Preliminaries

- Verify that your computer (> hostname) is a GIPSY client.
  If not, add your computer name to the file $gip_loc/clients
- Verify that you run the tc-shell (> echo $SHELL)
- In your home directory, add these lines to the .tcshrc file:

  setenv gip_root <path to gipsy installation>
  source $gip_root/sys/gipenv.csh
- Verify that the GIPSY environment is set (> printenv | grep gip)
- If you do not have a proper installation of GIPSY,
  work together with your neighbour

T8 : Analysis of spectral line data cubes

Marc Verheijen
Kapteyn Institute

Goal : Extract scientifically meaningful information from a 'dirty' HI data cube obtained with a standard 12^{hr} WSRT observation.

Object : NGC 4088 and its companion galaxy.

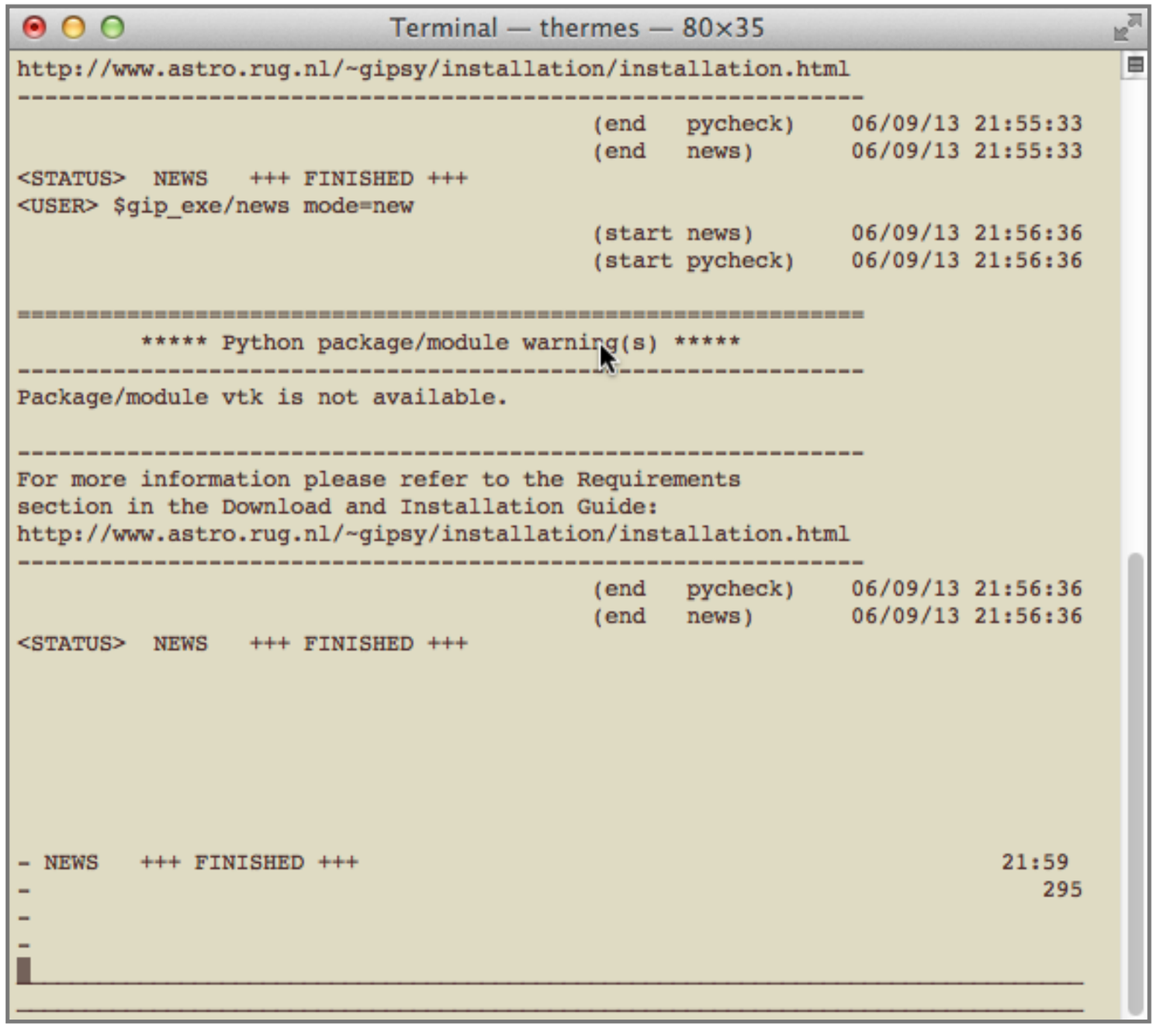
N4088 : $12^h03^m01.97^s$ $+50^d49^m02.3^s$ V_{hel}= 757 km/s
N4085 : $12^h02^m50.41^s$ $+50^d37^m52.4^s$ V_{hel}= 746 km/s

Tool : Groningen Image Processing SYstem (gipsy)
www.astro.rug.nl/~gipsy

We take the most elementary approach.

Introduction to Gipsy interface

Hermes : the terminal-based interface



Common Output Area

toggle *Tab*: hermes help on/off

With task-name on command line,
Tab provides help on task inputs

Ctrl+V/Z : page forward/backward
Ctrl+P : switch page mode on/off
Ctrl+Q : quit Hermes
Esc > / < : go to last/first page
Ctrl-X S : switch scrolling on/off
Esc B : switch bell on/off

active task slots

Ctrl-C : abort an active task

command line editor

↑↓ : scroll input history

Read FITS files into Gipsy GDS database

- Read cubes

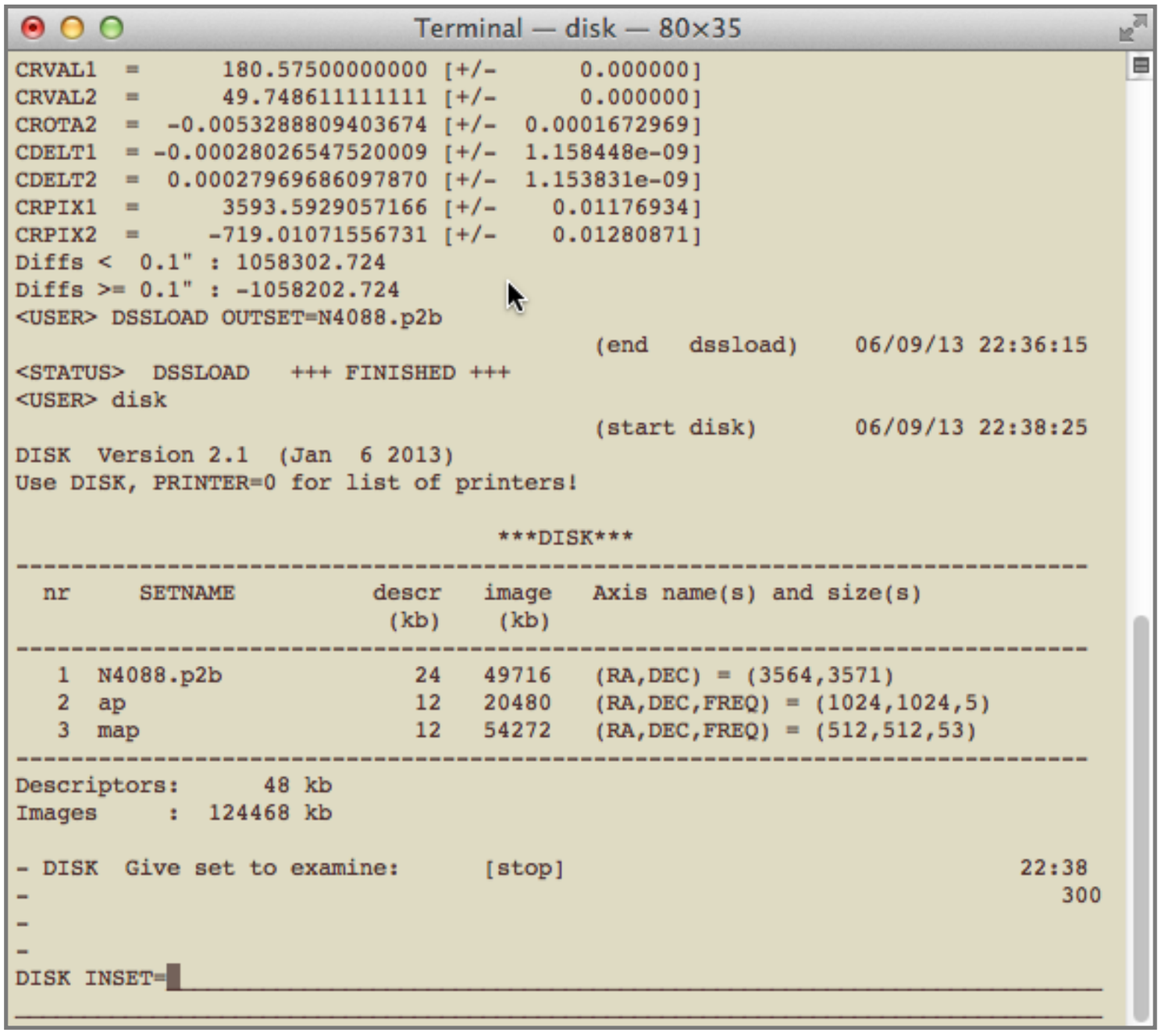
```
rfits
FITSFILE = map.fits
OUTSET = map
FREQ0 = ↩
FITSFILE = ↩
```

