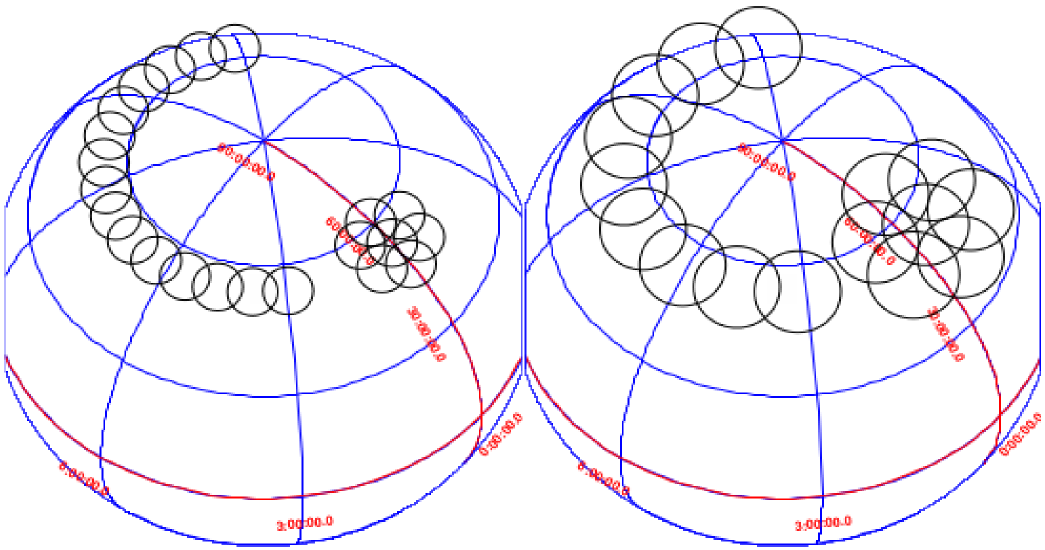
Quick summary of tutorial data set

* Seven sub-array pointings, frequency range 120.1 – 125.8 MHz
* Hexagonal pattern centred at RA 21:00:00.00, Dec. +52:54:00.0 (Galactic Plane field)
* Same fields as from a LOFAR 'radio sky monitor' programme (Transients KSP)
* Calibrator Cygnus A
* Dutch stations only

Tutorial can be downloaded here:
https://dl.dropboxusercontent.com/u/44848769/data_school_tutorial1_FINAL.pdf



Tutorial 1: flagging, averaging and demixing

Jess Broderick (ASTRON)

Goal: to inspect a LOFAR HBA measurement set of a particular target field, and carry out some initial basic processing in preparation for calibration (Tutorial 2).

Note: in this tutorial, '>' is used to indicate a new input line in the terminal.

1. Login sheet

- You should have already received one of these. Your username = **lodsXX**, your reservation = **tutorials_287**, and your node = **lof00Z**.

2. Log in to CEP3

- Log in to the LOFAR portal:
  > **ssh lodsXX@portal.lofar.eu**

- Log in to the head node, **lhd002**:
  > **ssh lhd002**

- Activate a dummy session using your reservation:
  - > **use Slurm**
  - > **srun -A tutorials --reservation=tutorials_287 -N5 -u bash -i**

- Now, open a new terminal (and keep the previous one open too), typing the following commands:
  - > **ssh -Y lodsXX@portal.lofar.eu**
  - > **ssh -Y lhd002**
  - > **ssh -Y lof00Z**

- Verify that graphics forwarding works:
  - > **geany**

- To start with, we are just going to do some light processing. So let's go back to the head node:
  - > **exit**

3. Initial inspection of the data using msoverview

- The data sets for this tutorial can be found in **lhd002:/data/scratch/DATASCHOOL16/T1**. Let's have a quick look at these data to get an idea about the sizes of the measurement sets, as well as the naming convention used.
  - > **cd /data/scratch/DATASCHOOL16/T1**
  - > **du -sh *uv.MS**

- The naming convention for each measurement set (also referred to as a sub-band) is L*aaaaaa*_SAP*bbb*_SB*ccc*_uv.MS, where L*aaaaaa* is the observation/pipeline ID, SAP*bbb* is the sub-array pointing (beam), and SB*ccc* is the sub-band number.

