

setup and configuration of Mac Mini

All Sky camera for LSST site- configuration of Mac Mini with software

Nov 29 2013. Setting up mac mini to run Canon camera:

Nov 29 2013.

Setting up Mac mini for gphoto2 using Dave Monet's notes:

libtool-2.4.2:

```
./configure --prefix=/home/dgm/linux  
make  
make install
```

libusb-1.0.9:

```
./configure --prefix=/home/dgm/linux  
make  
make install
```

libgphoto2-2.5.2:

```
setenv CPPFLAGS "-I/home/dgm/linux/include"  
setenv LDFLAGS "-L/home/dgm/linux/lib"  
setenv PKG_CONFIG_PATH /home/dgm/linux/lib/pkgconfig  
./configure --prefix=/home/dgm/linux  
make  
make install
```

gphoto2-2.5.2:

```
./configure --prefix=/home/dgm/linux  
make  
make install  
  
(does not need the CPPFLAGS and LDFLAGS of libgphoto)
```

downloaded

libgphoto2-2.5.2

gphoto2-2.5.2

libusb-1.0.9

libtool-2.4.2

I am going to set this up like Dave Monet's configuration, in a directory called linux off the user's home directory.

did sudo xterm

put the directories into /usr/local as root.

trying libtool: rats, no c compiler on the path!

so for that I need Xcode developer's toolkit

downloading that from Apple web site, 1.7 GB!

after some pain downloading and installing Xcode for this version of OSX,

trying again....

nope. It turns out I also need to install "Command Line Tools" for Xcode

ok, now we have gcc.

did same installation as Dave Monet has, except replace

/home/monet/linux

with

/Users/christopherstubbbs/linux

seemed to work for libtool and libusb. Had to switch to tcsh for setenv to work.

got a make error for libgphoto2. Try putting pkgconfig on path with

set path = (\$path /Users/christopherstubbbs/linux/lib/pkgconfig)

nope- try downloading and installing it

did

./configure --prefix=/Users/christopherstubbbs/linux --with-internal-glib

now try libgphoto again

set path = (\$path /Users/christopherstubbbs/linux/bin)

did setenv to this

set path = (\$path /Users/christopherstubbbs/linux/bin)

got compiler errors for libgphoto2, fixed some undeclared variables in

linux.c but then encountered

```
libtool: link: gcc -o .libs/ptp2.so -bundle  ptp2/.libs/ptp.o ptp2/.libs/library.o ptp2/.libs/usb.o ptp2/.libs/ptpip.o ptp2/.libs/config.o ptp2/.libs/olympus-wrap.o -L
/Users/christopherstubbbs/linux/lib ../libgphoto2/.libs/libgphoto2.dylib /usr/local/libgphoto2-2.5.2/libgphoto2_port/libgphoto2_port/.libs/libgphoto2_port.dylib -
lm ../libgphoto2_port/libgphoto2_port/.libs/libgphoto2_port.dylib /Users/christopherstubbbs/linux/lib/libltdl.dylib -lpthread -liconv -O2 -Wl,-dylib_file -Wl,/Users
/christopherstubbbs/linux/lib/libgphoto2_port.10.dylib:/usr/local/libgphoto2-2.5.2/libgphoto2_port/libgphoto2_port/.libs/libgphoto2_port.dylib -Wl,-
exported_symbols_list,.libs/ptp2-symbols.expSYM
```

Undefined symbols for architecture x86_64:

"_olympus_setup", referenced from:

_camera_init in library.o

ld: symbol(s) not found for architecture x86_64

collect2: ld returned 1 exit status

make[3]: *** [ptp2.la] Error 1

make[2]: *** [all-recursive] Error 1

make[1]: *** [all-recursive] Error 1

make: *** [all] Error 2

[server:/usr/local/libgphoto2-2.5.2] root#

and gave up.

OK so let's try a totally different way- macports.

installed macports for Lion, 10.7

it puts the executable into /opt/local/bin so execute that explicitly,

did port -v selfupdate and that worked.

also did install of gphoto2 using ports and that worked as well,

although it stuffed things into /opt/local/bin, so need to add that to path.

This is a good resource:

<http://photolifetoy.blogspot.com/2012/08/control-your-camera-with-gphoto2-via.html>

Before running gphoto2 you need to terminate this "hijack" by running the following command:

killall PTPCamera

This will need to be done every time the camera is plugged in or turned on while connected to your mac.

/opt/local/bin/gphoto2 --summary

/opt/local/bin/gphoto2 --capture-image-and-download

works!!

to do next:

add /opt/local/bin to path

get JT's analysis stuff installed here.

Figuring out when to start taking images:

Nov 30, 2013.

Twilight times

Pachon is at West long of 70.7333, lat -30.2333

used USNO site http://aa.usno.navy.mil/data/docs/RS_OneYear.php to determine civil twilight times in UT.

Data file is here [pachon_twilight.dat](#)

broke that into one file per month, called mxx.dat with xx ranging from 01 to 12. Did

grep -v "%" pachon_twilight.dat | awk '{print \$1,\$3,\$2}' > m01.dat

grep -v "%" pachon_twilight.dat | awk '{print \$1,\$5,\$4}' > m02.dat

note that this intentionally reverses the times so that sundown in column 1 and sunrise in col 2, in UT

can use this to determine when to start taking data.

