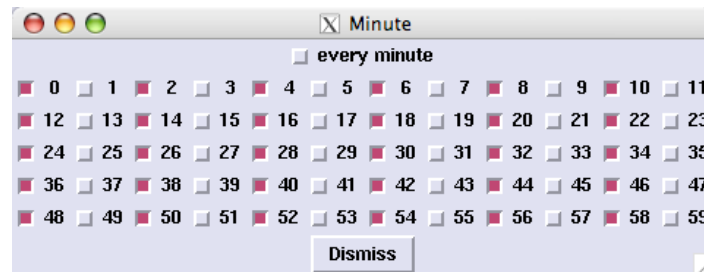
In the crontab GUI, proceed to the bottom of the task list and add in the dbheli task names and cron command lines (click “New”). If these already exist, find the task you wish to change.



In the above example there are seven dbheli tasks, named dbheli\_1, dbheli\_2, dbheli\_3, dbheli\_4, dbheli\_5, dbheli\_6, dbheli\_event. The dbheli command line format follows.

**usage: dbheli dbname sta chan tstart twinline nlines scale**  
**[-f filter] [-o dorigine] [-display] [-ps psfile]**

The general idea is to create a number of dbheli cron jobs, write the postscript output to a generic file name (e.g. dbheli\_1.ps) and then convert the postscript file to a pdf file. The cron jobs are set to run repeatedly with the output file overwriting the previously created file. Each file can then be displayed in a web browser set to automatically refresh, effectively creating “real-time” plots. This allows the user (such as duty personnel) to change the station name and channel within the command line without affecting the file name. This keeps the number of existing files to a minimum and requires no cleanup of unwanted files, unless the number of tasks is reduced. In the above example the dbheli cronjobs are set to run every 2 minutes. This can be altered by clicking on the minute bar in the crontab GUI and selecting the appropriate minutes in the resulting Minute GUI:



The key to keeping the time current in the dbheli command line is the execution of a command in back quotes, which results in a valid current time for tstart:

```
`epoch +"%Y%j" now`  
`epoch +"%E" now | awk '{print $1-86400}'`
```

Epoch is an antelope command to convert time formats, see the antelope man pages or reference guide for allowed formats. Examples of epoch command line execution to display current time and to display formatted current time (year and julian day in this case):

```
% epoch now  
1234546863.014 (044) 2009-02-13 17:41:03.01440 UTC Friday  
% epoch +"%Y%j" now  
2009044
```

## DBHELI CRONTAB COMMANDS

Be aware that differences may exist between running dbheli on the command line and running dbheli within the rtm crontab. Tasks generated from within the crontab are run in the real-time environment defined in rtexec.pf. This may differ from the user login environment.

### a) dbheli plots starting at 00:00 UT

The first three dbheli tasks (dbheli\_1, 2, 3) in the crontab GUI illustrate how to create a plot beginning at 00:00 UT on the current day and displaying waveforms up until the current time. The plot starts over at 00:00 UT the next day. This is equivalent to changing the seismic helicorder paper at 00:00 UT each day. The cron command line to create this is:

```
dbheli db/cnsn WHY BHZ `epoch +"%Y%j" now` 1800 48 9000 -f "BW 1 5 10 5" -ps /tmp/dbheli_1.ps;  
/usr/sfw/bin/ps2pdf /tmp/dbheli_1.ps /tmp/dbheli_1.pdf
```

b) dbheli continuous 24 plots

The next three dbheli tasks (dbheli\_4, 5, 6) in the crontab GUI illustrate how to create a continual 24-hour plot. The downside of this plot is that the hour axis (y-axis) is labeled in hours since the plot start time (tstart), exactly 24 hours previously, rather than UT.

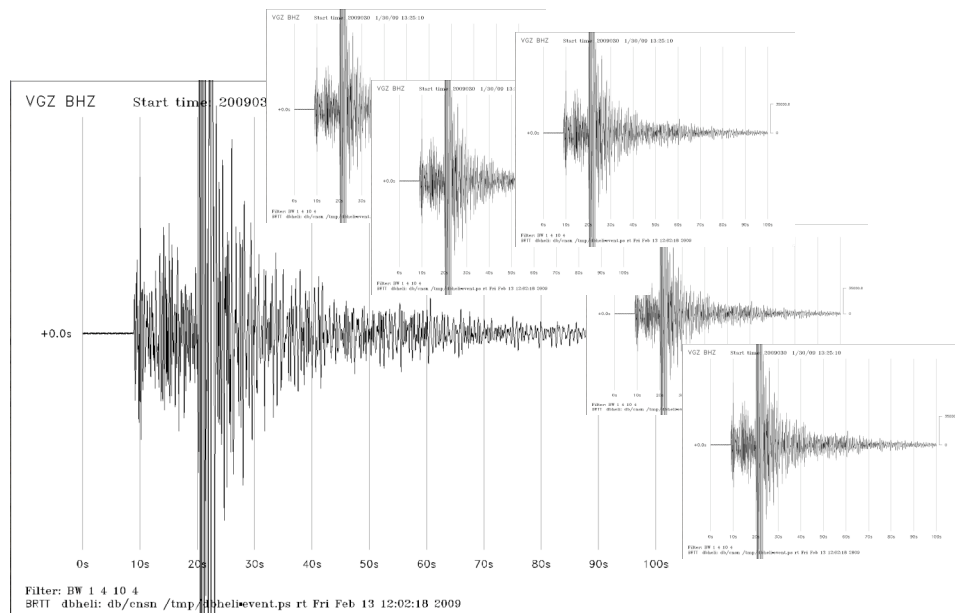
```
dbheli db/cnsn HOLB EHZ `epoch +"%E" now | awk '{print $1-86400}'` 1800 48 9000 -f "BW 1 5 10 5" -ps
/tmp/dbheli_4.ps; /usr/sfw/bin/ps2pdf /tmp/dbheli_4.ps /tmp/dbheli_4.pdf
```

