Radio Astronomy Software

presented by
Victor de Souza Magalhães

07/07/2016

Summary

- GILDAS
- AIPS
- CASA
- MIRIAD
- Python

IRAM - GILDAS

- GILDAS - Grenoble Image and Line Data Analysis Software
  - Core:
    - SIC; GreG; pySic
  - Calibration:
    - MIRA; CLIC; MRTCAL
  - Data reduction:
    - MAPPING; CLASS
  - Preparation for observations:
    - ASTRO

SIC: Simple interpreter of commands

- It is the core of GILDAS:
    - Command line interpreter
    - Parser
    - Scripting language.
        - syntax similar to FORTRAN.

```
@ window_tools
define logical gameover
define integer used[9] won
let used 0
let won 0
let gameover .false.
define integer pos
for i 1 to 9
   SIC\@ window_xy 3 3 'i'
   g\draw text -1 -1 'i' /box 9
   GREG1\BOX n n n
SIC\NEXT
@ window_xy 1 1 1
set expand 2
g\draw text 0 0.5 "GREG tic tac toe!" /box 8
set expand 1.0
pen /c 7
g\box n n n
define integer play
let play 0
for /while .not.gameover
   if nint(play/2).eq.(1.0*play/2.0) then
      say "Make your move O!"
```

-:--- tictactoe.greg 32% (64,0) (F90)

pySic

- It is the python binding to GILDAS.
- enables the python prompt inside GILDAS.
- Enables the use of python capabilities in GILDAS (functions, modules, plotting, list manipulation, etc).
- Enables the use of GILDAS functionalities in python (plotting, spectrum manipulation, data handling, etc).
- works through a python module called pySic.

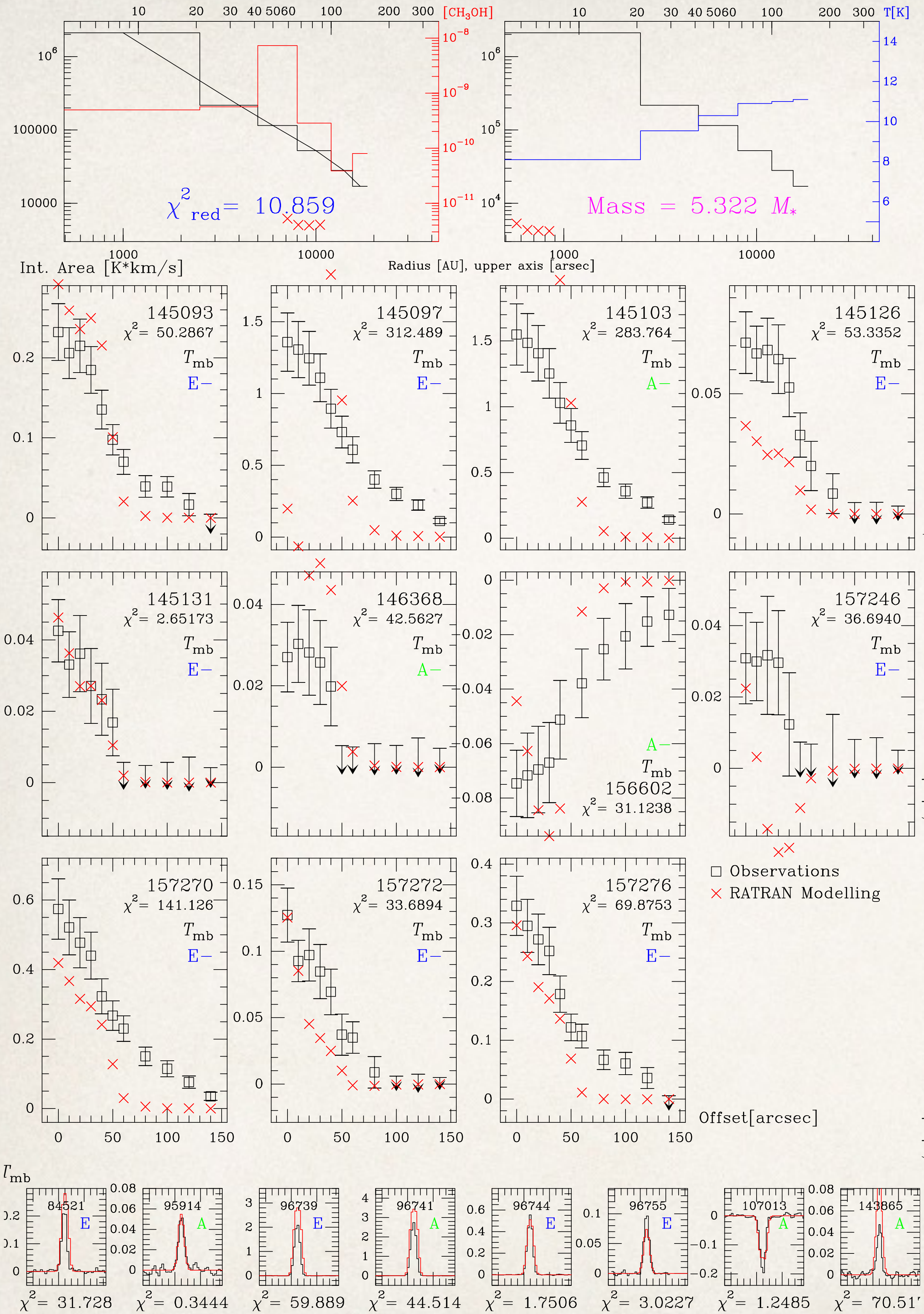
```
### Python modules ###

import pysic

pysic

# List the content of attribu
    and methods. See full
# documentation for details.
dir(pysic)

# .comm() method send a singl
    command line to the SIC
    interpretor
print pysic.comm.__doc__
pysic.comm('EXA PI')

# .enter() starts the SIC pro
pysic.enter()
# go back with the SIC\PYTHON
    command
SIC\PYTHON


### Sharing variables ###
```