- *What is the data volume per sub-band? In fact, there are 210 sub-bands for this observation (although we won't look at all of them in this tutorial): what is the total data volume then, and how does this compare with data from other telescopes that you may have used?*

- The next step is to take a closer look at the data. To do this, we load a set of standard software packages:
  > **use Lofar**

- Now we can, for example, obtain a summary of a measurement set using **msoverview**:
  > **msoverview in=L456104_SAP000_SB001_uv.MS**

- *Which field was observed? What was the duration of the observation? What was the centre frequency of this particular sub-band? How many channels (frequencies) are in the data set? What array configuration was used and what does the configuration name imply?*

- More detailed information in **msoverview** can be displayed using the 'verbose' argument:
  > **msoverview in=L456104_SAP000_SB001_uv.MS verbose=T**

- *What is the number of time slots? What is the integration time per time step? How many stations (core and remote), and how many baselines? What is the relation between number of stations and baselines (excluding autocorrelations)?*

- You may have noticed the message 'This is a raw LOFAR MS (stored with LofarStMan)' in **msoverview**. This means that the data cannot be handled with Casa. Let's fix the problem. First, go back to your assigned compute node, and make a directory to do your work in.

```
> ssh -Y lof00Z
> cd /data/scratch
> mkdir lodsXX ; cd lodsXX
> mkdir T1 ; cd T1
> use Lofar
```

(note: we have to load the LOFAR packages again on this node)

- Copy over sub-band SB001 to this directory:

```
> scp -r lhd002:/data/scratch/DATASCHOOL16/T1/L456104_SAP000_SB001_uv.MS .
```

- A simple **DPPP** (Default PreProcessing Pipeline) parset is required to make the data readable in Casa:

```
> geany DPPP-makecopy.parset     # (or vim, emacs, nano, nedit etc.)
```

Put the following commands in the parset and save it:

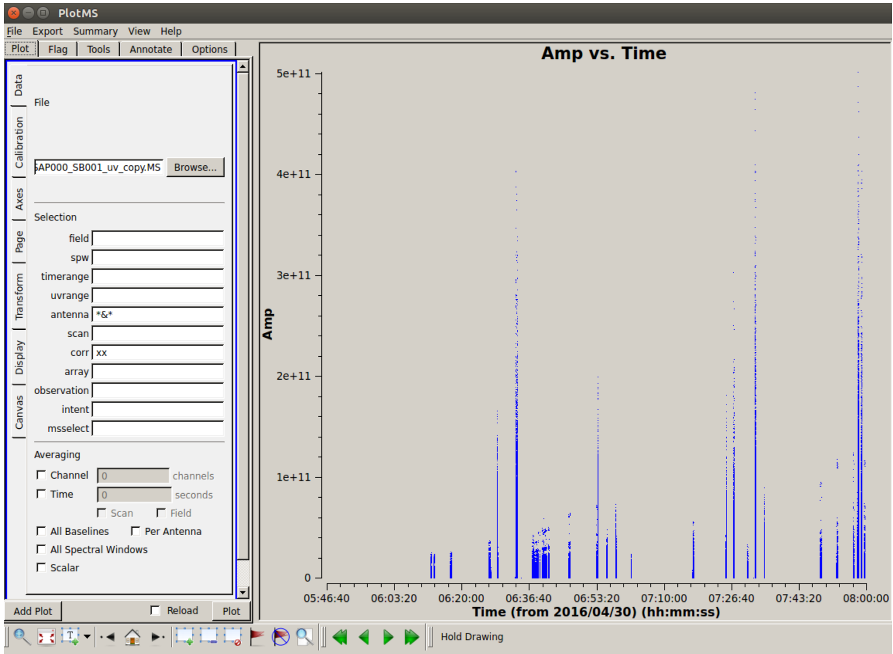
```
msin=L456104_SAP000_SB001_uv.MS
msout=L456104_SAP000_SB001_uv_copy.MS
msin.autoweight=True
numthreads=4     # so that the nodes are not overloaded, usually can skip this line
steps = [ ]
```

- Run **DPPP** on this parset:
  - > **DPPP DPPP-makecopy.parset**

- The output measurement set can now be opened using the standard Casa tools. Moreover, proper weighting of the data has been implemented (see the LOFAR cookbook for more information).