- Read POSS-2 image

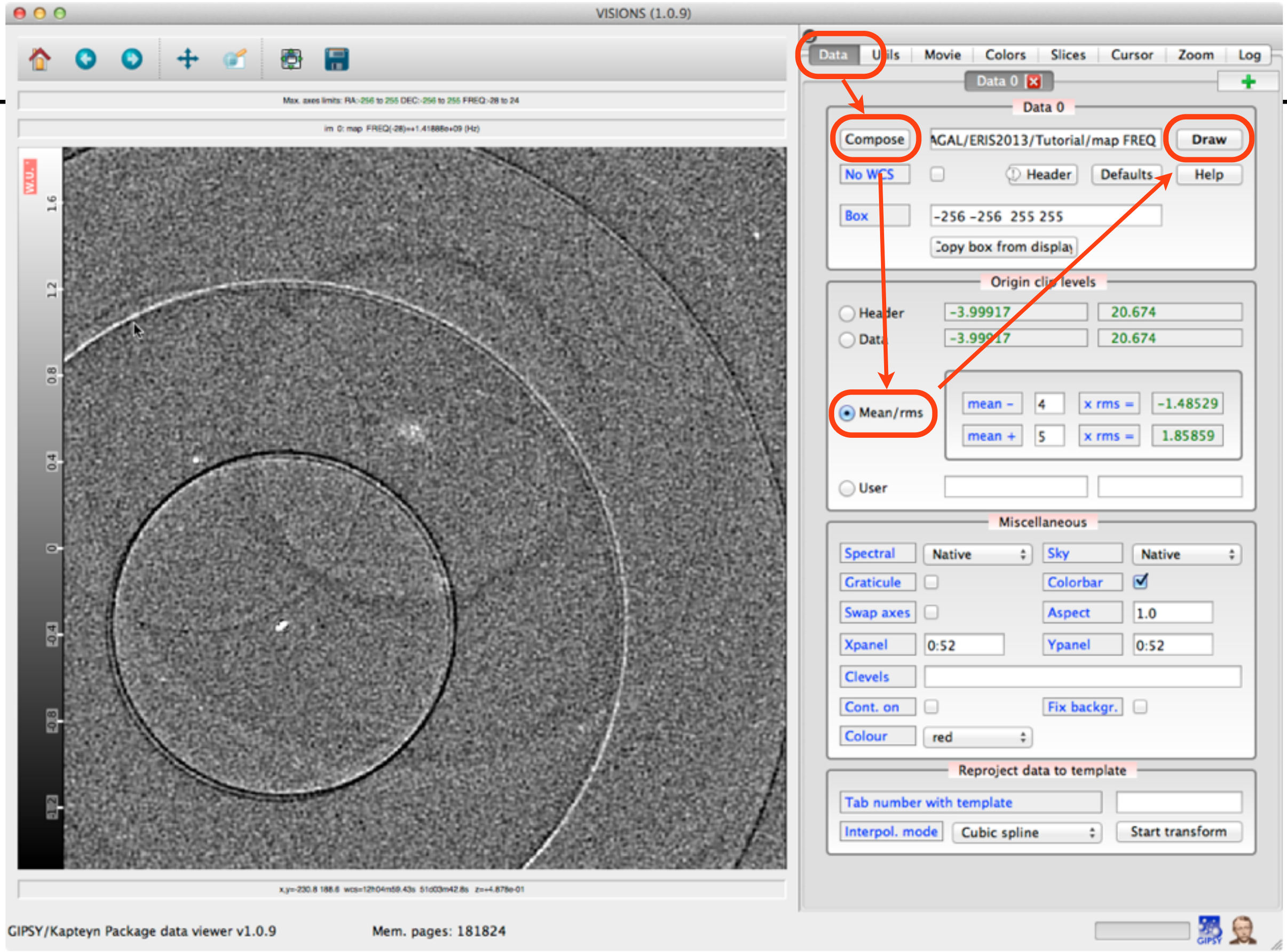
```
dssload
FILENAME = N4088.p2b.fits
OUTSET = N4088.p2b
```

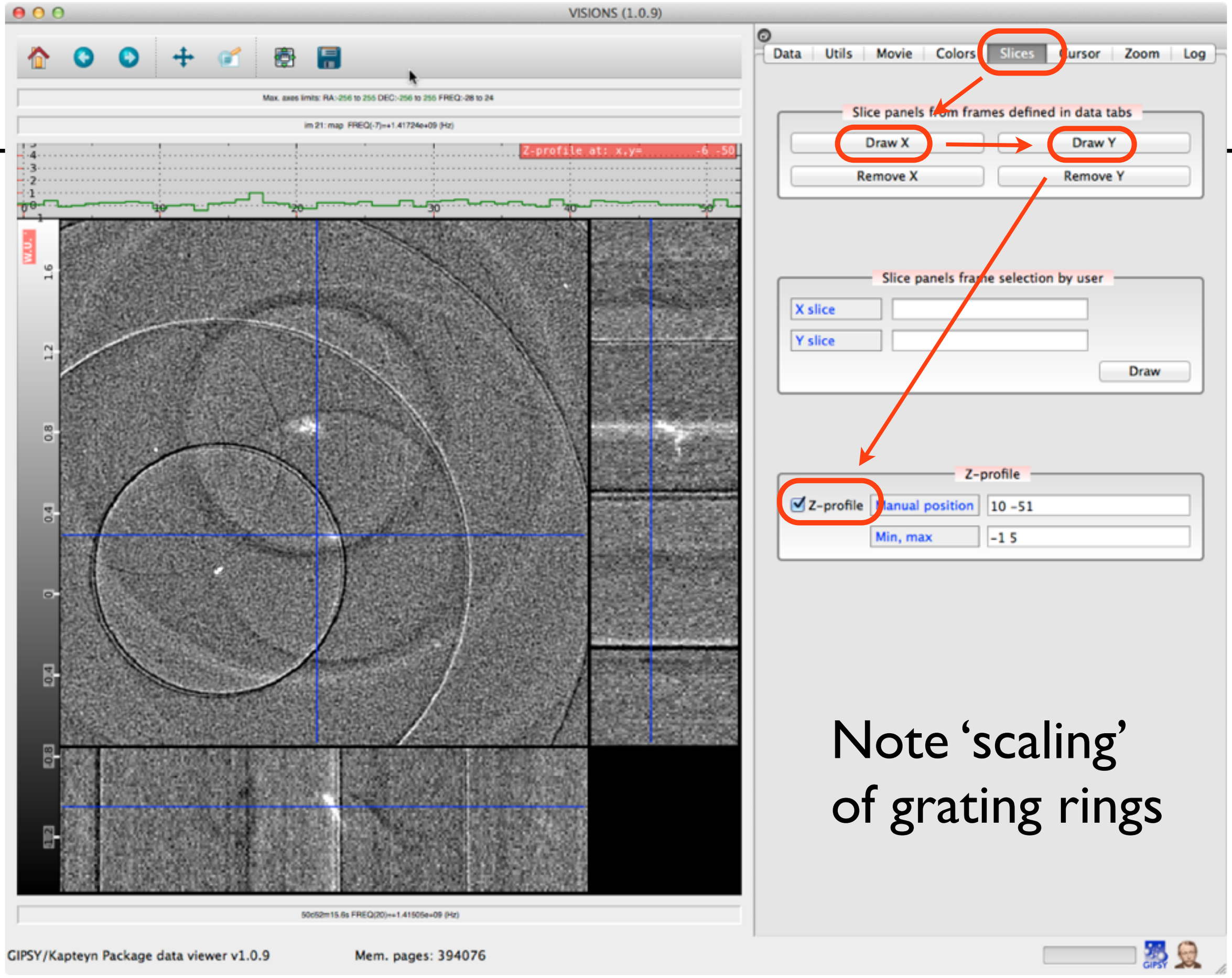
- Check GDS catalog

```
disk
INSET = ↩
```



Inspect image and cubes with visions





Note ‘scaling’
of grating rings

Inspect the headers

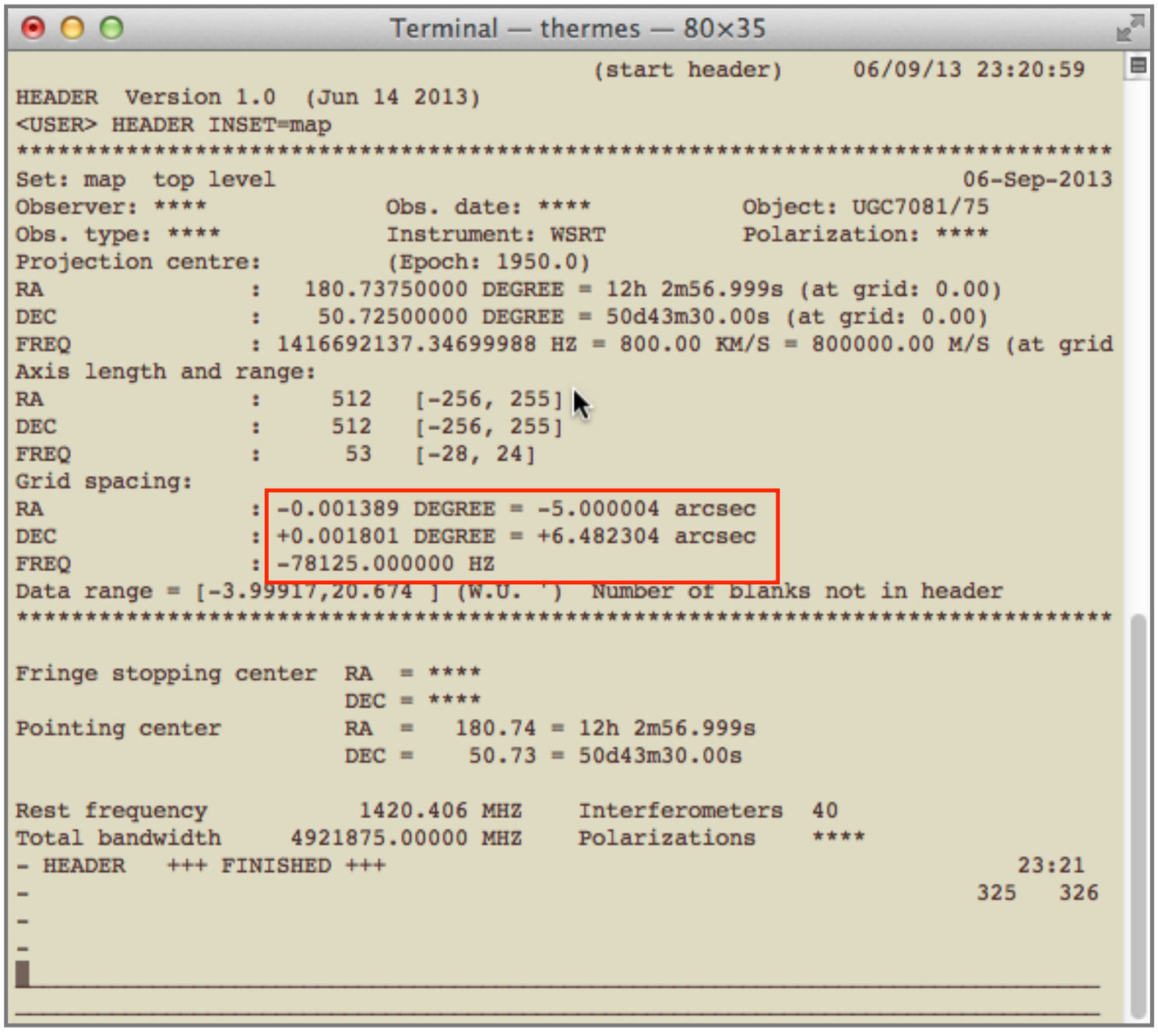
- Check data structure

```
header
INSET = map
```

map :	RA	:	512	[-256, 255]
	DEC	:	512	[-256, 255]
	FREQ	:	53	[-28, 24]
ap :	RA	:	1024	[-512, 511]
	DEC	:	1024	[-512, 511]
	FREQ	:	5	[-2, 2]

Center of cube at (0,0,0)