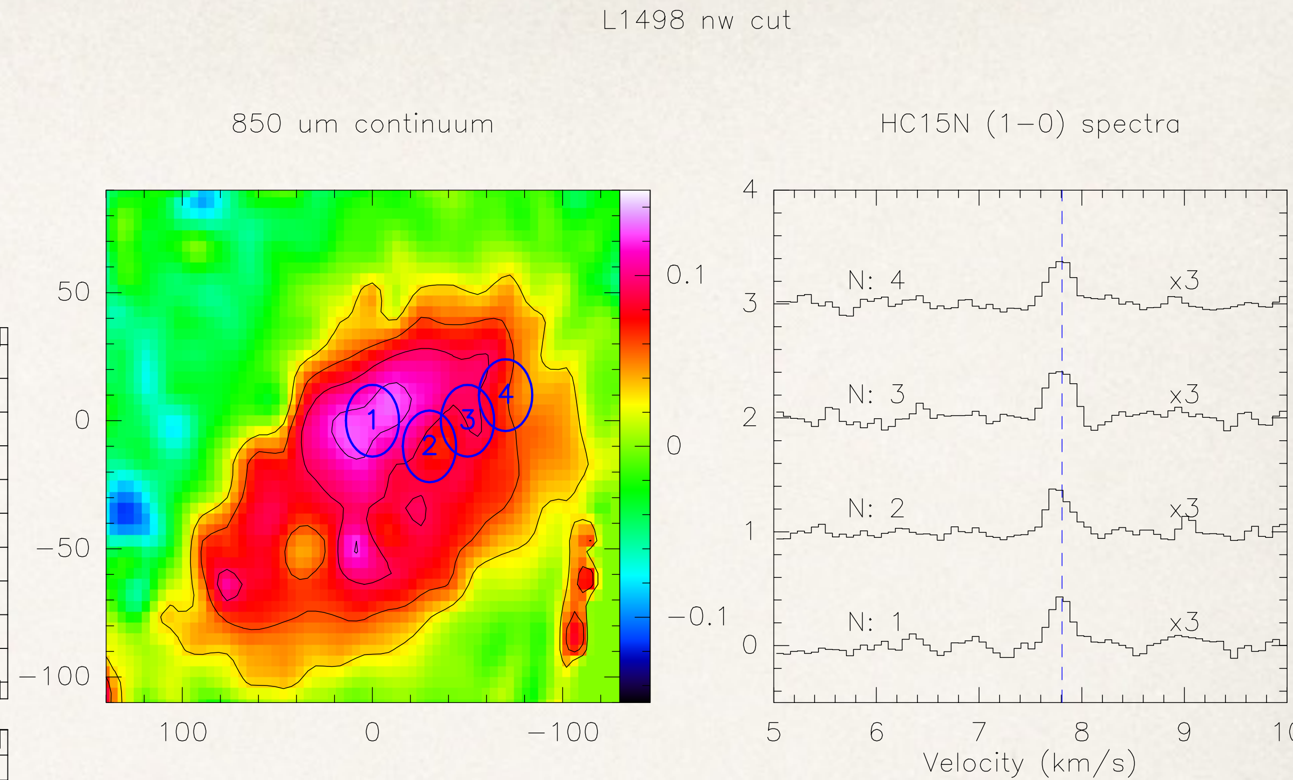
MIRA | MRTCAL | CLIC

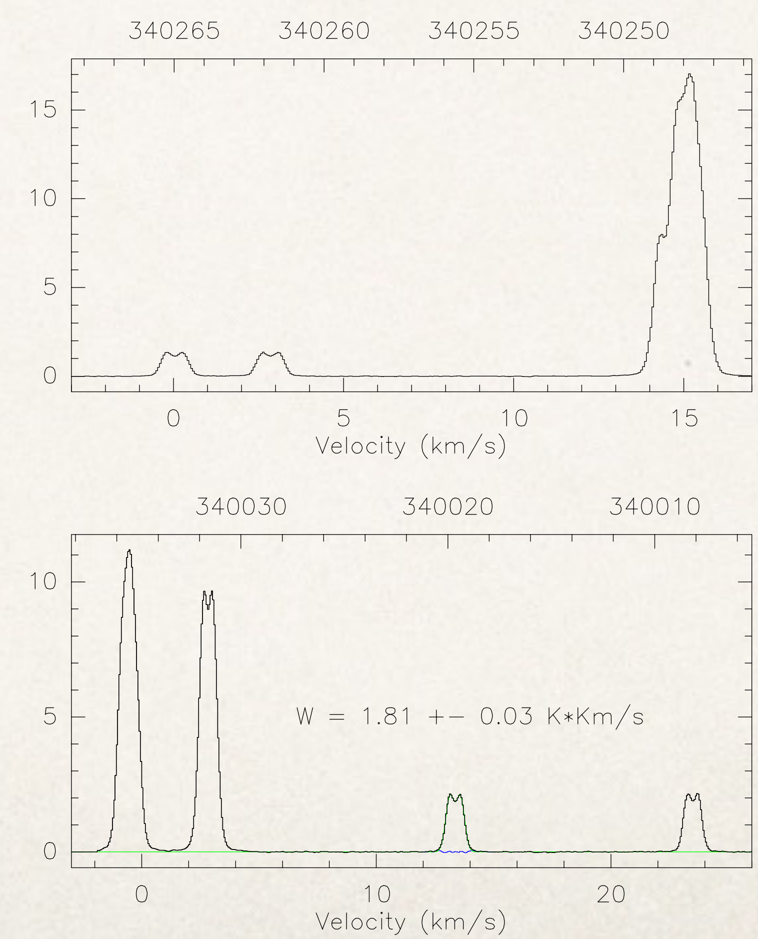
- MIRA is the software currently in use for the calibration of IRAM-30m data.
- MRTCAL is the software in development to replace MIRA, as faster, more reliable software.
- CLIC is the software for the calibration of IRAM-NOEMA data, it produces plots so you can follow what is happening during the pipelines.

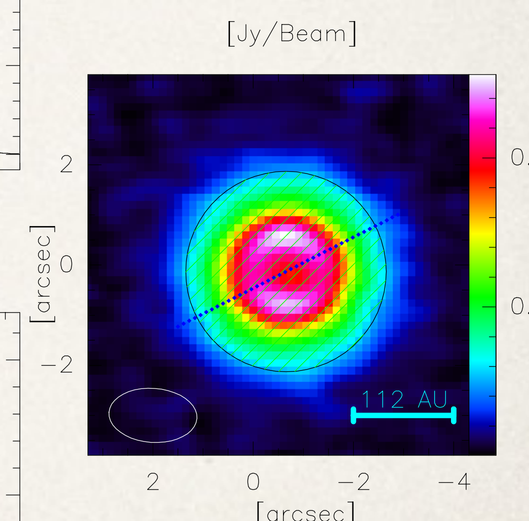
GreG - Grenoble Graphic

- It is a very powerful plot generator.
- It can be called from the other software inside GILDAS.
- Can produce very beautiful plots.
- Can even produce interactive graphics.


[CH3OH]
T[K]
χ² red = 10.859
Mass = 5.322 M*
Radius [AU], upper axis [arsec]
Int. Area [K*km/s]
145093 χ² = 50.2867 Tmb E−
145097 χ² = 312.489 Tmb E−
145103 χ² = 283.764 Tmb A−
145126 χ² = 53.3352 Tmb E−
145131 χ² = 2.65173 Tmb E−
146368 χ² = 42.5627 Tmb A−
156602 χ² = 31.1238 Tmb A−
157246 χ² = 36.6940 Tmb E−
157270 χ² = 141.126 Tmb E−
157272 χ² = 33.6894 Tmb E−
157276 χ² = 69.8753 Tmb E−
Observations
RATRAN Modelling
Offset[arcsec]
Tmb
84521 E χ² = 31.728
95914 A χ² = 0.3444
96739 E χ² = 59.889
96741 A χ² = 44.514
96744 E χ² = 1.7506
96755 E χ² = 3.0227
107013 A χ² = 1.2485
143865 A χ² = 70.51'

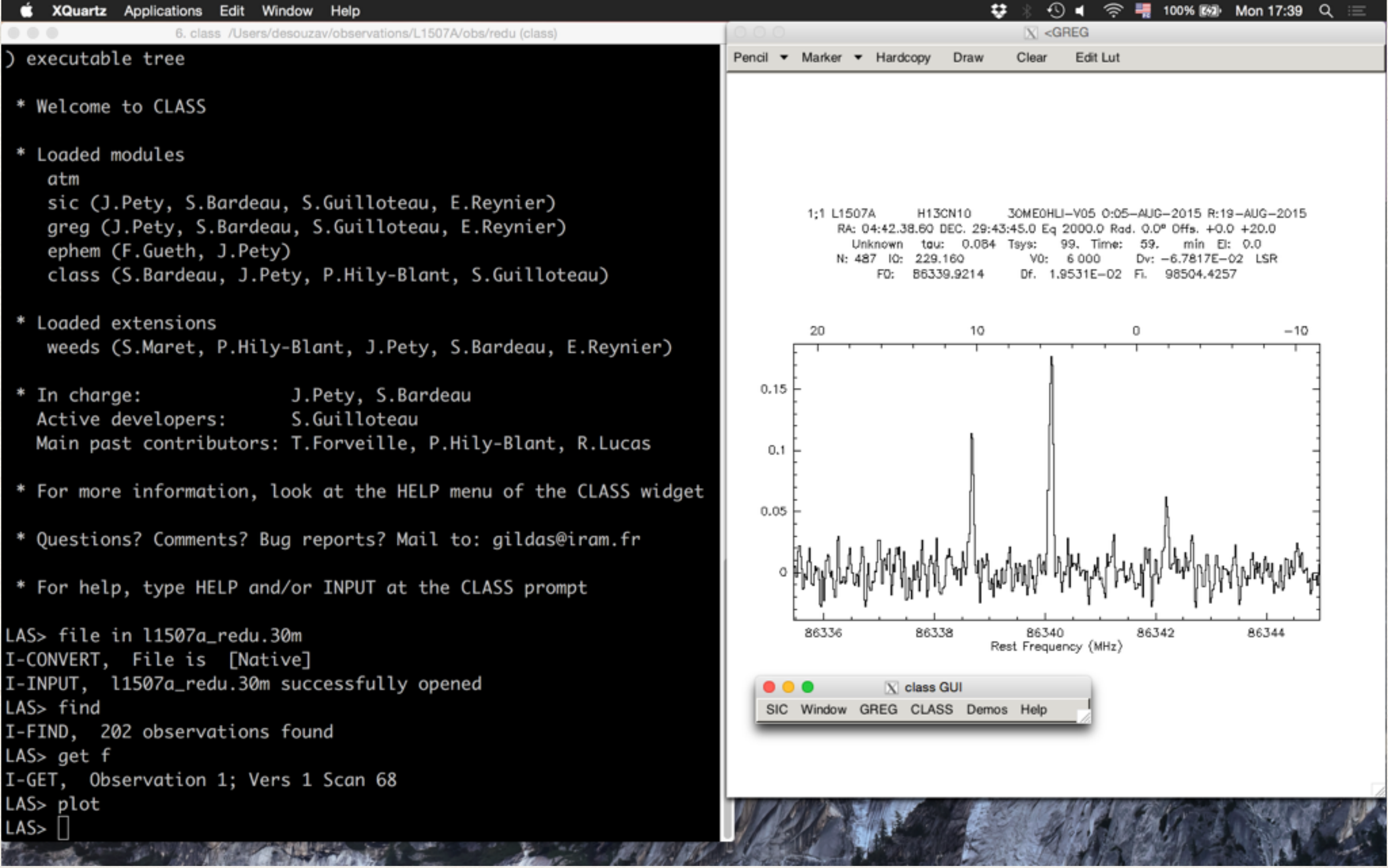

L1498 nw cut
850 um continuum
HC15N (1−0) spectra
N: 4 x3
N: 3 x3
N: 2 x3
N: 1 x3
Velocity (km/s)


340265 340260 340255 340250
Velocity (km/s)
340030 340020 340010
W = 1.81 +− 0.03 K*Km/s
Velocity (km/s)


[Jy/Beam]
[arcsec]
112 AU
[arcsec]