OSX cron jobs

1. set up time as GMT
2. sudo touch /etc/crontab
3. crontab -e edits the crontab file.
4. This is a good tutorial: <http://rossb.biz/blog/2011/os-x-cron-jobs-a-simple-tutorial/>
5. syntax we want is to start the task at a sensible time before sunset, say 2200 UT, i.e. 6 pm local. The script should then
 - a. check space avail.

- b. create data directory
- c. change image prefix
- d. cd to data directory
- e. compute appropriate number of images to take
- f. wait for sunset time according to date
 - i. then start collecting N images
 - ii. end when sunup or N is exceeded
 - iii. run simple analysis, convert to fits files, compress data, update status web page
 - iv. end

cron entry time format is

<minute> <hour> <month-day> <month> <week-day> <command to execute>

so we want

```
1 22 * * * /usr/local/bin/startnight.sh
```

I suppose we could just go ahead and use crontab to initiate each night's data collection at the desired time, within a range of dates. Say we break things up into two week intervals. For example we could do

```
1 22 1-15 01 * <tasks for first half of Jan>
```

```
30 22 16-31 01 * <tasks for second half of Jan>
```

etc.

So let's make the table of startup times, in the format crontab wants. Make a list of

minute hour dayrange month

```
15 00 01-15 01
```

```
10 00 16-31 01
```

```
05 00 01-15 02
```

```
45 23 16-28 02
```

```
30 23 01-15 03
```

```
15 23 16-30 03
```

```
00 23 01-15 04
```

```
30 22 16-31 04
```

```
30 22 01-15 05
```

```
15 22 16-31 05
```

```
15 22 01-15 06
```

```
15 22 16-31 06
```

```
40 22 01-15 07
```

```
45 22 16-31 07
```

```
45 23 01-15 08
```

```
45 23 16-31 08
```

```
00 23 01-15 09
```

```
00 23 16-31 09
```

```
15 23 01-15 10
```

```
15 23 16-31 10
```

```
30 23 01-15 11
```

```
45 23 16-31 11
```

```
00 00 01-15 12
```

```
15 00 16-31 12
```

Here is tar file of shell scripts in use as of Jan 15 2014:

[allskyscripts.tar](#)

Here is a dump of crontab -l:

```
59 23 01-15 01 * source ~/startnight.sh 1000 >> ~/cronlog
59 23 16-31 01 * source ~/startnight.sh 1000 >> ~/cronlog
59 23 01-15 02 * source ~/startnight.sh 1000 >> ~/cronlog
45 23 16-28 02 * source ~/startnight.sh 1050 >> ~/cronlog
30 23 01-15 03 * source ~/startnight.sh 1110 >> ~/cronlog
15 23 16-30 03 * source ~/startnight.sh 1200 >> ~/cronlog
00 23 01-15 04 * source ~/startnight.sh 1250 >> ~/cronlog
30 22 16-31 04 * source ~/startnight.sh 1250 >> ~/cronlog
30 22 01-15 05 * source ~/startnight.sh 1300 >> ~/cronlog
15 22 16-31 05 * source ~/startnight.sh 1320 >> ~/cronlog
15 22 01-15 06 * source ~/startnight.sh 1350 >> ~/cronlog
15 22 16-31 06 * source ~/startnight.sh 1350 >> ~/cronlog
40 22 01-15 07 * source ~/startnight.sh 1300 >> ~/cronlog
45 22 16-31 07 * source ~/startnight.sh 1300 >> ~/cronlog
45 23 01-15 08 * source ~/startnight.sh 1300 >> ~/cronlog
45 23 16-31 08 * source ~/startnight.sh 1300 >> ~/cronlog
00 23 01-15 09 * source ~/startnight.sh 1200 >> ~/cronlog
00 23 16-31 09 * source ~/startnight.sh 1200 >> ~/cronlog
15 23 01-15 10 * source ~/startnight.sh 1100 >> ~/cronlog
15 23 16-31 10 * source ~/startnight.sh 1100 >> ~/cronlog
30 23 01-15 11 * source ~/startnight.sh 1000 >> ~/cronlog
45 23 16-31 11 * source ~/startnight.sh 1000 >> ~/cronlog
59 23 01-15 12 * source ~/startnight.sh 1000 >> ~/cronlog
59 23 16-31 12 * source ~/startnight.sh 1000 >> ~/cronlog
```

and as a file: [cronlist.txt](#)

OS version:

server:~ christopherstubbbs\$ system_profiler SPSSoftwareDataType

Software:

System Software Overview:

System Version: Mac OS X Server 10.7.5 (11G63)

Server Configuration: Advanced

Kernel Version: Darwin 11.4.2

Boot Volume: Server HD

Boot Mode: Normal

Computer Name: LSSTserver

User Name: Christopher Stubbs (christopherstubbs)

Secure Virtual Memory: Enabled

64-bit Kernel and Extensions: Yes

Time since boot: 6 days 13:56