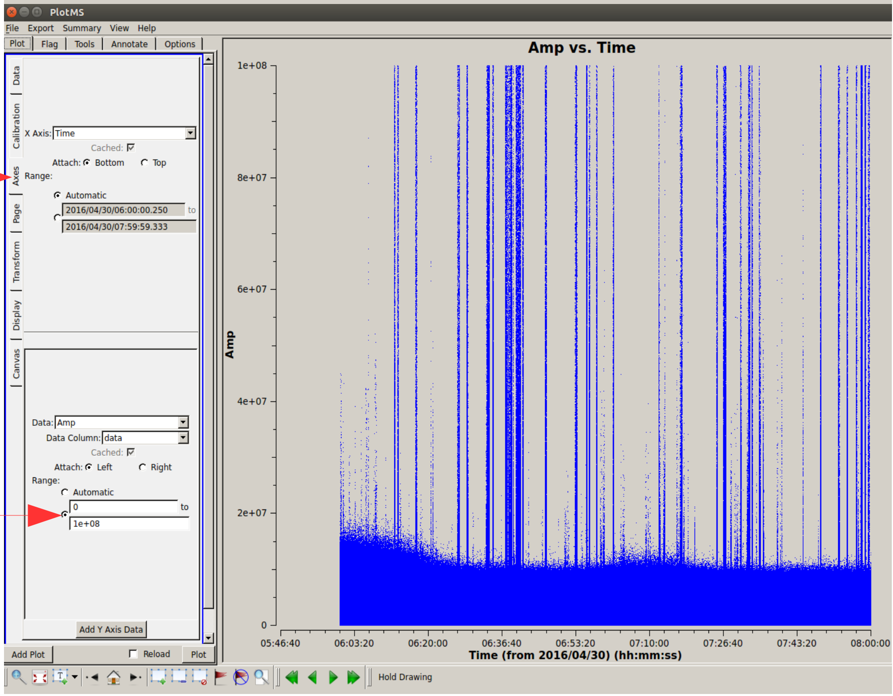
4. casaplotms

- Let's take a look at the measurement set in **casaplotms**:
  - > **use Casa # initialize CASA software**
  - > **casaplotms &**



- Open the measurement set from the GUI. Plot the XX correlation only by using the 'corr' argument (to speed things up). Select only cross correlations by typing '*&*' in the 'antenna' field (note: * - any antenna; & - cross correlations).

- There are many large spikes, likely caused by RFI. Click on the 'Axes' tab and adjust the y-axis range to something more sensible to find the real astronomical signal (e.g. max. = 1e+08).



- *Does the y-axis scale make sense to you? What other plotting options could be useful?*

5. rfigui

- We can also further visualize the impact of RFI in the measurement set using **rfigui**.
  - **> rfigui L456104_SAP000_SB001_uv_copy.MS &**