CLASS

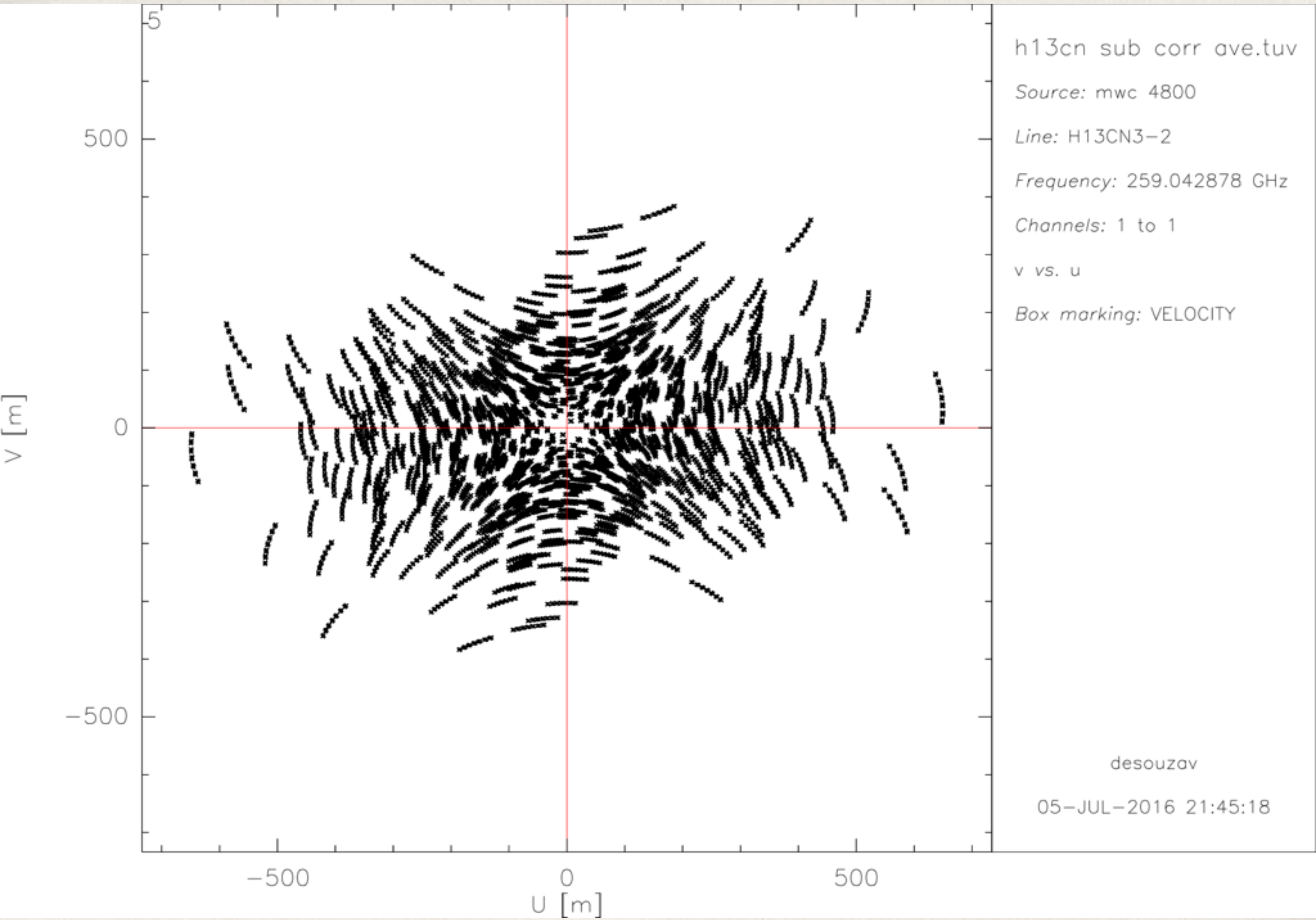
- It is a subset of GILDAS fine tuned for spectral line analysis.
- it contains tools for:
  - Line identification
  - Line fitting
  - visualisation of spectra