N4088.p2b	RA	:	3564	[-3592, -29]
	DEC	:	3571	[721, 4291]



- Full header listing:

```
fixhed
INSET = map
ITEM = ↩
```

Determine beam size and assign to channel maps

- measure FWHM of dirty beam

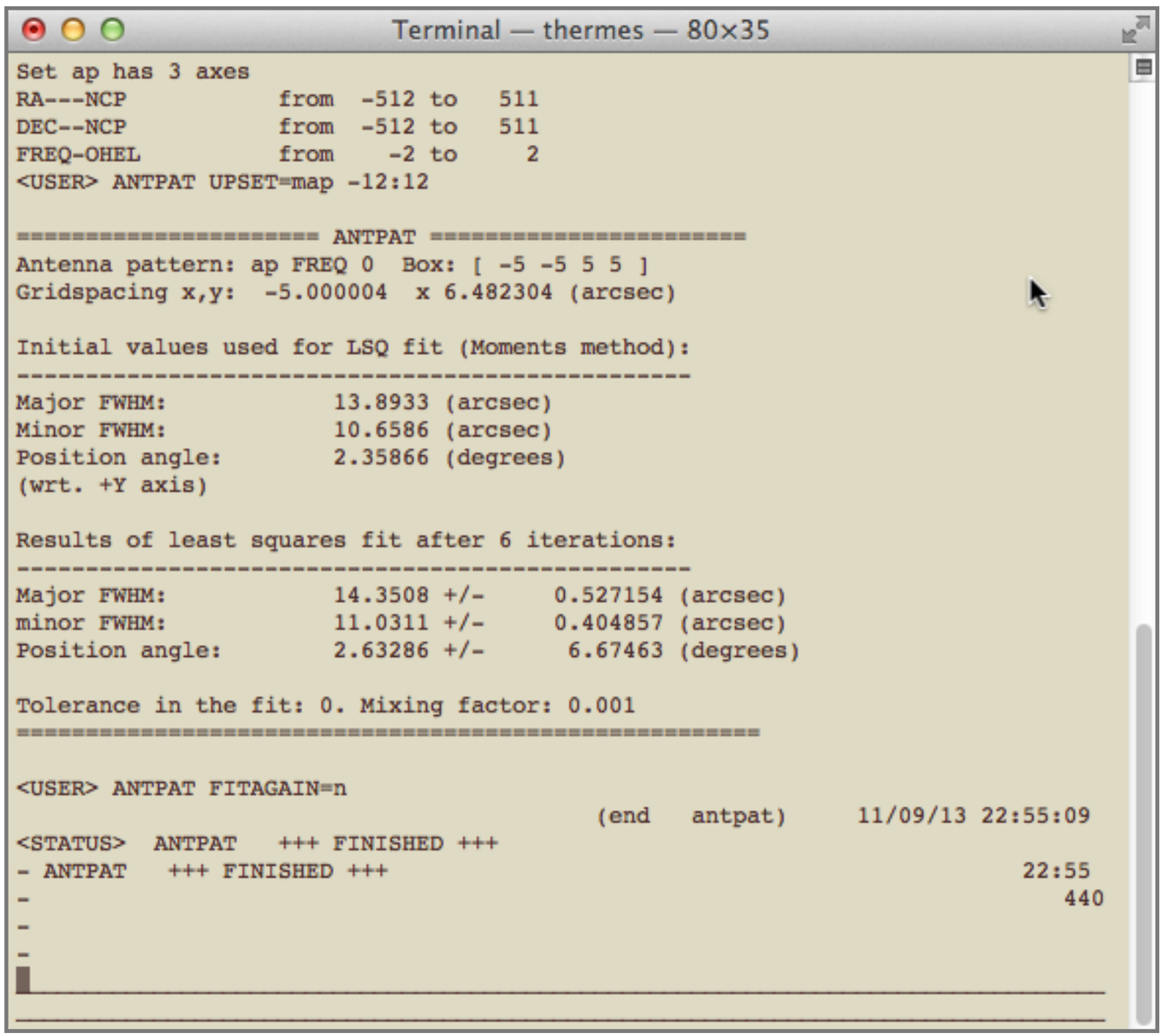
```
antpat box =-5 -5 5 5
APSET = ap 0
UPSET = map f -6:6
FITAGAIN = no
```

Do the same for the other
4 antenna patterns:

```
APSET = ap -2
UPSET = map f -28:-20

APSET = ap -1
UPSET = map f -19:-7

APSET = ap 1
UPSET = map f 7:19

APSET = ap 2
UPSET = map f 20:24
```

Check updated header:

```
header
INSET = map f 9
```



What is your mJy→K conversion factor?

Determine the line-free channels

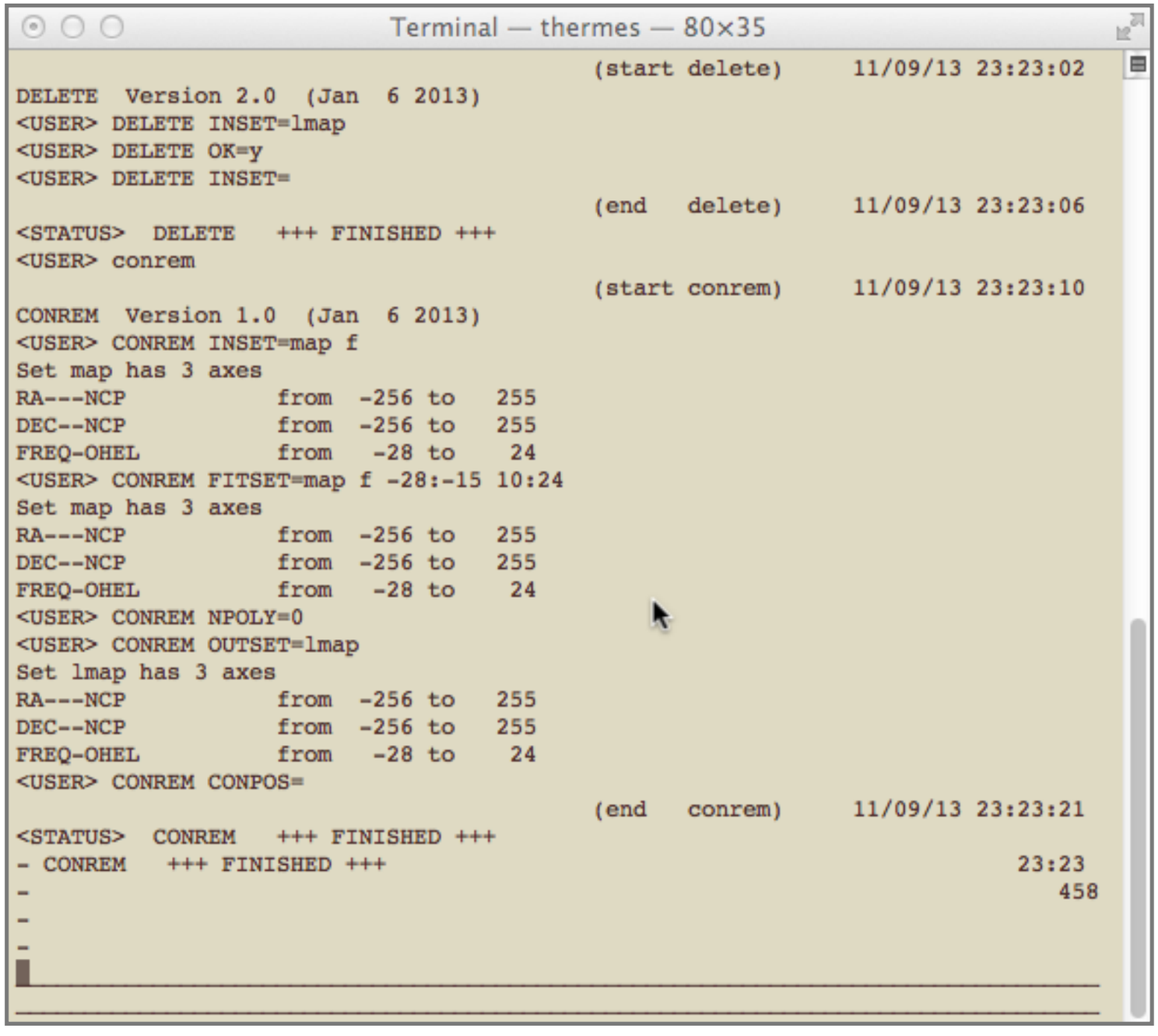
- inspect the cube with visions

  → line free channels: −28 : −15 and 10 : 24

- subtract the continuum

```
conrem
INSET = map f -28:24
FITSET = map f -28:-15 10:24
NPOLY = 1
OUTSET = lmap
CONPOS = 0
CONSET = cont
```

Check the continuum-subtracted cube with `visions` and confirm the line-free channels.
Check the continuum image too.