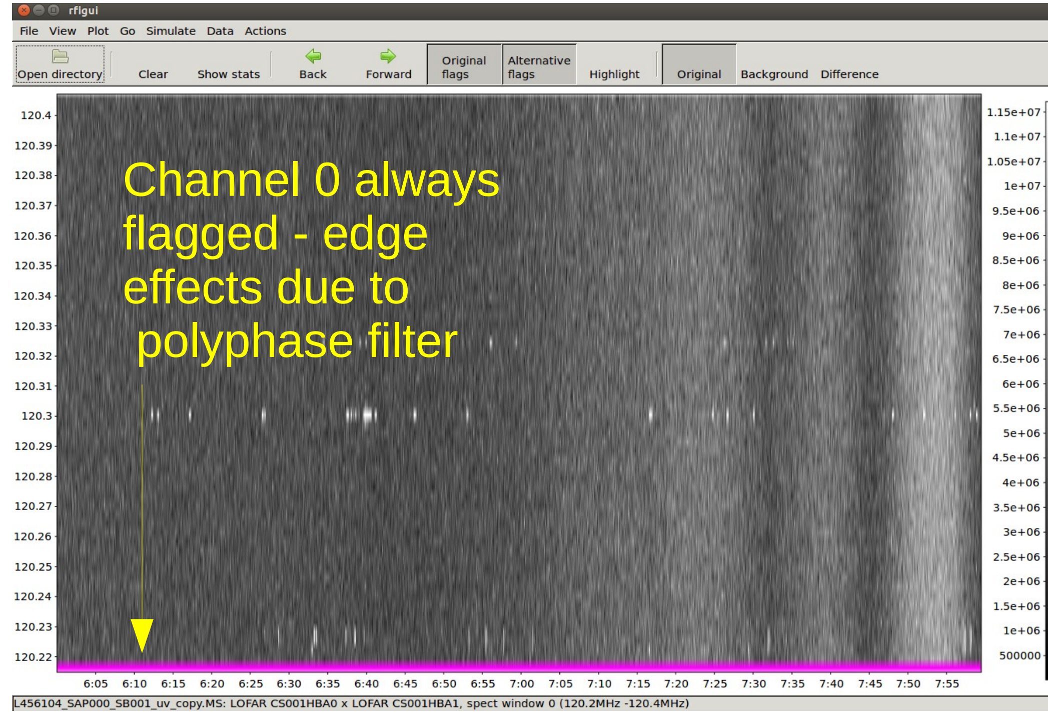
- In the smaller menu that pops up, just choose ‘Open’. Then, in the new window, select index 0 (default) for antenna 1, and index 1 for antenna 2, before pressing 'Load'. This will plot the short intra-station baseline CS001HBA0 - CS001HBA1.

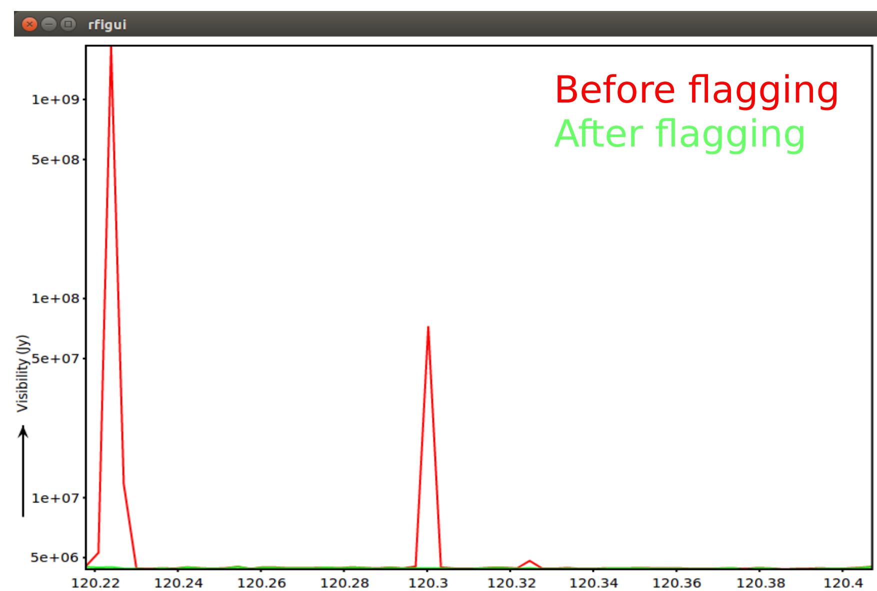
- There is some clear narrow-band RFI. Create a power spectrum (under 'Plot').