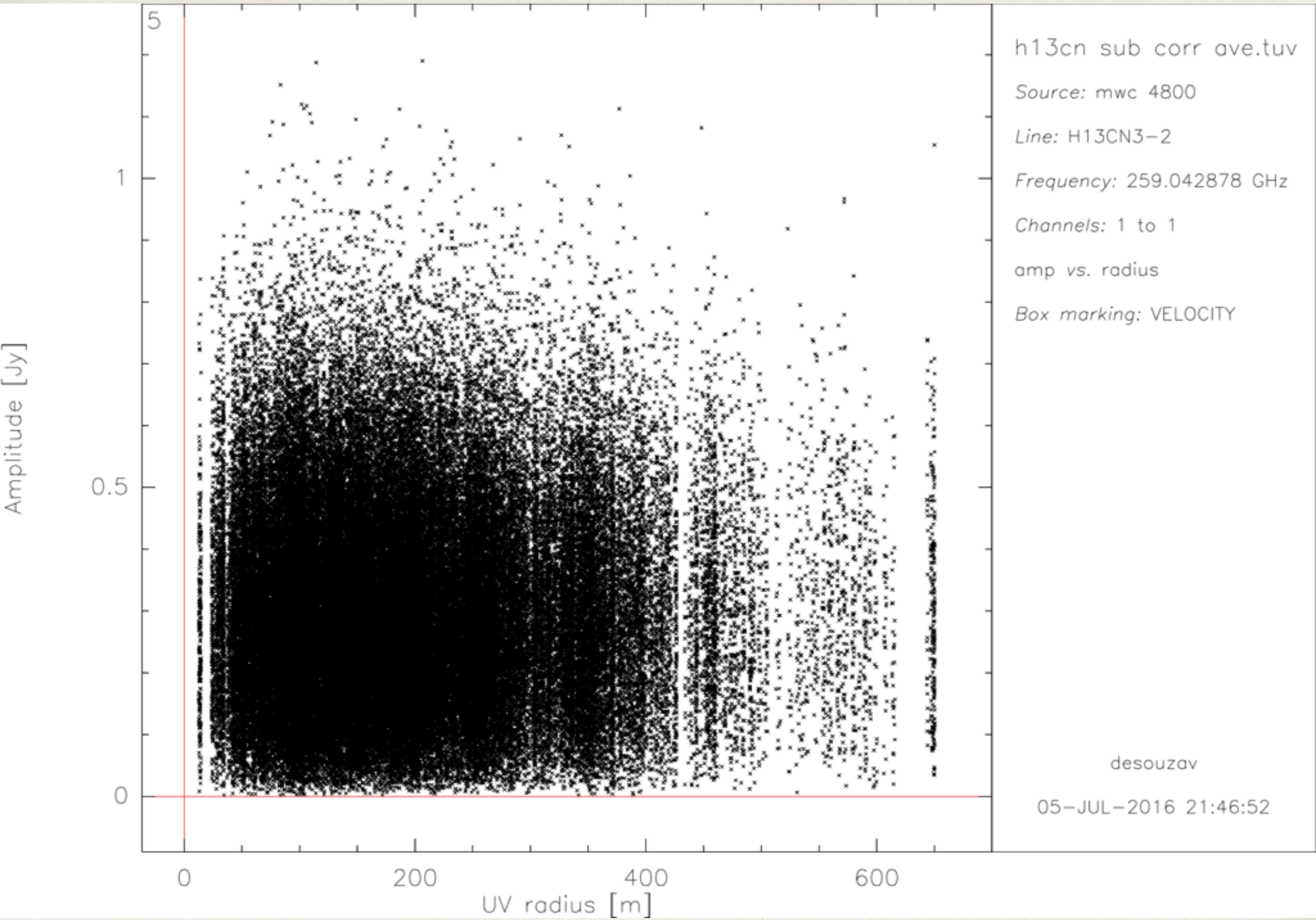
A typical CLASS session

MAPPING

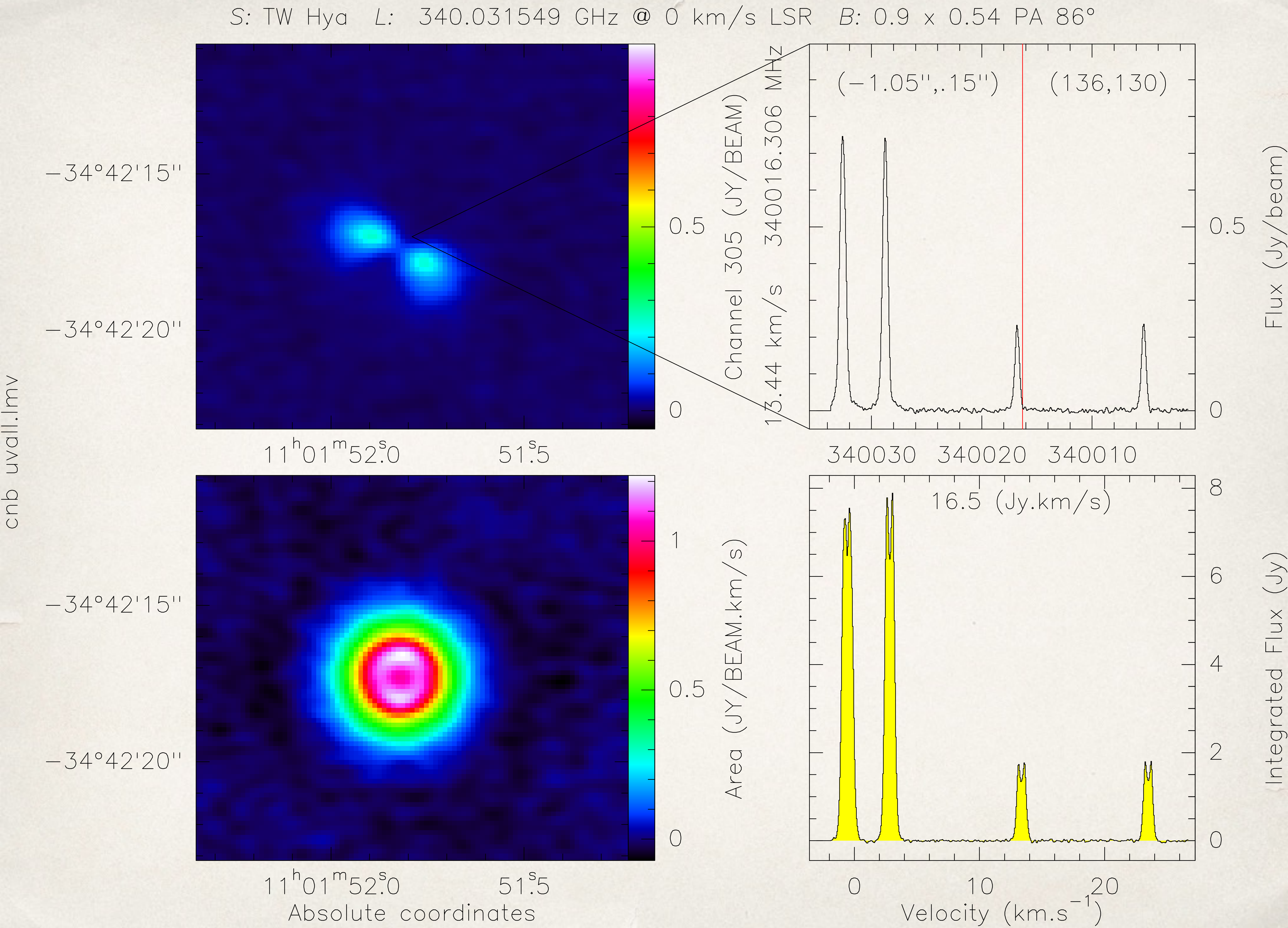
- Reduction of interferometric data.
- UV plane analysis.
- Dirty images.
- Deconvolution.
- Comprehensive plots to understand the data



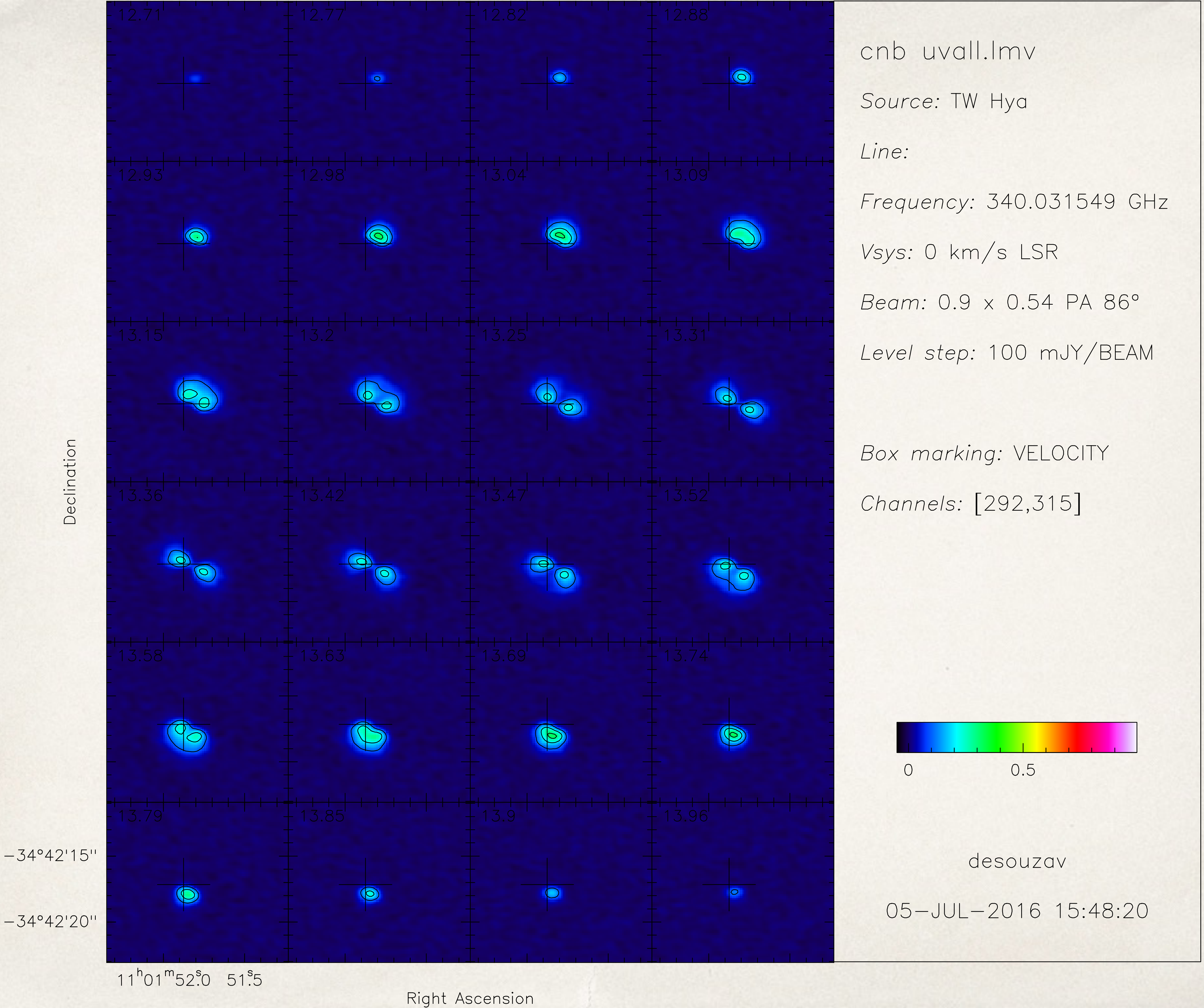
5
500
0
−500
V [m]
−500
0
500
U [m]
h13cn sub corr ave.tuv
Source: mwc 4800
Line: H13CN3−2
Frequency: 259.042878 GHz
Channels: 1 to 1
v vs. u
Box marking: VELOCITY
desouzav
05−JUL−2016 21:45:18



5
1
0.5
0
Amplitude [Jy]
0
200
400
600
UV radius [m]
h13cn sub corr ave.tuv
Source: mwc 4800
Line: H13CN3-2
Frequency: 259.042878 GHz
Channels: 1 to 1
amp vs. radius
Box marking: VELOCITY
desouzav
05-JUL-2016 21:46:52