c) dbheli event plots

The last dbheli task, dbheli\_event, in the crontab GUI illustrates how to create a plot of the latest significant event.

```
dbheli db/cnsn VGZ BHZ 2009030:13:25:10 100 1 35000 -f "BW 1 5 10 5" -ps /tmp/dbheli_event.ps;
/usr/sfw/bin/ps2pdf /tmp/dbheli_event.ps /tmp/dbheli_event.pdf
```

In this cron command line, the user or duty personnel will probably want to change the station to the closest station to the event, ensure the station channel code is correct, and change the start time (tstart) to the origin time for the event. This example uses the year and julian day format, YYYYJJ:HH:MM:SS. Some experimentation may be required to get a nice looking plot. This could involve a slightly later tstart and a shorter or longer time window (twinline in seconds). Here twinline = 100 seconds. In the case of the dbheli\_event task, this plot only needs to be created once. It will certainly not be required to update every 2 minutes. For this task, it is probably best to ensure that at least the minute and hour are set to a single choice. In this example, the task recreates the same dbheli\_event plot each day at 12:00 UT. When experimenting to find the best looking plot, the *-display* option is useful. This will produce an image of the plot. Make sure to remove the *-display* option prior to saving this command. Otherwise you will create endless instances of this image and you will likely be beaten with a wet noodle by the person who discovers this for you.

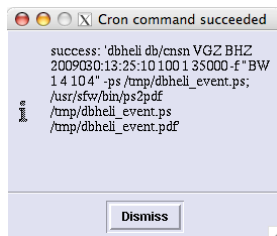


## TESTING & SAVING

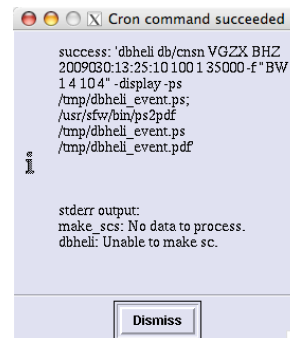
Crontab GUI:

The screenshot shows the Crontab GUI for user 'rt@taimi'. The main window displays a list of cron jobs under the heading 'Cron Table for rt@taimi'. The jobs include system tasks like 'run other processes', 'check\_pfiles', and 'db2msd', as well as several 'dbheli' jobs scheduled at various times. The 'dbheli\_event' job is highlighted. Below the list, there are buttons for 'Enable all' and 'Disable all'. A 'Comments' section is empty. The 'Task Name' field contains 'dbheli\_event'. The 'Cron command line' field shows the full command: 'dbheli db/cnsn VGZ BHZ 2009030:13:25:10 100 1 35000 -f "BW 1 5 10 5" -ps /tmp/dbheli\_event.ps; /usr/sfw/bin/ps2pdf /tmp/dbheli\_event.ps /t'. The 'When to run' section shows the job is set to run at 0 minutes past the hour, every 12 hours, every day of the month, every month, and every day of the week. There are buttons for 'try command', 'Ok', 'New', 'run as at job', 'Revert', 'Disable', and 'Delete'. At the bottom are buttons for 'Quit without saving', 'Save', and 'Save and Quit'.

Test commands first by clicking “try command”. A message will appear with a status report:



Success.



Command ran, but no data to process due to error in command line.

When command works as intended, click “Ok”. This clears the command from the “Cron command line” and updates the cron command line in the crontab screen. When finished with all changes click “Save” or “Save and Quit” to write changes to rtxec.pf for execution.

## OUTPUT & DISPLAY

These crontab examples illustrate plots that are written to /tmp, with the following plot names. The web browser should point to each of these files. Ensure the web browser is set to automatically refresh at a rate consistent with the generation rate of these plots.