- *What could be causing the RFI? Are the fainter broadband structures at later times also RFI?*

- **rfigui** can flag the data using **AOFlagger**: 'Actions'; 'Execute Strategy'. Also, find other useful plots in the plot menu.

- *Would the plot look different if we had chosen e.g. a baseline between a core and remote station? Check this ('Go'; 'Go to...').*





6. AOFlagger (Offringa et al. 2010, 2012a,b; http://aoflagger.sourceforge.net/)

- The **AOFlagger** can be called with **DPPP**. Create and run this parset (this time pipe the output to a log file, e.g. **> DPPP your.parset | tee DPPP.log**):

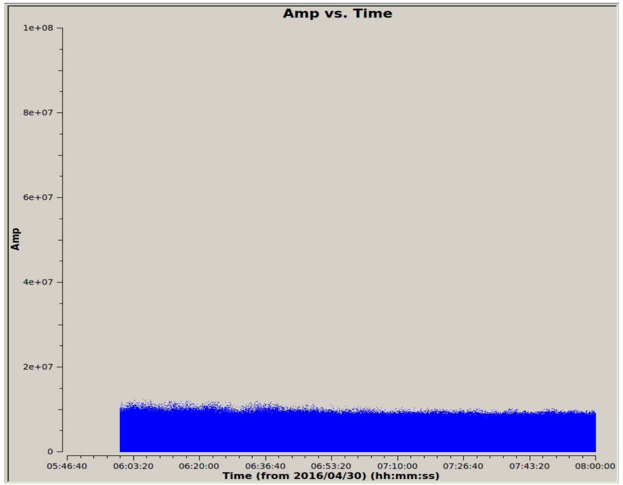
```
msin=L456104_SAP000_SB001_uv_copy.MS
msout=L456104_SAP000_SB001_uv_copy_flg.MS
numthreads=4
steps=[preflagger,aoflagger]
preflagger.baseline=*&&&;RS208HBA;RS210HBA;RS310HBA;RS406HBA;RS407HBA;RS409HBA;
RS508HBA;RS509HBA;RS205HBA&RS307HBA    # all these stations on one line
```

- An explanation of the 'preflagger' step is as follows. The first argument (arguments separated by semi-colons) removes the autocorrelations (although note that **AOFlagger** also does this by default). For the next eight arguments, we remove all baselines including the stated remote station (for calibration reasons related to Tutorial 2). The last argument removes one particular baseline between two remote stations, again for reasons related to calibration.

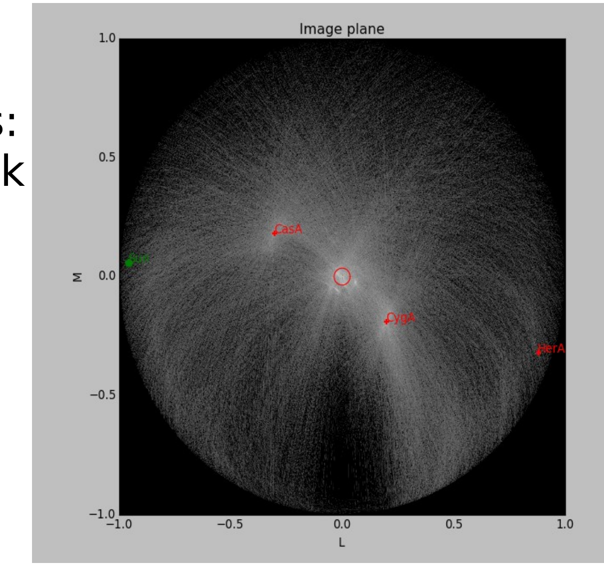
- The default **AOFlagger** strategy is run, but many options can be specified in **DPPP** (see cookbook as well as LOFAR wiki).