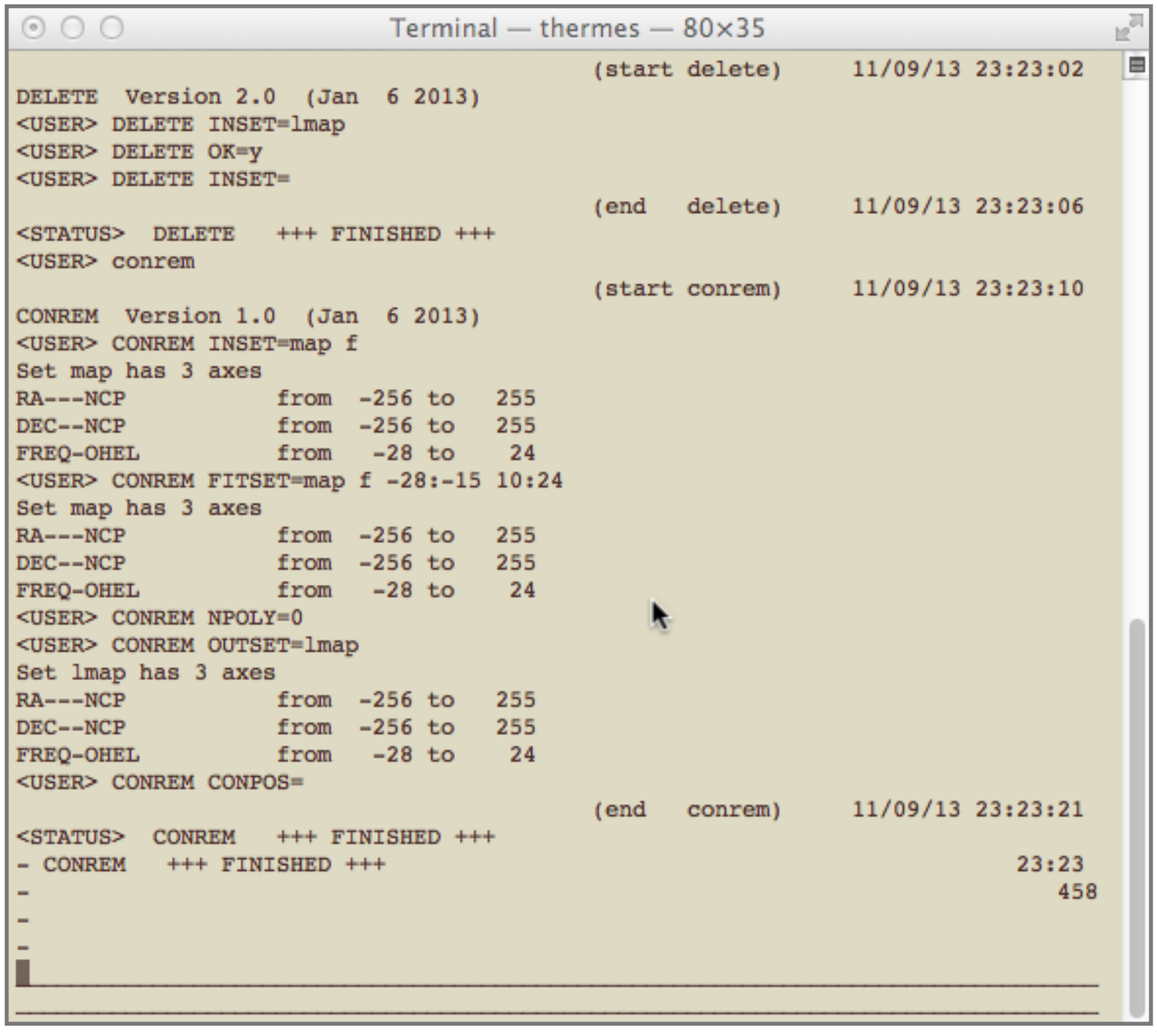
Smooth the channel maps

- smooth to a beam of 30” to enhance N(HI) sensitivity

```
smooth
INSET = lmap f -28:24
BOX = ↩
AUTO = no
OLDBEAM = 14.35 11.03
OLDPOSANG = 2.63
NEWBEAM = 30.0 30.0
NEWPOSANG = 0.0
DECIM = 1 1
OUTSET = lmap.bs30
SCALE = ↩
```

Check the smoothed cube with `visions`: add an extra data-tab.

What is your brightness temperature sensitivity?



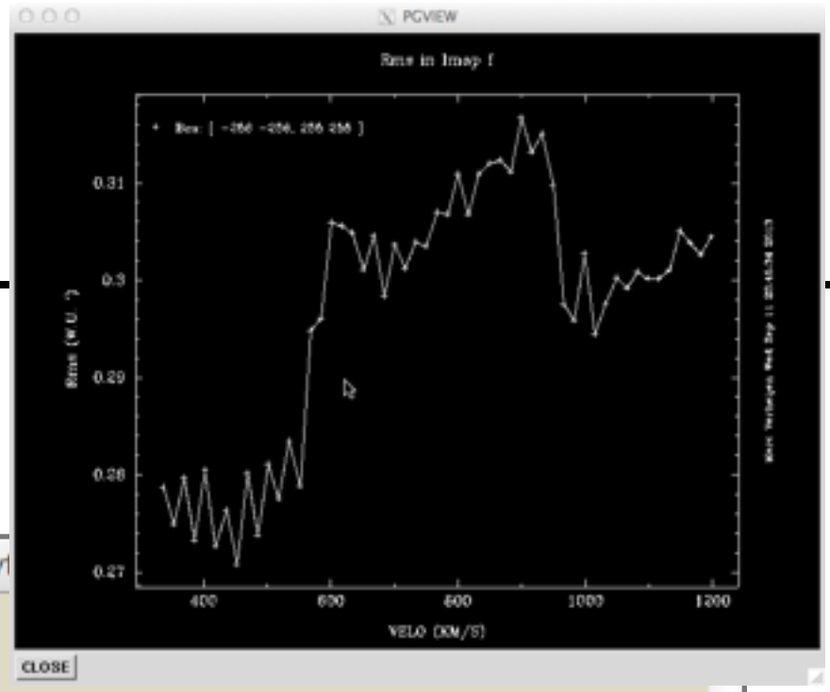
Inspect the noise levels

- determine statistics on channel maps

```
stat
INSET = lmap f
BOX1 = ↩
TOSCREEN = yes
FILENAME = ↩
FORMAT = fff.ffffff
PLOT = 3
GRDEVICE = screen
PLOT = 0
```

Do the same for the beam-smoothed cube and the continuum map, with an appropriate box.

Do the numbers make sense?



Terminal — pyt

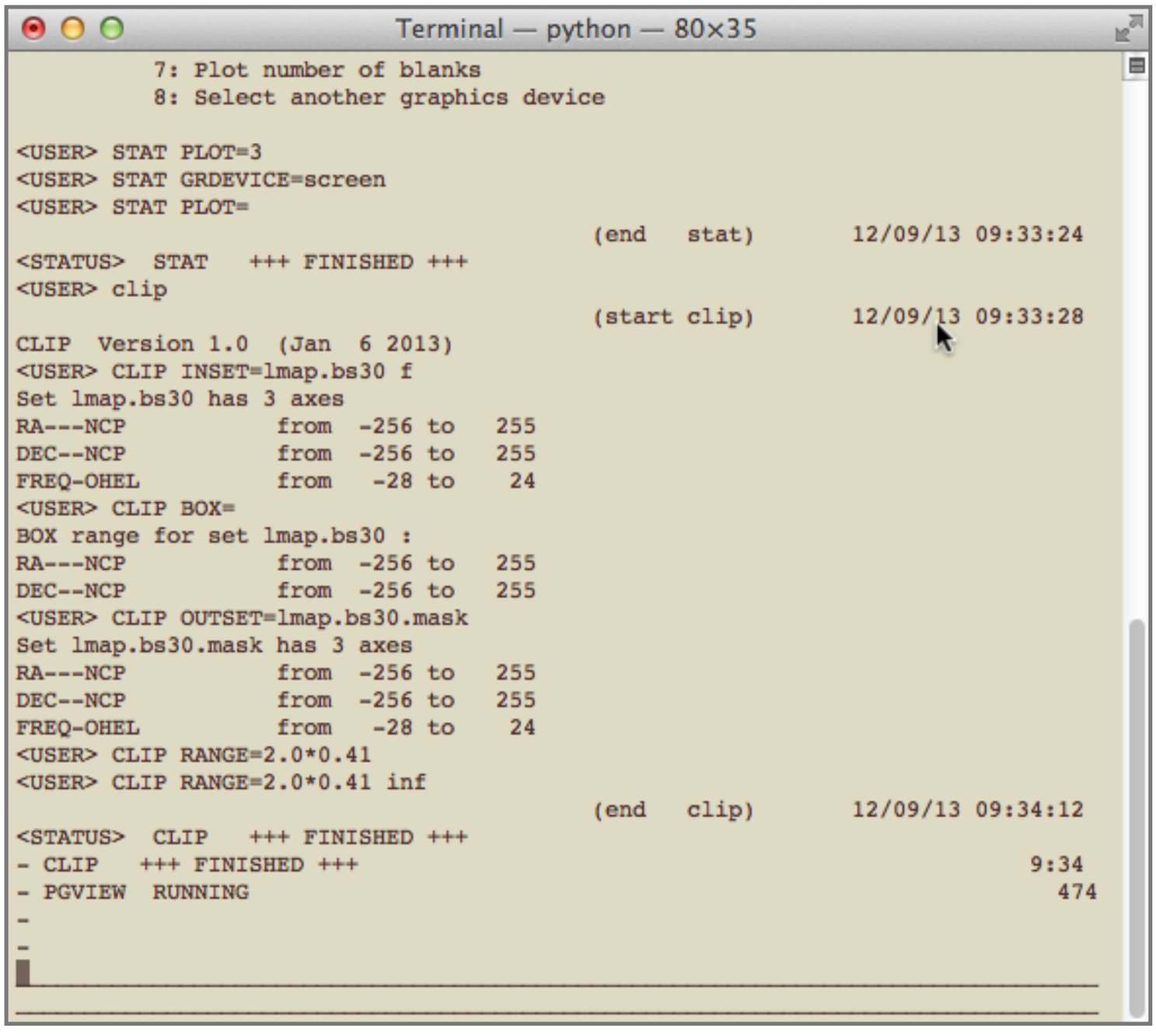
```
Statistics on columns:
column ->
tot. stats.:
          Sum:           *   0.012356  15.743275 -71.865203  99.482511
         Mean:  61.116203   0.000233   0.297043  -1.355947   1.877029
          Rms: 103.918398   0.000396   0.013059   0.096677   0.656167
          Min:           *  -0.000477   0.270757  -1.567011   1.157792
          Max: 297.058457   0.001133   0.316716  -1.186568   3.245931


        ====== PLOT OPTIONS =====
        0: Quit this plot loop
        1: Plot sum (default)
        2: Plot mean
        3: Plot rms
        4: Plot minimum values
        5: Plot maximum values
        6: Plot number of points without blanks
        7: Plot number of blanks
        8: Select another graphics device

<USER> STAT PLOT=3
<USER> STAT GRDEVICE=screen
<USER> PGVIEW
                                              (start pgview)     11/09/13 23:45:25
Warning: Unable to load any usable fontset
<USER> STAT PLOT=0
                                              (end   stat)       11/09/13 23:45:34
<STATUS>  STAT   +++ FINISHED +++
- STAT   +++ FINISHED +++                                                  23:48
- PGVIEW  RUNNING                                                           467
-
-
```

Define the `clean` areas - I

- Clip the smoothed map at 2-sigma

```
clip
INSET = lmap.bs30 f
BOX = ↩
OUTSET = lmap.bs30.mask
RANGE = 2.0*0.41 inf
```

Check the resulting cube with `visions`



Define the `clean` areas - 2

- Isolate the areas with HI emission

`pyblot`

- Select the input set :

  `FILE → Input set`

  `lmap.bs30.mask ↩`

  `→ OK`

- Adjust the contrast :

  `SLOPE / SHIFT`

- Define the output set :