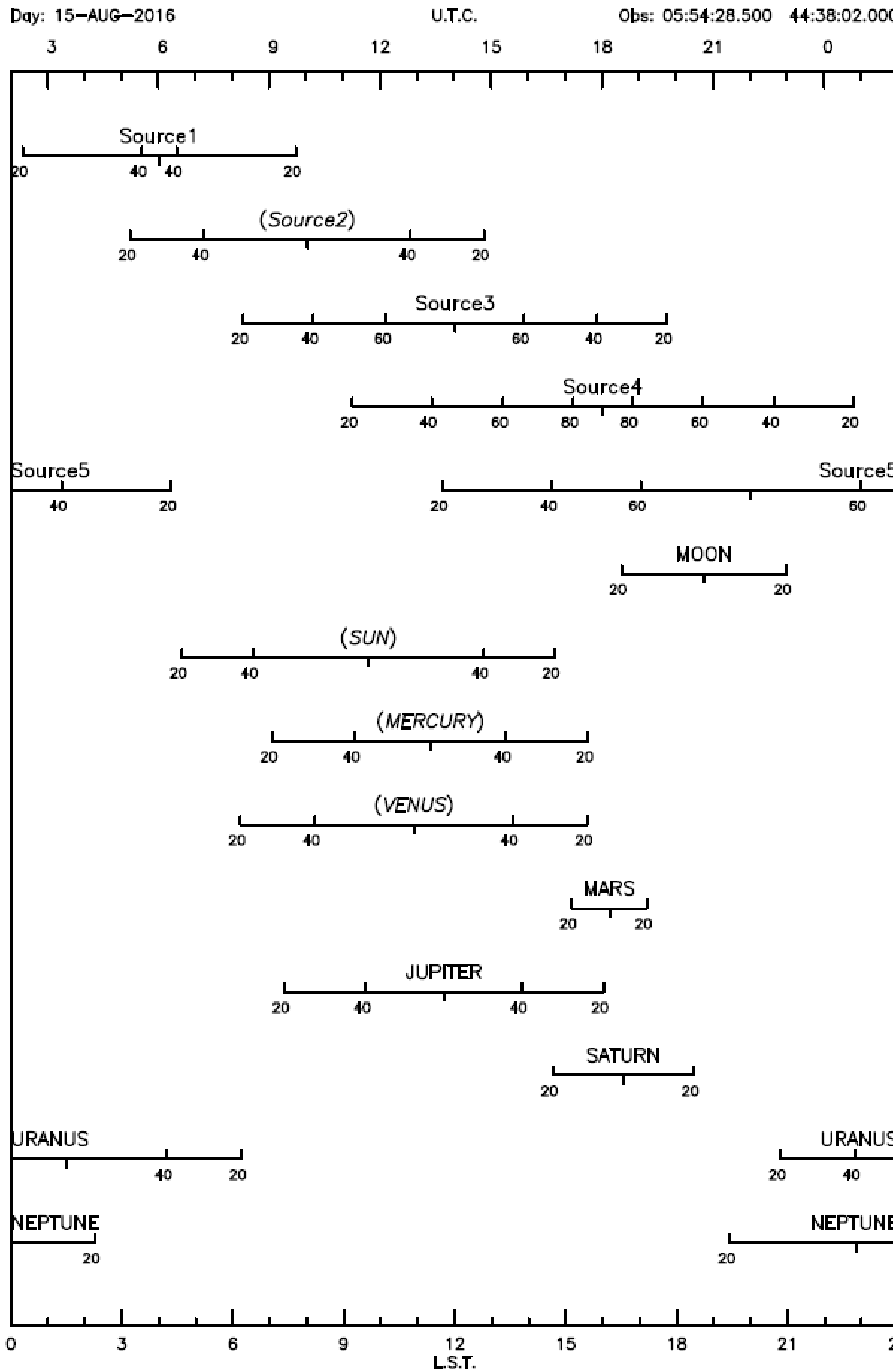
S: TW Hya L: 340.031549 GHz @ 0 km/s LSR B: 0.9 x 0.54 PA 86°
cnb uvall.lmv
−34°42'15''
−34°42'20''
11h01m52.0s
51.5s
Absolute coordinates
0.5
0
1
Channel 305 (JY/BEAM) 340016.306 MHz
13.44 km/s
(−1.05'',.15'')
(136,130)
340030 340020 340010
Flux (Jy/beam)
16.5 (Jy.km/s)
Area (JY/BEAM.km/s)
Integrated Flux (Jy)
8
6
4
2
0
10
20
Velocity (km.s−1)

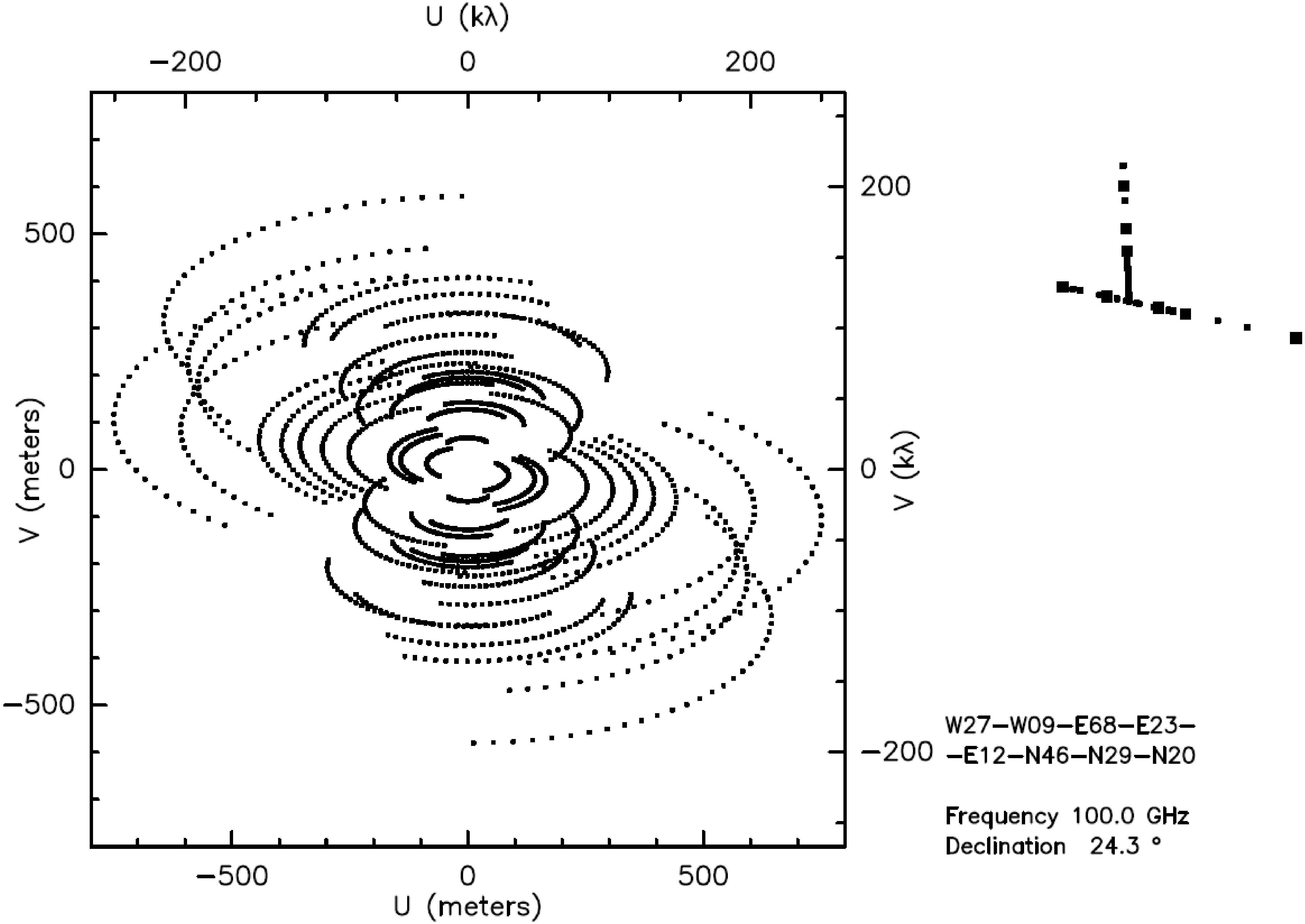


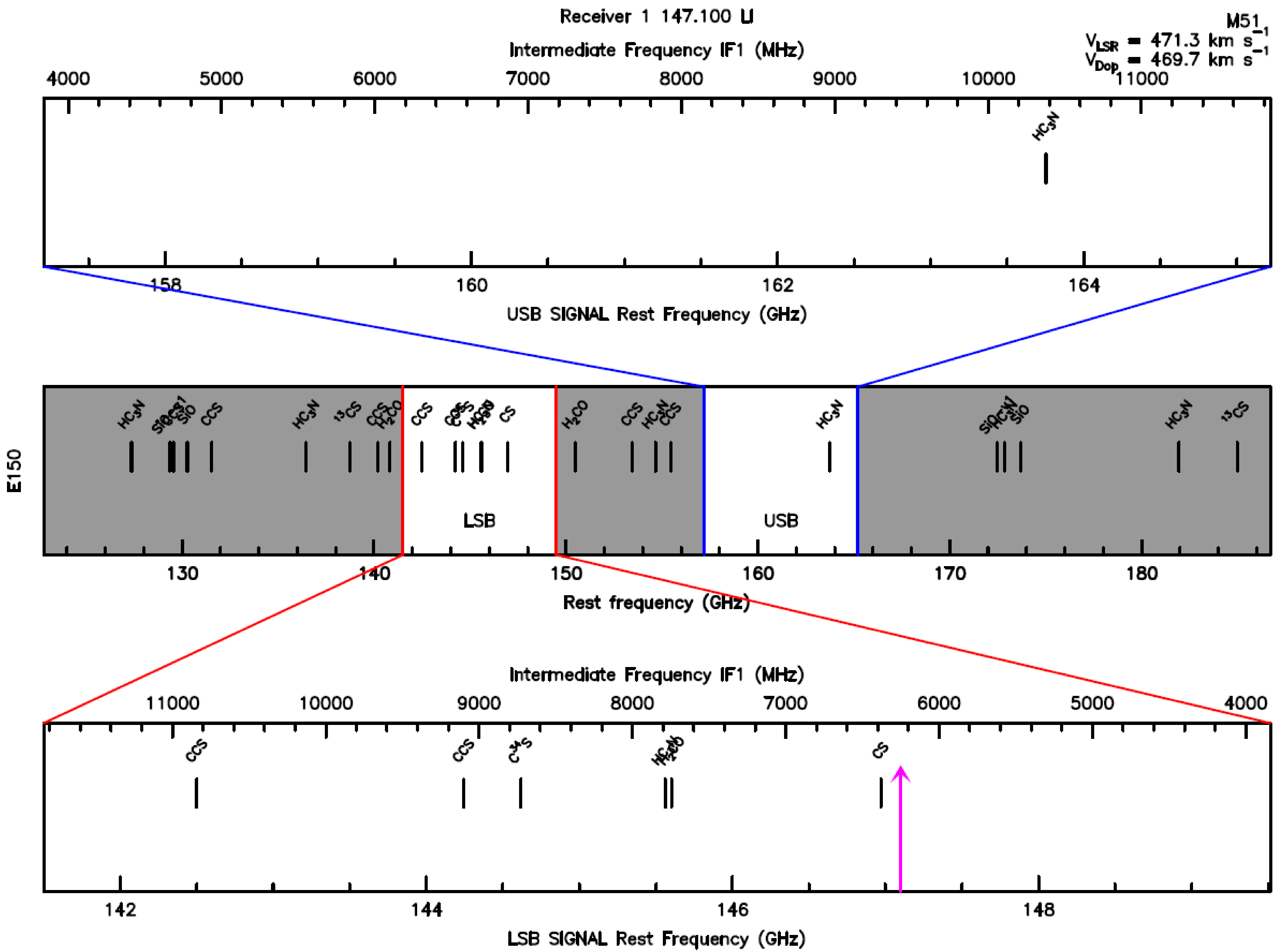
12.71
12.77
12.82
12.88
12.93
12.98
13.04
13.09
13.15
13.2
13.25
13.31
13.36
13.42
13.47
13.52
13.58
13.63
13.69
13.74
13.79
13.85
13.9
13.96
Declination
−34°42'15''
−34°42'20''
11h01m52.0s 51.5s
Right Ascension
cnb uvall.lmv
Source: TW Hya
Line:
Frequency: 340.031549 GHz
Vsys: 0 km/s LSR
Beam: 0.9 x 0.54 PA 86°
Level step: 100 mJY/BEAM
Box marking: VELOCITY
Channels: [292,315]
0
0.5
desouzav
05−JUL−2016 15:48:20