- *Looking at the **DPPP** output, which channels were most affected? What are their frequencies?*

- *Re-examine the data with **casaplotms**. Is the RFI gone now?*



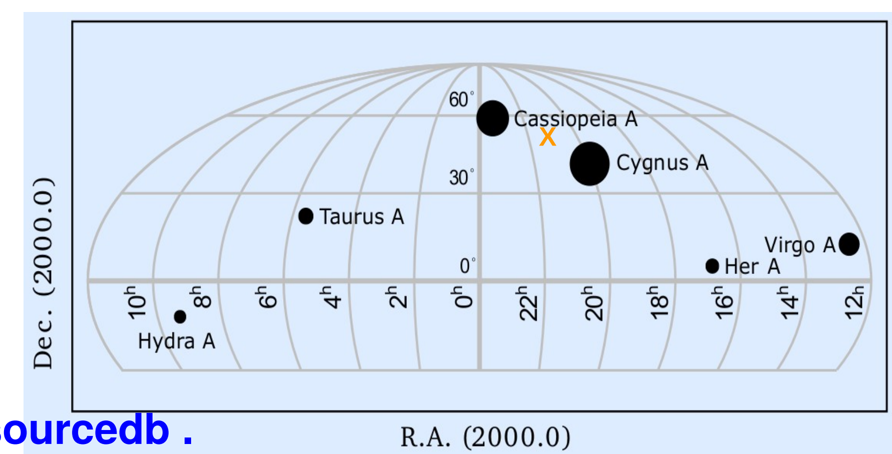
7. Demixing



- The observation we are working on is close to two 'A-team' sources: about 16 deg from Cyg A, and 21 deg from Cas A. We need to check if their responses should be 'demixed' from the data. Firstly, let's check their elevations during the observation:
  - **> use Cookbook # loading more software**
  - **> plot_Ateam_elevation.py L456104_SAP000_SB001_uv_copy_flg.MS**

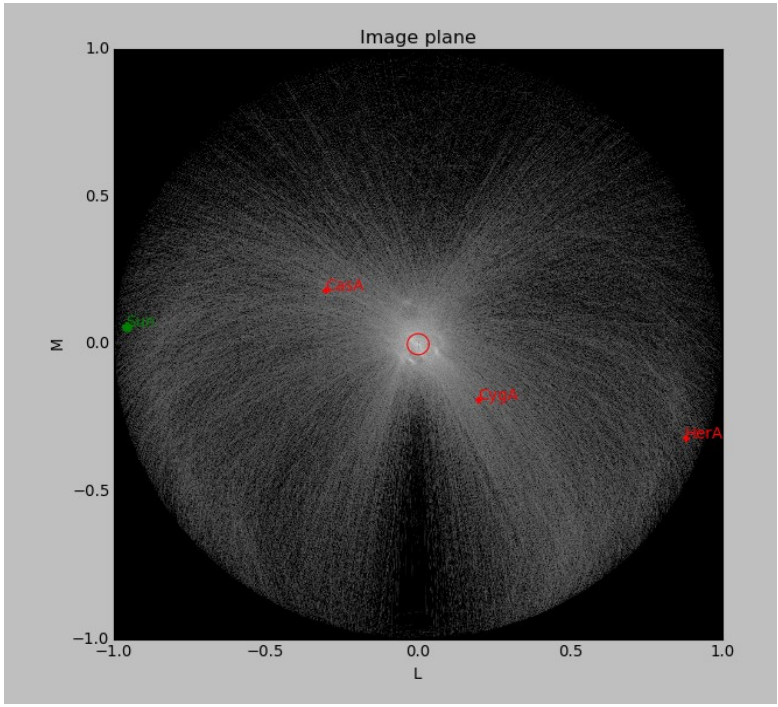
- We can predict the associated visibilities through simulations (see cookbook). Today, however, we will run a tool called the **Drawer**, allowing one to quickly inspect a measurement set and investigate which sources are contributing to the visibilities.
  - **> scp -r lhd002:/data/scratch/DATASCHOOL16/T1/L456104_SAP000_SB001_uv_copy_demix_demo.MS .**
  - **> /home/tasse/drawMS/drawMS --ms=L456104_SAP000_SB001_uv_copy_demix_demo.MS**

- Note that we are running the **Drawer** on pre-prepared averaged data (to save time); we will learn how to do averaging shortly. Examine the **Drawer** output. One can see contributions from Cas A and Cyg A. Time to demix!