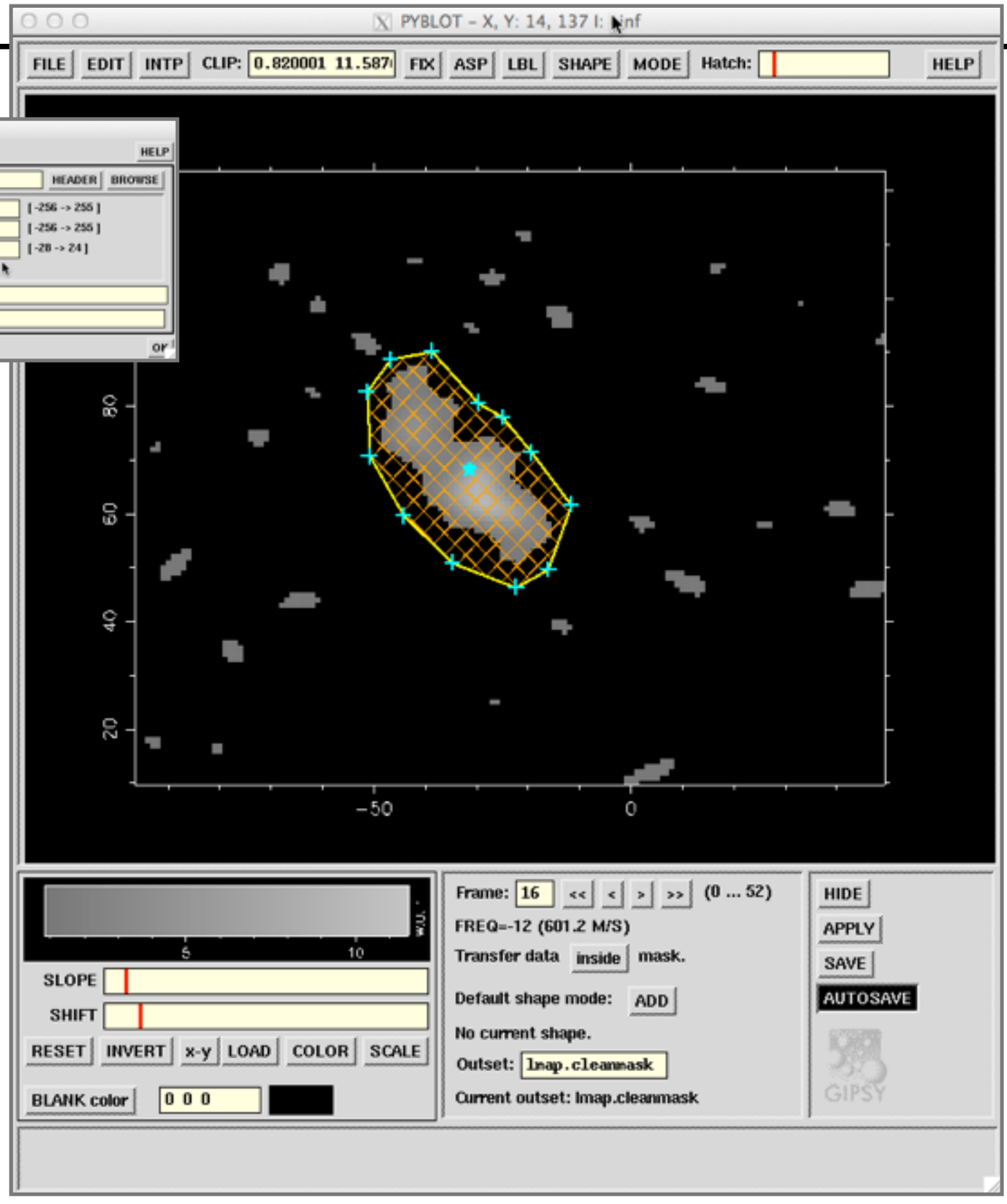
  `Outset: lmap.cleanmask ↩`

- Turn on `AUTOSAVE`
- Select a mask shape :

  `SHAPE → Polygon`

- Apply the masks to the outset :

  `EDIT → Apply all masks to outset`



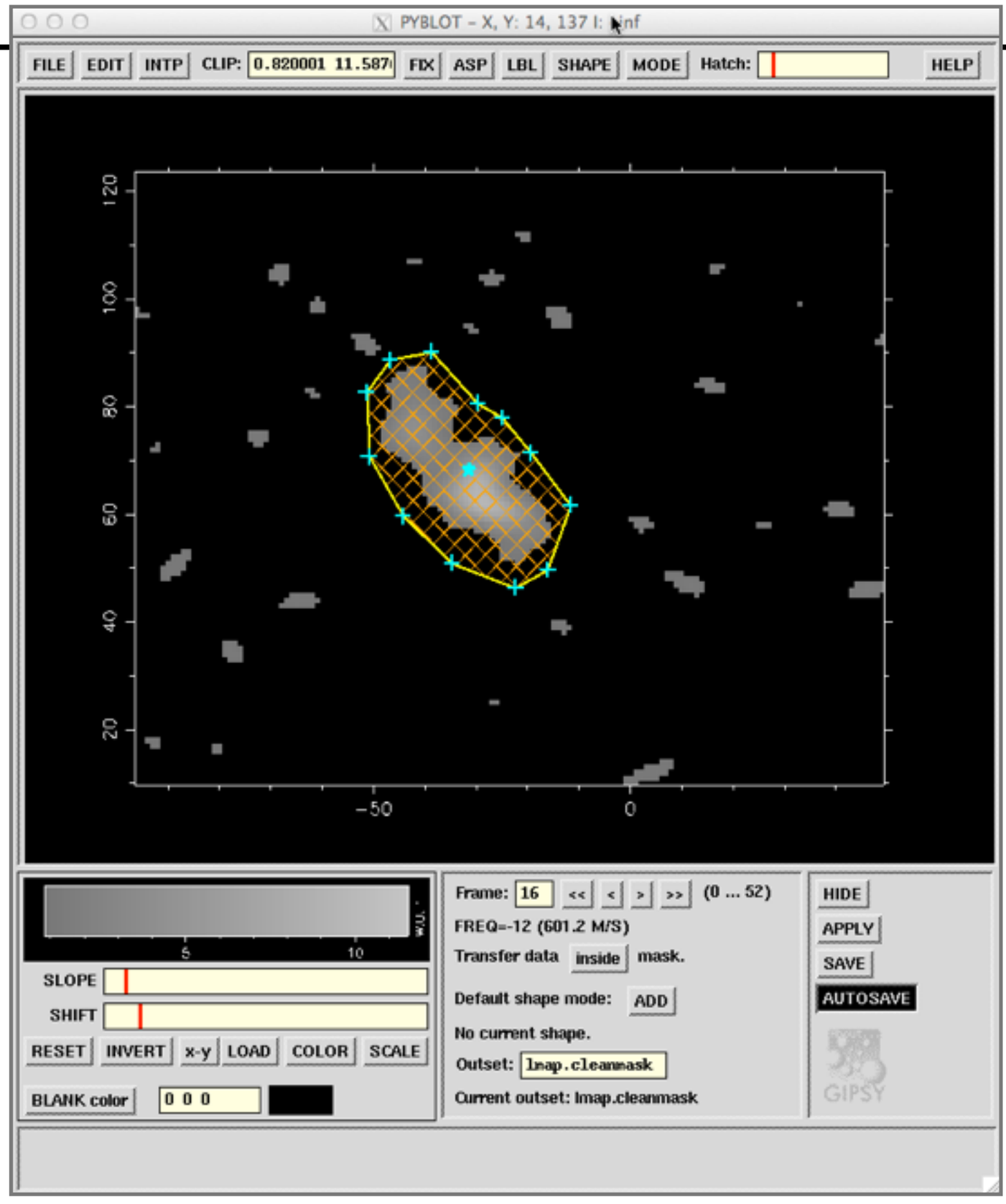
Define the `clean` areas - 2

- Isolate the areas with HI emission

`pyblot`

- Zoom on area of interest:
  `Shift + left mouse button`
- Define a polygon:
  `press P and use mouse`
  `(left: new point, right drag)`
- Step through the channel maps
  with `<` and `>` buttons
- When ready, press `APPLY`
  The masks will be stored in the input set and applied to the output set.
- Exit pyblot:
  `FILE → Exit`

Check the resulting cube with `visions`



Clean the channel maps

- Run clean

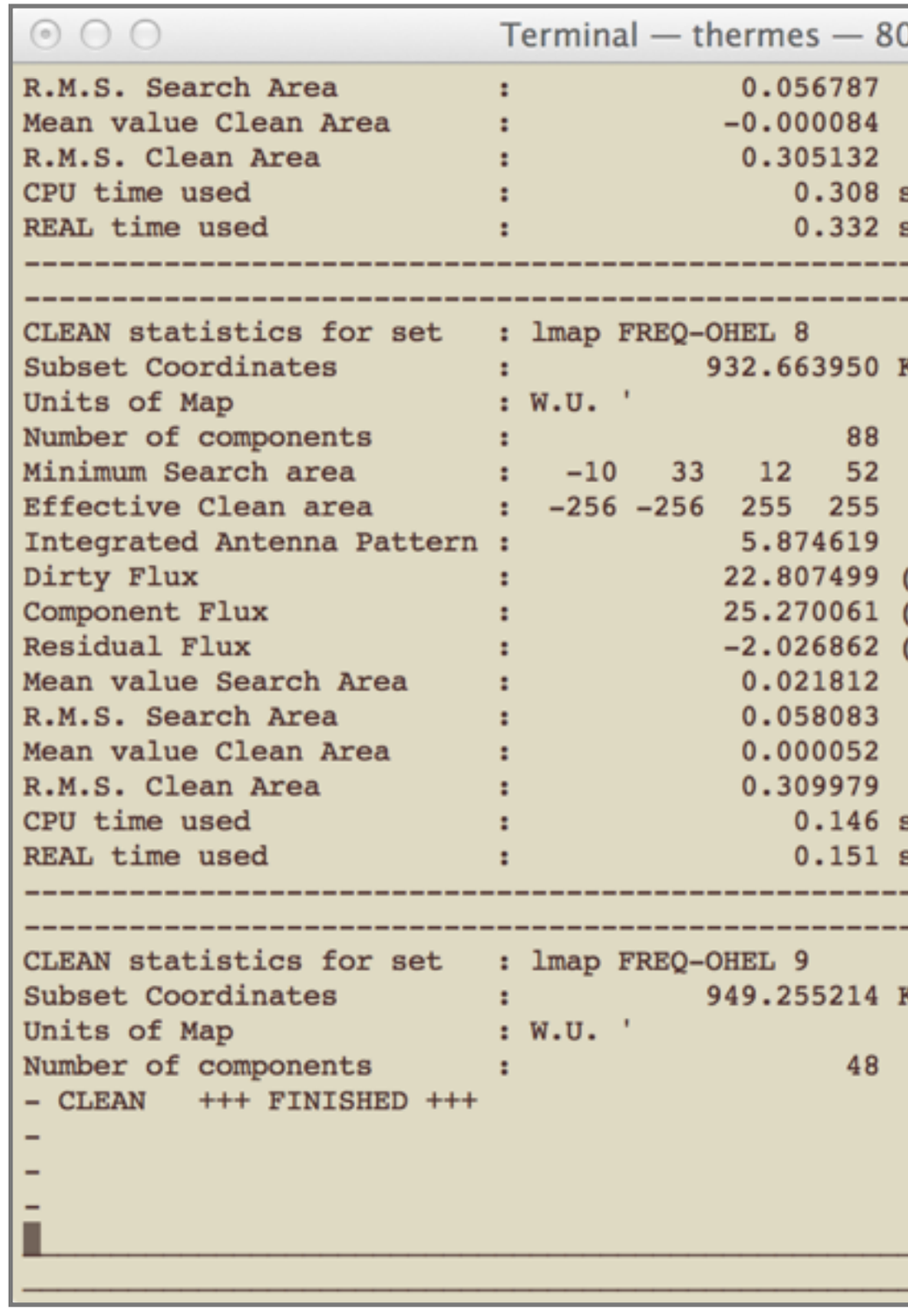
```
clean
INSET = lmap f -28:24
APSET = ap -2::9 -1::13 0::13 1::13 2::5
OUTSET = lmap.cl f -28:24
DEFSET = lmap.bs30.cleanmask f -28:24
CLEANBOX = ↩
NMAX = ↩
CUTOFF = 0.5*0.28
```