ASTRO

- Help you prepare observations.
- Check source visibility.
- Prepare tuning setups for the observations.
- Check UV coverage for interferometric observation.


Day: 15–AUG–2016
U.T.C.
Obs: 05:54:28.500 44:38:02.000
Source1
(Source2)
Source3
Source4
Source5
MOON
(SUN)
(MERCURY)
(VENUS)
MARS
JUPITER
SATURN
URANUS
NEPTUNE
L.S.T.


U (kλ)
−200
0
200
500
0
−500
V (meters)
200
0
−200
V (kλ)
−500
0
500
U (meters)
W27−W09−E68−E23−
−E12−N46−N29−N20
Frequency 100.0 GHz
Declination 24.3 °



Receiver 1 147.100 LI
M51
V_{LSR} = 471.3 km s^{-1}
V_{Dop} = 469.7 km s^{-1}
Intermediate Frequency IF1 (MHz)
4000 5000 6000 7000 8000 9000 10000 11000
HC_3N
158 160 162 164
USB SIGNAL Rest Frequency (GHz)
E150
HC_3N SiO SiO CCS HC_3N ^{13}CS CCS H_2CO CCS CS H_2CO CCS HC_3N CCS HC_3N SiO SiO HC_3N ^{13}CS
LSB
USB
130 140 150 160 170 180
Rest frequency (GHz)
Intermediate Frequency IF1 (MHz)
11000 10000 9000 8000 7000 6000 5000 4000
CCS CCS $C^{34}S$ HC_3N H_2CO CS
142 144 146 148
LSB SIGNAL Rest Frequency (GHz)

To know more

- IRAM provides a website to support GILDAS: http://www.iram.fr/IRAMFR/GILDAS/
  - There are some tutorials on how to use GILDAS software.
  - Documentation on all the tasks.
  - And a Download area, where you can fetch GILDAS for yourself.

NRAO - AIPS

- AIPS is a very old software from NRAO from the 1970s.
- It is has new releases, basically annually.
- It was created to handle radio interferometric data.
- It was built to be fast.
- It has been mainly superseded by CASA, but it is still faster than CASA.

NRAO - CASA

- CASA - Common Astronomy Software Applications
- Its development started as a rewrite of AIPS into more modern software.
- It is a GUI oriented software for radio astronomy data reduction.
- It is the software with the ALMA calibration pipeline.
- Can be scripted in Python, its prompt is actually a Python prompt, called casapy.

NRAO - CASA

- Holds all data treatment related functionality in one interface:
  - Calibration, reduction.
- Posses various data viewing GUIs, UV plane viewing, image viewing, data cube analysis.
- Some tasks are realised using a GUI, ex: Cleaning.
- Drawback: Some tasks can be very slow (ex: a few hours per cleaning).

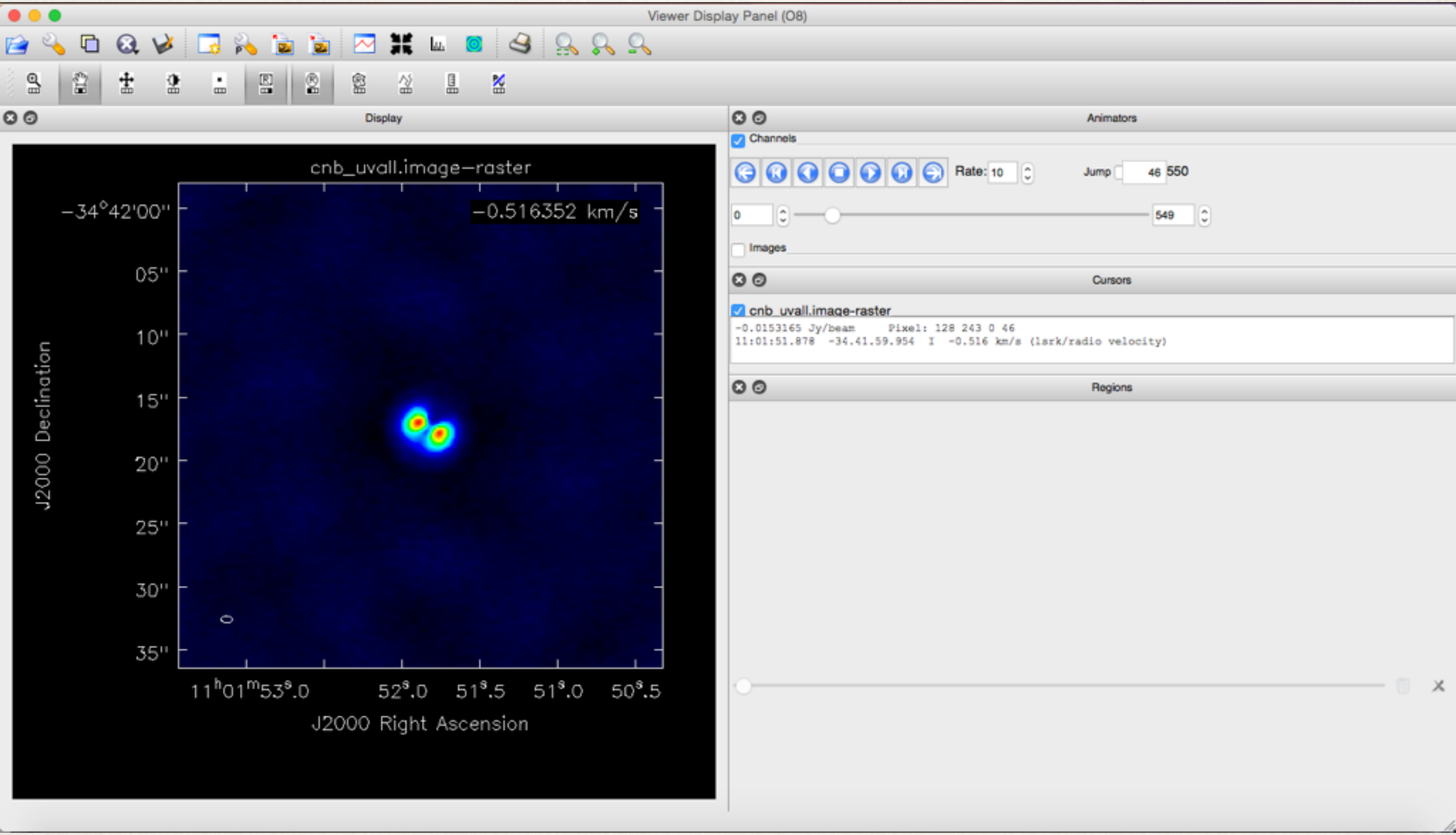
Log Messages (:/Users/desouzav/observations/alma/tw_hya/casa_images/casapy-20160706-121848.log)