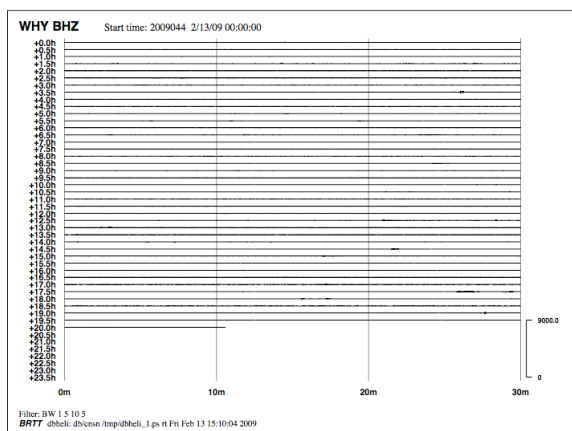
```
% ls /tmp/dbheli*
```

```
/tmp/dbheli_1.pdf   /tmp/dbheli_3.ps   /tmp/dbheli_6.pdf
/tmp/dbheli_1.ps    /tmp/dbheli_4.pdf  /tmp/dbheli_6.ps
/tmp/dbheli_2.pdf   /tmp/dbheli_4.ps   /tmp/dbheli_event.pdf
/tmp/dbheli_2.ps    /tmp/dbheli_5.pdf  /tmp/dbheli_event.ps
/tmp/dbheli_3.pdf   /tmp/dbheli_5.ps
```

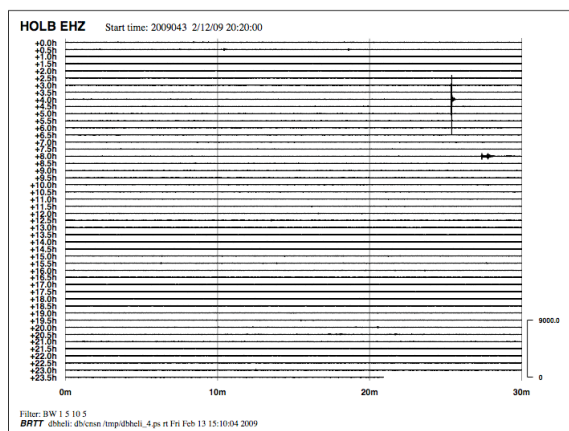
If the number of dbheli cron tasks have been reduced, remove those corresponding output files.

### Display plots

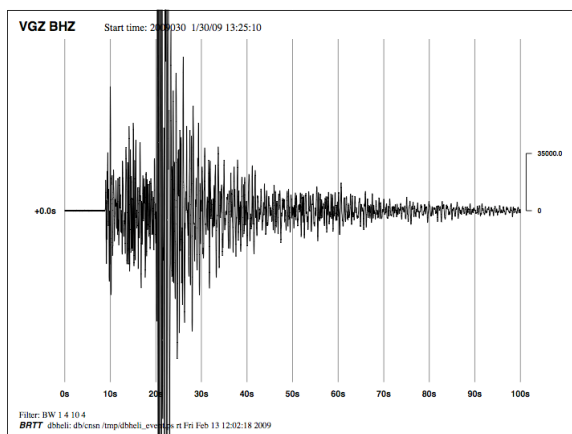
dbheli plots starting at 00:00UT  
(dbheli\_1.pdf)



dbheli continuous 24 hour  
(dbheli\_4.pdf)



dbheli event plots (dbheli\_event.pdf)



## REFERENCE GUIDE - dbheli display plots

**Go to rtsystem directory (rtexec.pf directory) from which the display is run & bring up the real-time monitor, i.e.**

```
% cd /home/rt/rtsystems/display (or the equivalent in your directory structure)
```

```
% rtm &
```

    Edit → crontab

**For dbheli information:**

```
% man dbheli
```

```
% dbheli
```

**usage: dbheli dbname sta chan tstart twinline nlines scale**

**[-f filter] [-o dborderin] [-display] [-ps psfile]**

**dbheli plots starting at 00:00 UT**

```
dbheli db/cnsn WHY BHZ `epoch +"%Y%j" now` 1800 48 9000 -f "BW 1 5 10 5" -ps /tmp/dbheli_1.ps;  
/usr/sfw/bin/ps2pdf /tmp/dbheli_1.ps /tmp/dbheli_1.pdf
```

**dbheli continuous 24 hour plots**

Note: time axis in hours since tstart

```
dbheli db/cnsn HOLB EHZ `epoch +"%E" now | awk '{print $1-86400}'` 1800 48 9000 -f "BW 1 5 10 5" -ps  
/tmp/dbheli_4.ps; /usr/sfw/bin/ps2pdf /tmp/dbheli_4.ps /tmp/dbheli_4.pdf
```

**dbheli event plots**

```
dbheli db/cnsn VGZ BHZ 2009030:13:25:10 100 1 35000 -f "BW 1 5 10 5" -ps /tmp/dbheli_event.ps;  
/usr/sfw/bin/ps2pdf /tmp/dbheli_event.ps /tmp/dbheli_event.pdf
```

**preview event plots, use *–display* option**

```
dbheli db/cnsn VGZ BHZ 2009030:13:25:10 100 1 35000 –display -f "BW 1 5 10 5" -ps /tmp/dbheli_event.ps;  
/usr/sfw/bin/ps2pdf /tmp/dbheli_event.ps /tmp/dbheli_event.pdf
```

\* Remove *–display* option prior to crontab “Save and Quit”.

**To test, click:**

“try command”

**To save changes, click in the following order:**

“Ok”

“Save” or “Save and Quit”