- Inspect the residual cube (`visions`)

- Restore the clean components

```
restore
INSET = lmap.cl f -28:24
BEAM = 14.0 11.0
BEAMPA = 0.0
```

- Inspect the residual cube (`visions`)



Cut out the areas with HI emission

- Conditionally transfer the data using the cleanmasks

```
condit
INSET = lmap.cl f -28:24
BOX = ↩
MASKSET = lmap.bs30.cleanmask f -28:24
OUTSET = N4088.HIcube
RGMODE = T
RANGE = -inf inf
BLMODE = B
BLOCKVAL = 0.0
```

- Inspect the outset cube (`visions`)

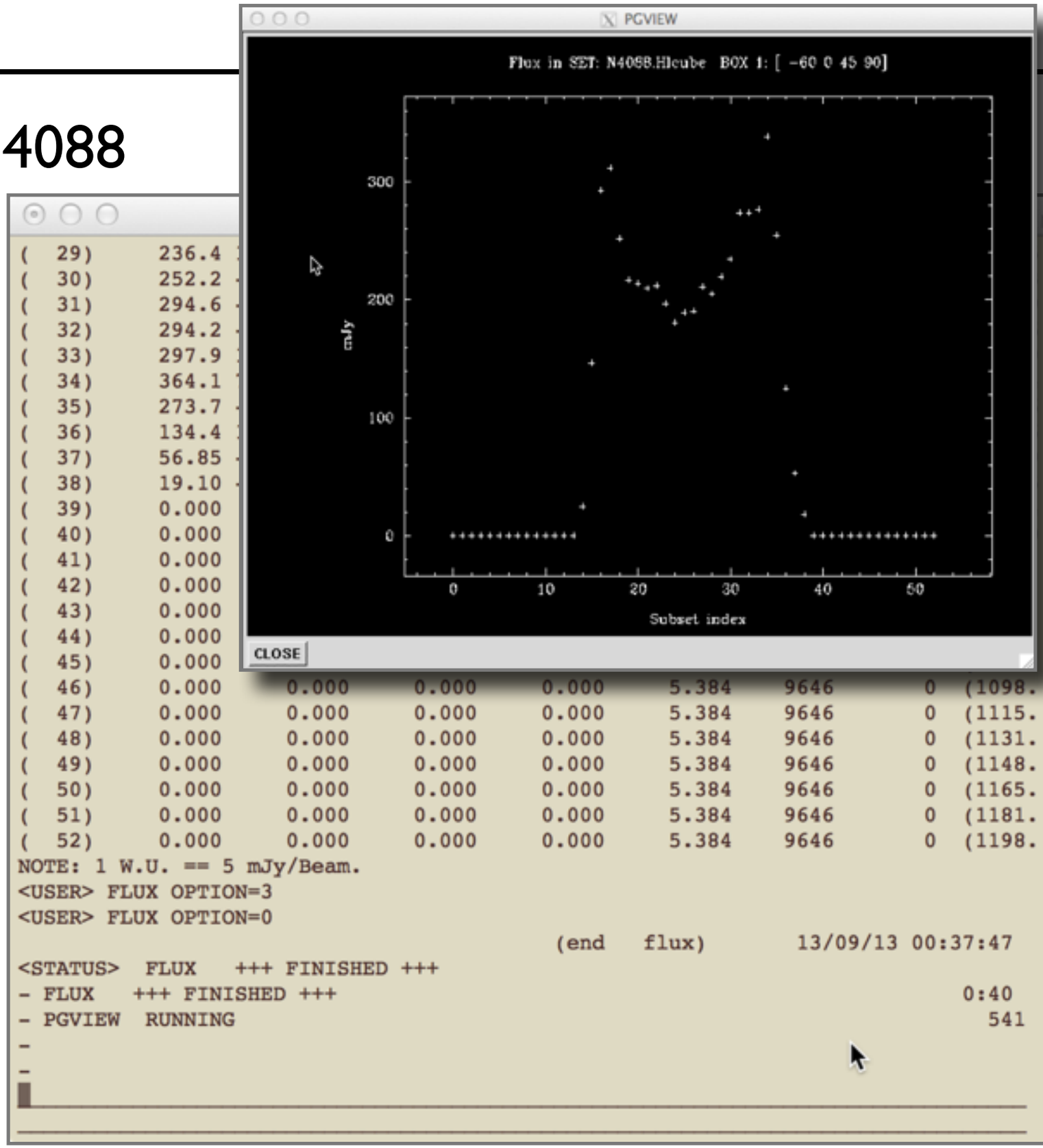
```
Terminal — python — 80×
CONDIT  Version 1.0  (Jan  6 2013)
<USER> CONDIT INSET=lmap.cl f -28:24
Set lmap.cl has 3 axes
RA---NCP            from  -256 to   255
DEC--NCP            from  -256 to   255
FREQ-OHEL           from   -28 to    24
<USER> CONDIT BOX=
BOX range for set lmap.cl :
RA---NCP            from  -256 to   255
DEC--NCP            from  -256 to   255
<USER> CONDIT MASKSET=lmap.bs30.cleanmask f -28:24
Set lmap.bs30.cleanmask has 3 axes
RA---NCP            from  -256 to   255
DEC--NCP            from  -256 to   255
FREQ-OHEL           from   -28 to    24
BOX range for set lmap.bs30.cleanmask :
RA---NCP            from  -256 to   255
DEC--NCP            from  -256 to   255
<USER> CONDIT OUTSET=N4088.HIcube
Set N4088.HIcube has 3 axes
RA---NCP            from  -256 to   255
DEC--NCP            from  -256 to   255
FREQ-OHEL           from   -28 to    24
<USER> CONDIT RGMODE=T
<USER> CONDIT RANGE=
<USER> CONDIT BLMODE=B
<USER> CONDIT BLOCKVAL=0.0
                                        (end   co
<STATUS>  CONDIT   +++ FINISHED +++
- CONDIT   +++ FINISHED +++
- PGVIEW  RUNNING
-
-
```

Calculate the HI global profiles

- Calculate the flux of N4088 in each channel

```
flux
INSET = N4088.HIcube f
BOX1 = -60 0 45 90
BOX2 = ↵
CLEANBEAM = y
BEAM = 14.0 11.0
GRDEVICE = screen
FILENAME = N4088.gp.tbl
OPTION = 3
OPTION = 0
```

Repeat this for a box that encloses N4085.

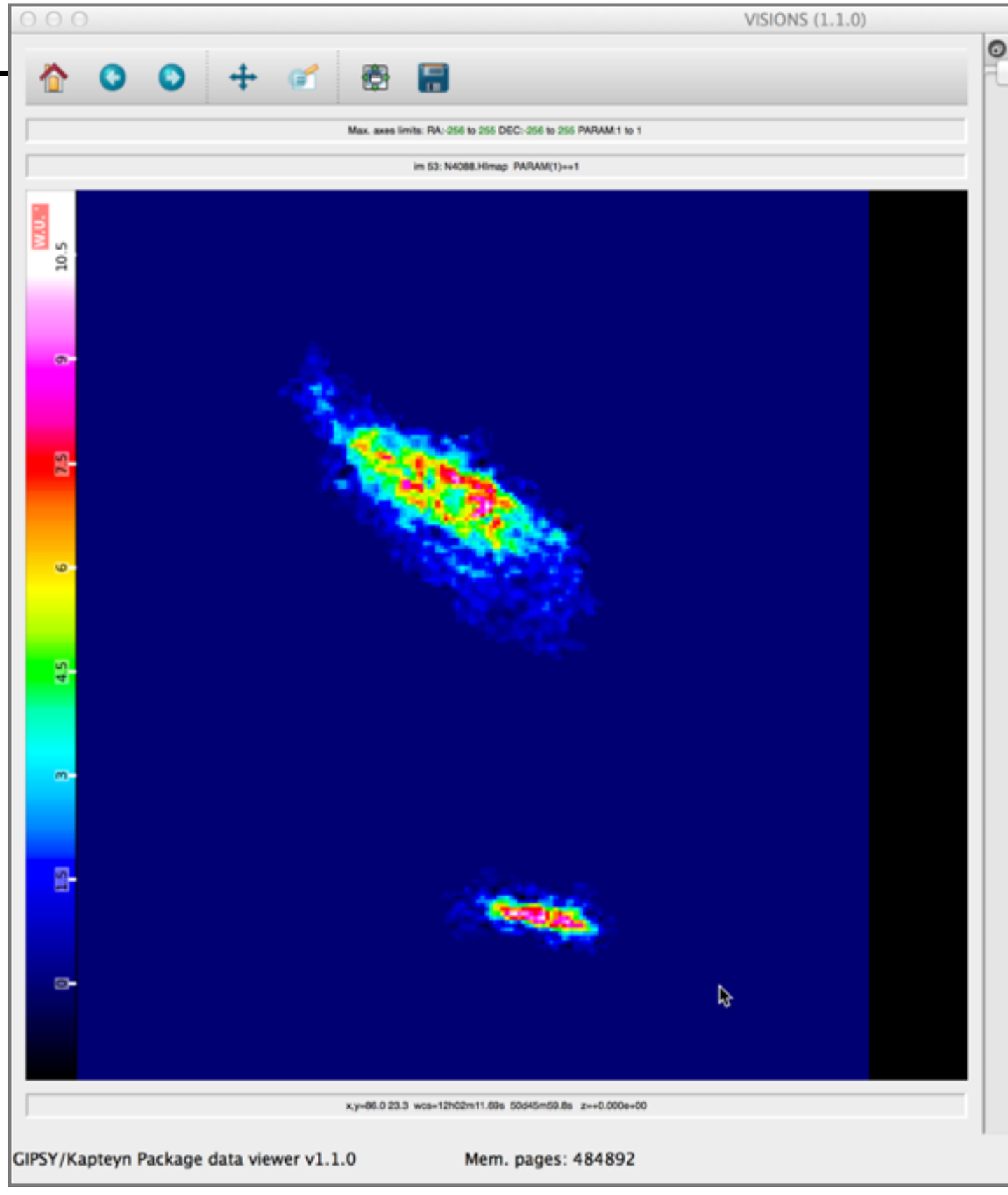


Make the total HI map

- sum all the channels in the conditionally transferred cube

```
sum
INSET = N4088.HIcube f
BOX = ↩
CUT = ↩
OUTSET = N4088.HImap
```