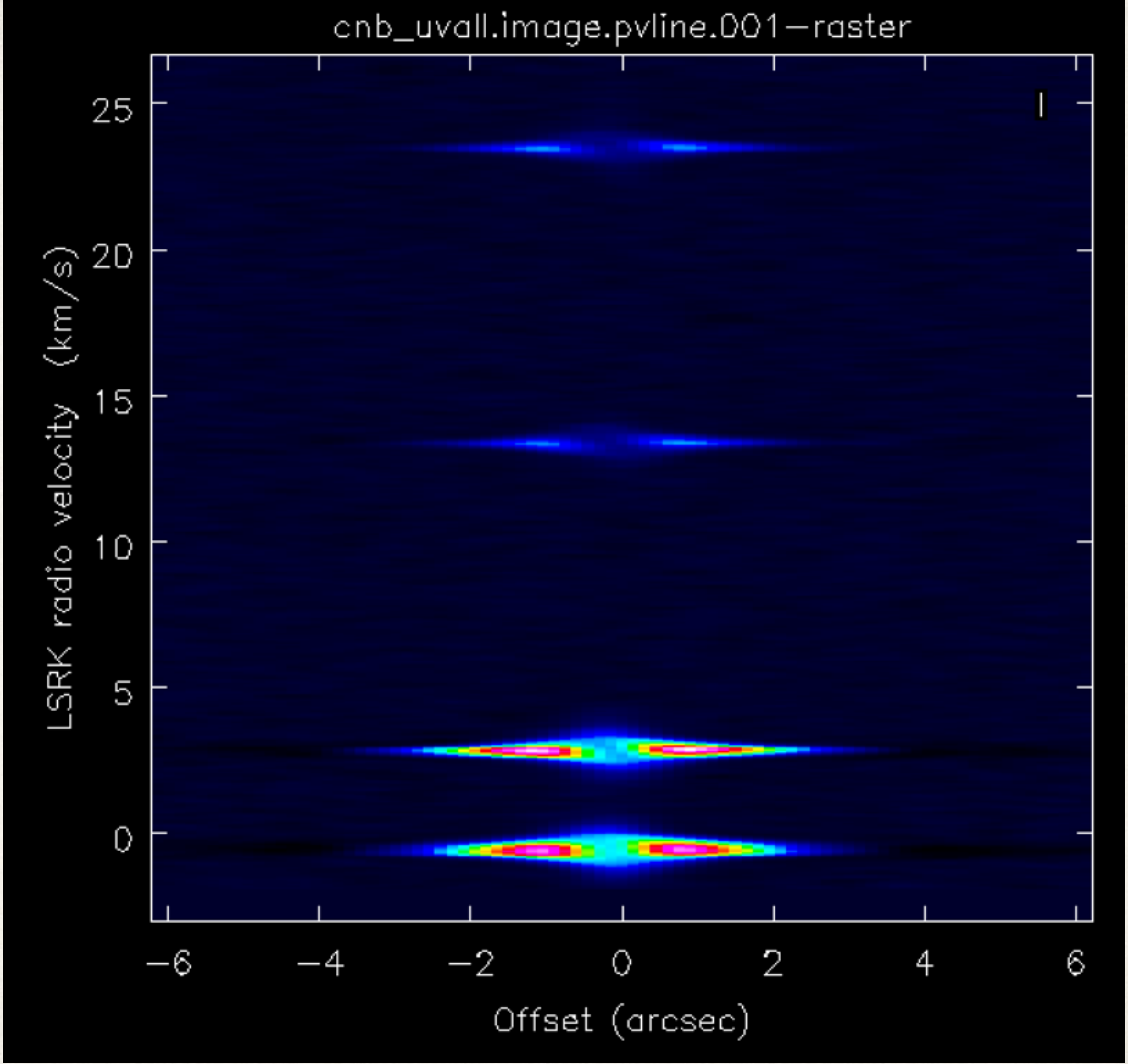
Search Message: Filter: Time

Time	Priority	Origin	Message
	INFO	casa:::casa	---
	INFO	casa:::casa	CASA Version 4.2.2 (prerelease r30986)
	INFO	casa:::casa	Tagged on: Thu, 21 Aug 2014

Insert Message: Lock scroll

```
Raw input log   : False
Timestamping    : False
State           : active
*** Loading ATNF ASAP Package...
*** ... ASAP (4.2.0a rev#30794) import complete ***
##################################################

Major interface changes to SINGLE DISH tasks have been
taken place in CASA 4.2.2 release

The interface of the following tasks are modified:
sdbaseline, sdcal, sdcal2, sdfit, sdflag, sdgrid,
sdimaging, sdmath, sdplot, sdreduce, sdsave, and sdstat.
Additionally, a new task called sdaverage is available. Task
sdsmooth has been incorporated in the new task and removed.

The tasks with old interfaces are available with name
{taskname}old. They will be kept until CASA 4.3 release
and removed from later releases. Users are advised to
update existing scripts.

##################################################

CASA <2>:
```


Viewer Display Panel (O8)
Display
cnb_uvall.image−raster
−0.516352 km/s
−34°42'00"
05"
10"
15"
20"
25"
30"
35"
J2000 Declination
11h01m53s.0
52s.0
51s.5
51s.0
50s.5
J2000 Right Ascension
Animators
Channels
Rate: 10
Jump
46
550
0
549
Images
Cursors
cnb_uvall.image-raster
-0.0153165 Jy/beam Pixel: 128 243 0 46
11:01:51.878 -34.41.59.954 I -0.516 km/s (lsrk/radio velocity)
Regions



cnb_uvall.image.pvline.001−raster
I
LSRK radio velocity (km/s)
25
20
15
10
5
0
−6
−4
−2
0
2
4
6
Offset (arcsec)

To know more

- NRAO maintains a website for CASA: https://casa.nrao.edu
- There is also a good amount of tutorials (some with data to follow them) at: https://casaguides.nrao.edu/

SMA - MIRIAD

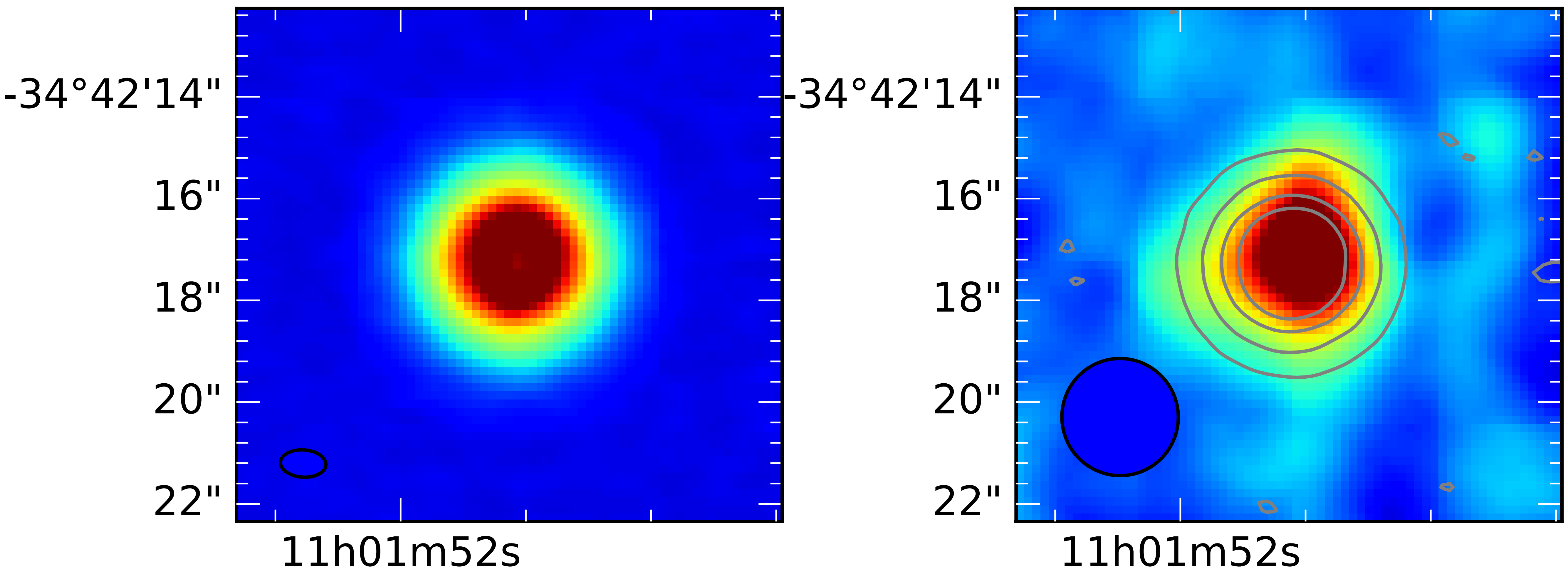
- Its is the software for the reduction of data from the Sub-Millimetre array.
- It can be configured to be used with other observatories as well.
- It integrates into the OS shell, no separate prompt.
- Its scripts are Shell scripts.