- Inspect the HI map with `visions`



Measure the total HI mass

- Calculate the total flux in the HI map

```
flux
INSET = N4088.HImap f 0
BOX1 = -60 0 45 90
BOX2 = ↩
CLEANBEAM = y
BEAM = 14.0 11.0
GRDEVICE = null
FILENAME = ↩
OPTION = 0
```

channel width [km/s]

$\int S_\nu dv = 5.118 \times 15.0 = 76.8$ [Jy km/s]

D = 18.6 [Mpc]

$M_{HI} = 2.36 \times 10^5 \times D^2 \times \int S_\nu dv$

$= 6.3 \times 10^9$ [M_{sun}]

```
Terminal — python — 80×35
     -plots the results
     -writes data to COLUMNS in a TABLE (TABNAME=)
     -can use polygons instead of boxes (hidden keyword PO

================================================================
<USER> FLUX INSET=N4088.HImap p 1
Set N4088.HImap has 3 axes
RA---NCP            from  -256 to   255
DEC--NCP            from  -256 to   255
PARAM-SUM           from     1 to     1
PBC:  Used frequency:0.141669D+01 GHz
FLUX: Data is identified as radio data
<USER> FLUX BOX1=-60 0 45 90
<USER> FLUX BOX2=
<USER> FLUX CLEANBEAM=y
<USER> FLUX BEAM=14.0 11.0
<USER> FLUX GRDEVICE=null
<USER> FLUX FILENAME=

SET: N4088.HImap  BOX 1: [ -60 0 45 90]
========================================
subset       sum    sumPBC      flux   fluxPBC      sumAP
(  # )    W.U. '    W.U. '       mJy       mJy          *
================================================================
(   0)     5510. 2.933e+20     5118. 2.724e+20      5.384
NOTE: 1 W.U. == 5 mJy/Beam.
<USER> FLUX OPTION=0
                                                (end   flux)
<STATUS>  FLUX   +++ FINISHED +++
- FLUX   +++ FINISHED +++
- PGVIEW  RUNNING
-
-
```

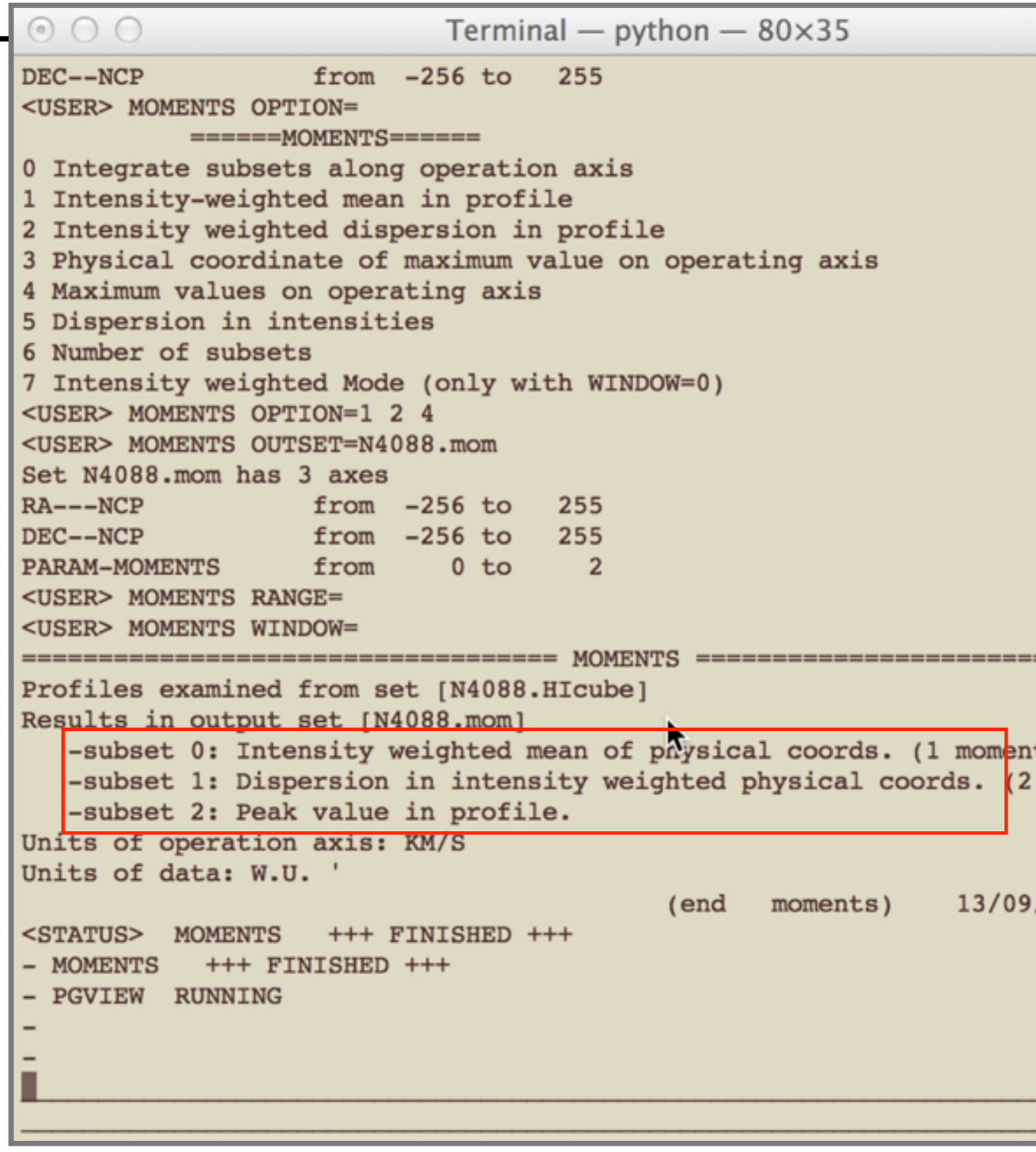
Construct a velocity field - 1

- Use moments to make initial estimates

```
moments
INSET = N4088.HIcube f
BOX = ↩
OPTION = ↩
OPTION = 1 2 4
OUTSET = N4088.mom
RANGE = ↩
WINDOW = ↩
```

- re-order the subsets
  → run copy 3 times:

```
copy
INSET = N4088.mom p 2   0   1
BOX = ↩                 ↓   ↓   ↓
OUTSET = N4088.ini 1    2   3
```



Construct a velocity field - 2

- Make Gauss fits using the initial estimates

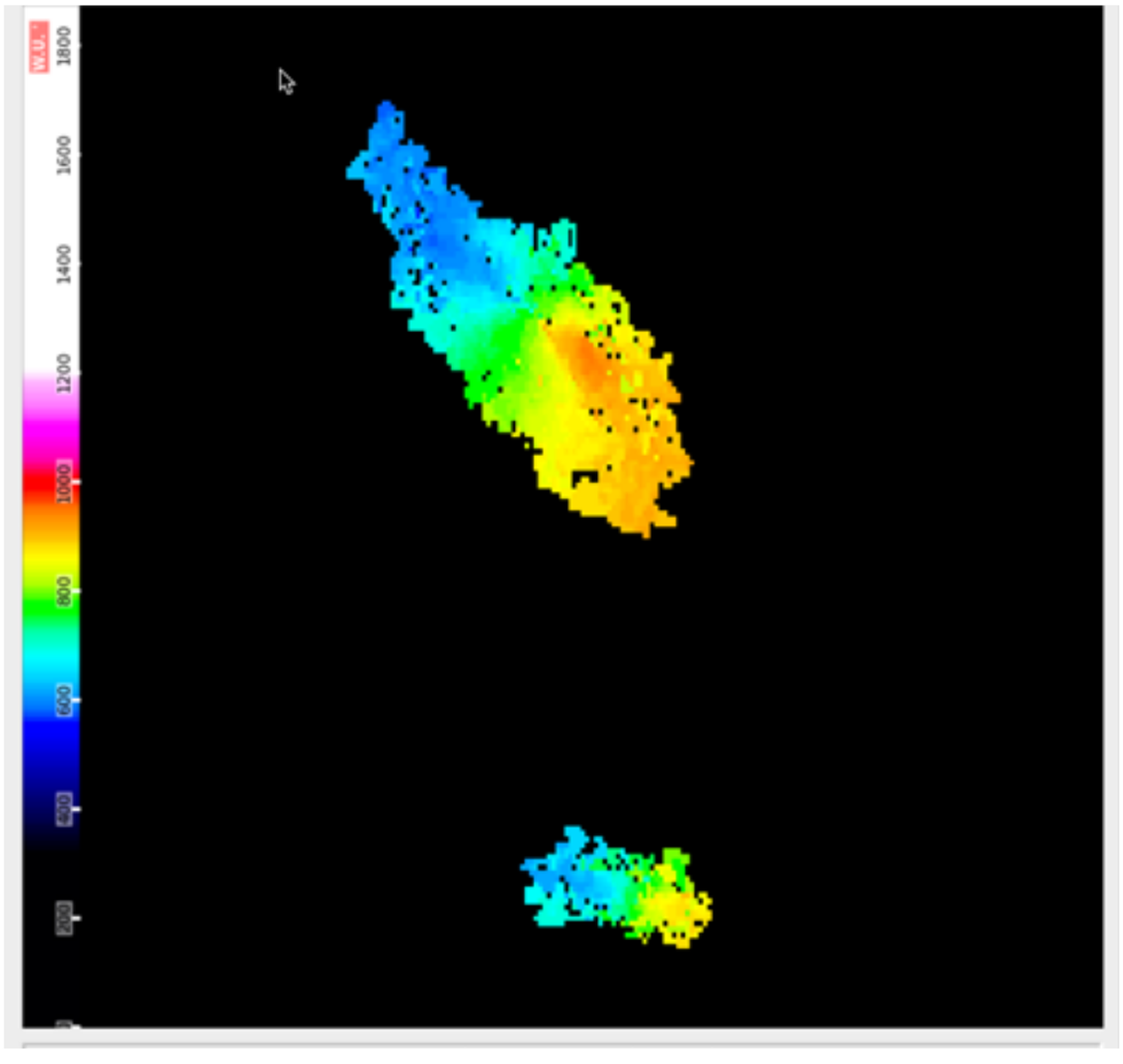
```
gaufit
STARTNEW = n
INSET = N4088.HIcube f
BOX = ↩
OPTION = ↩
OPTION = 0 1 2
NGAUSS = 1
OUTSET = N4088.velfi
ESTRMS = 0.28
NCORR = 1
Q = 2
CUTAMP = 3*0.28
CUTDISP = 5
TOLERANCE = ↩
```