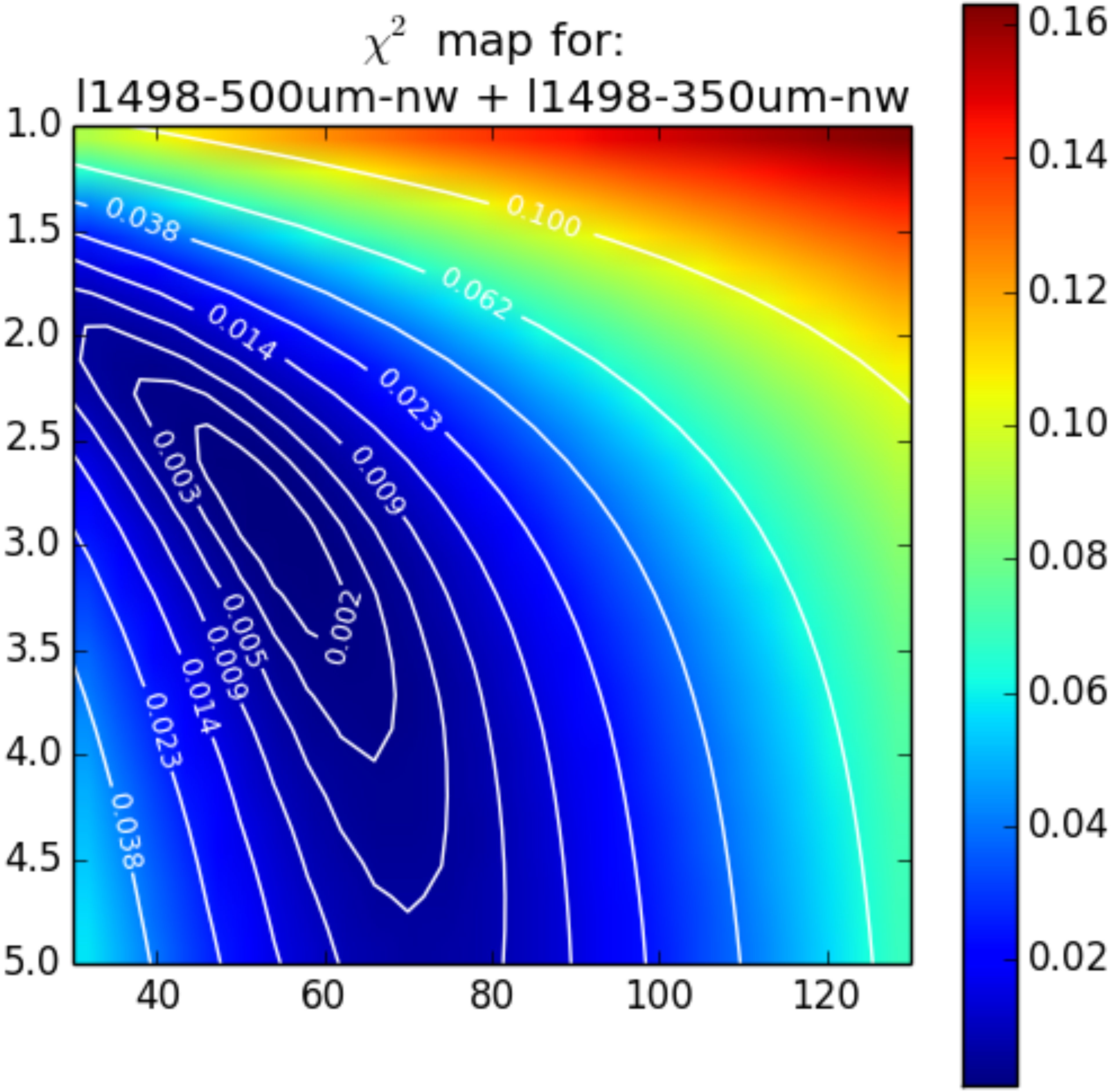
Python

- Python itself has seen the development of Astronomy oriented modules.
- These are not specific for radio astronomy but are very useful to make plots and do some data treatment or analysis.
- Astronomy specific python modules:
  - astropy | pyfits | aplpy | pyspeckit
- Useful modules:
  - numpy | scipy | matplotlib | pandas

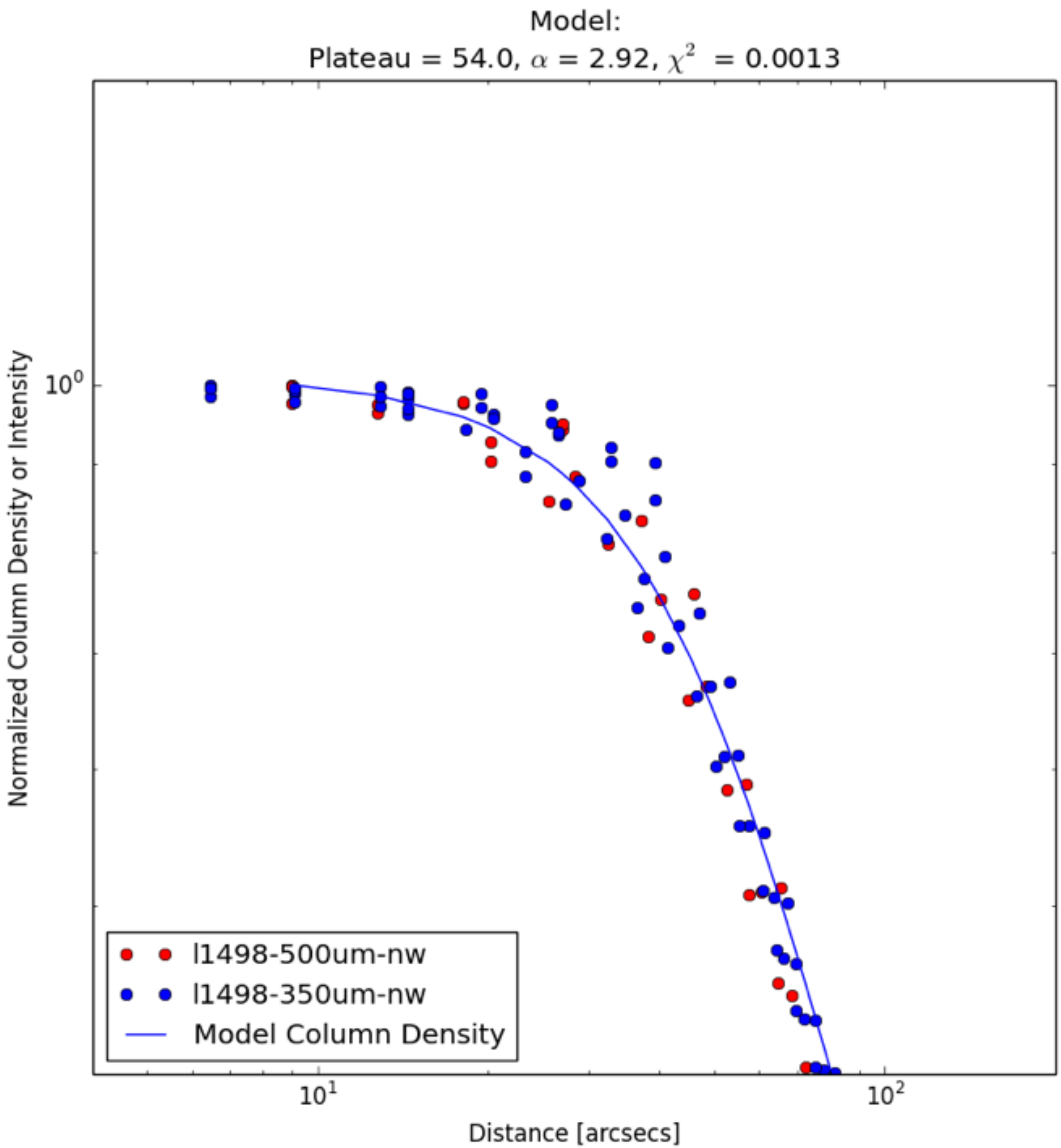
aplpy Example

```
import aplpy
from matplotlib import pyplot as plt
radius = 0.0014 # Radius of the plots in degrees
fig = plt.figure() # matplotlib figure to enable subplots
# Creating a subplot from the fits file
figcn = aplpy.FITSFigure('cnb_uvall.fits',figure=fig,subplot = [0.1,0.1,0.35,0.5])
figcn.show_colorscale() # Displaying the fits file in color
figcn.add_beam() # adding the beam size
xw,yw = figcn.pixel2world(133,128) # Geting the position of the center in WCS
figcn.recenter(xw,yw,radius) # Recentering and resizing the plot
# Decreasing the precision of the WCS displayed on the plot
figcn.tick_labels.set_xformat('hh:mm:ss')
figcn.tick_labels.set_yformat('dd:mm:ss')
# Hiding RA(J2000) and DEC(J2000)
figcn.axis_labels.hide()
## Same as for CN
figc15n = aplpy.FITSFigure('c15nb.fits',figure=fig,subplot = [0.45,0.1,0.35,0.5])
figc15n.show_colorscale()
figc15n.add_beam()
figc15n.recenter(xw,yw,radius)
figc15n.tick_labels.set_xformat('hh:mm:ss')
figc15n.tick_labels.set_yformat('dd:mm:ss')
figc15n.axis_labels.hide()
# Overlaying the CN image as grey contours
figc15n.show_contour('cnb_uvall.fits',colors='grey')
### Hide the tick labels.
figc15n.tick_labels.hide()
figcn.save('CN_C15N.eps')
```


-34°42'14"
16"
18"
20"
22"
11h01m52s
-34°42'14"
16"
18"
20"
22"
11h01m52s



χ^2 map for:
l1498-500um-nw + l1498-350um-nw
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
40
60
80
100
120
0.100
0.062
0.038
0.023
0.014
0.009
0.003
0.005
0.002
0.16
0.14
0.12
0.10
0.08
0.06
0.04
0.02


Model:
Plateau = 54.0, α = 2.92, χ^2 = 0.0013
10^0
Normalized Column Density or Intensity
10^1
10^2
Distance [arcsecs]
l1498-500um-nw
l1498-350um-nw
Model Column Density