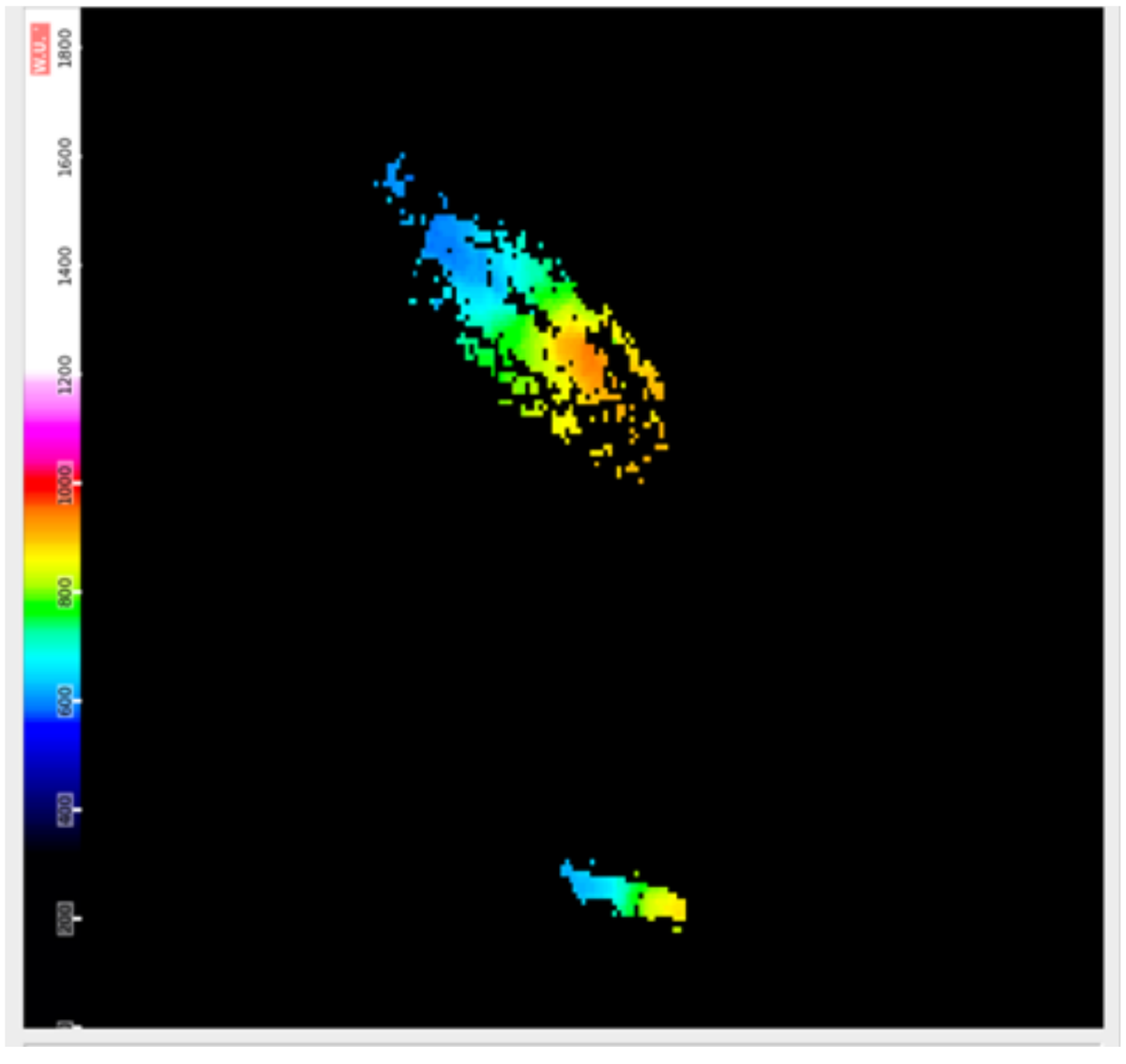
Terminal — python — 80×35

```
<USER> GAUFIT Q=2
<USER> GAUFIT CUTAMP=3*0.28
<USER> GAUFIT CUTDISP=5
<USER> GAUFIT TOLERANCE=

============================ RESULTS ========================
  Profiles from set:  N4088.HIcube
  Results in set:     N4088.velfi
  param 1: Estimated amplitudes of component 1
  param 2: Estimated centre of component 1
  param 3: Estimated dispersion of component 1
  param 4: fitted amplitudes of component 1
  param 5: fitted centre of component 1
  param 6: fitted dispersion of component 1
  param 7: Error in fitted amplitudes of component 1
  param 8: Error in fitted centre of component 1
  param 9: Error in fitted dispersion of component 1
  Max. number of gaussians in profile is 1
  Range operation axis: [336.595,1198.34] KM/S
  Fitted gaussians will have amplitude > 0.840000 W.U. '
  Fitted gaussians will have dispersion > 5.000000 KM/S
  One pixel on operation axis has (average) width 16.572090 KM/S
  Processed 262144 profiles in 0.67 sec (0.65 cpu sec)
  Number of profiles fitted with Q= 2:  2155
  For 259124 profiles no estimate was found.
  For 259989 profiles the fitting failed.
  367 times the max. no. iterations was exceeded.
                                          (end   gaufit)     13/09/13 0
<STATUS>  GAUFIT   +++ FINISHED +++
- GAUFIT   +++ FINISHED +++
- PGVIEW  RUNNING
-
-
```

Construct a velocity field - 3

MOMENTS → initial estimate → GAUFIT



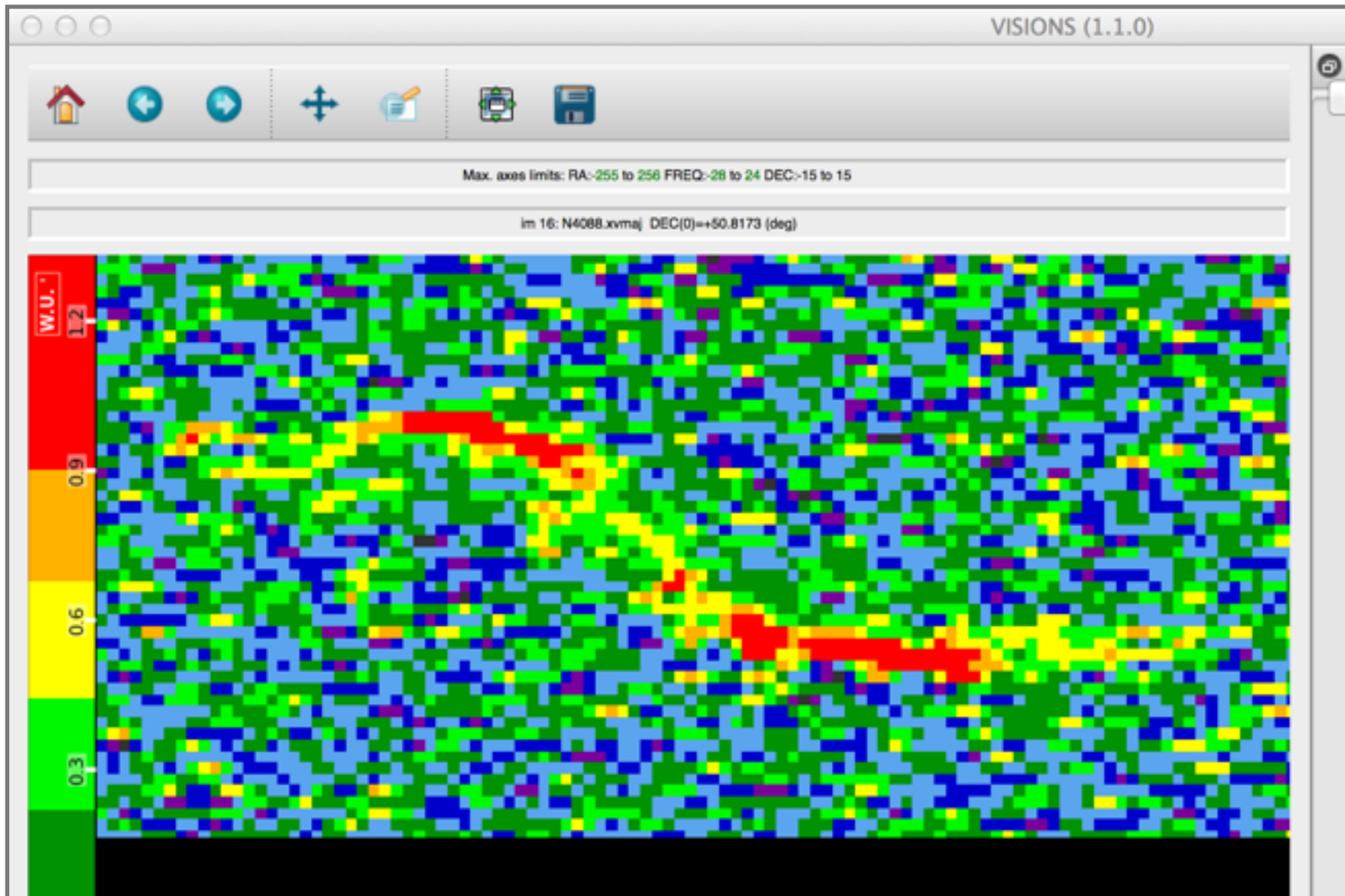


Extract a Position-Velocity diagram

- Cut along the kinematic major axis

```
slice
INSET = lmap.cl f
POSITION = * 12 03 01.97 * 50 49 02.3
ANGLE = 230
GRIDOUT = ↩
POINTS = ↩
SLICES = 15,15
SPACE = ↩
OUTSET = N4088.xvmaj
PLOT = n
```

Note that de ‘DEC-axis’ contains the 31 slices.



Extract a rotation curve

- Example of `rotcur` parameters :

```
rotcur
INSET = N4088.velfi p 5
BOX = -60 0 45 90
RADII = 5:205:10
WIDTHS = 10.0 ;
VSYS = 757.0 ;
VROT = 200.0 ;
VEXP = 0.0 ;
PA = 230.0 ;
INCL = 65.0 ;
CENTRE = * 12 03 01.97 * 50 49 02.3
FREEANGLE = 0.0
SIDE = both
WEIGHT = uniform
FIXED = VSYS XPOS YPOS VEXP
FILENAME = N4088.rotcur.tbl
```

Note that `rotcur` can be run multiple times, keeping different sets of parameters fixed.
For example:

Step 1 : fit Vsys, Center
Step 2 : fit PA, INC, VROT
Step 3 : fit INC, VROT
Step 4 : fit VROT