- We need a model of the A-team sources:
  - **> scp -r lhd002:/data/scratch/DATASCHOOL16/T1/Ateam.sourcedb .**
  - **> showsourcedb in=Ateam.sourcedb mode=patch # verify database contents**

- Use the following parset for **DPPP**:
  msin=L456104_SAP000_SB001_uv_copy_flg.MS
  msout=L456104_SAP000_SB001_uv_copy_flg_demix_avg.MS
  numthreads=4
  steps=[demix]
  demix.subtractsources=[CasA,CygA]
  demix.skymodel=Ateam.sourcedb
  demix.timestep=10
  demix.freqstep=16
  demix.demixtimestep=60
  demix.demixfreqstep=64



- The arguments 'timestep' and 'freqstep' compress the data in time and frequency, respectively, by the given factors.

- Demixing two sources at once can be very time consuming. Thus, 'demixtimestep' and 'demixfreqstep' have been chosen to have rather coarse values (usually default to 'timestep' and 'freqstep' without needing to be specified).

- *Run the **Drawer** again. Did we succeed in demixing Cyg A and Cas A from the data?*
  > **/home/tasse/drawMS/drawMS --ms=L456104_SAP000_SB001_uv_copy_flg_demix_avg.MS**

- *What is the size of the compressed, demixed measurement set? Is this what you expect?*

- *Do you think that the time and frequency averaging settings are reasonable? How could they affect the calibration and imaging?*

8. Averaging and combining tasks in DPPP

- What if demixing is not needed? We can also just average in time and frequency. Here is a parset that could be run through **DPPP**:

```
msin=L456104_SAP000_SB001_uv_copy_flg.MS
msout=L456104_SAP000_SB001_uv_copy_flg_avg.MS
numthreads=4
steps=[averager]
averager.timestep=10
averager.freqstep=16
```

- **DPPP** was designed to run pipelines containing multiple steps. Save the following parset as 'DPPP-all.parset', in preparation for the final part of the tutorial. This parset combines all the steps we have run in this tutorial, except that we skip demixing (too time consuming).

```
msin=L456104_SAP000_SB001_uv.MS
msin.autoweight=true
msout=L456104_SAP000_SB001_uv.MS.dppp
numthreads=4
steps=[preflagger,aoflagger,averager]
preflagger.baseline=*&&&;RS208HBA;RS210HBA;RS310HBA;RS406HBA;RS407HBA;RS409HBA;
RS508HBA;RS509HBA;RS205HBA&RS307HBA
averager.timestep=10
averager.freqstep=16
```

9. Scripting

- Clearly, it is impractical to do a manual reduction for a full observation of hundreds of sub-bands, although keep in mind that looking at your LOFAR data at least at some level is always highly recommended. Write a script (e.g. in Python or Bash) that will automate the process that we have just been through, applying it to the four sub-bands in the tutorial directory.

- Firstly, note that you can download target sub-bands to your directory using a command like this:

  > scp -r lhd002:/data/scratch/DATASCHOOL16/T1/L456104_SAP000_SB000_uv.MS .

- Secondly, you can override keys in your parset from the previous slide by specifying them on the command line, e.g.:

  > DPPP DPPP-all.parset msin=your.MS msout=your.MS.dppp

- A possible solution in bash:

```
#!/bin/bash
for inputname in $(ssh lhd002 'ls -d /data/scratch/DATASCHOOL16/T1/*uv.MS | xargs -n1 basename'); do
  echo "Copying ${inputname} from head node to working node"
  scp -r lhd002:/data/scratch/DATASCHOOL16/T1/${inputname} .
  outputname=$(echo ${inputname} | sed "s/MS/MS.dppp/g")
  echo ${outputname}
  DPPP DPPP-all.parset msin=${inputname} msout=${outputname}
  echo "Removing raw data ${inputname} from working node"
  rm -rf ${inputname}
done
```

- After running your script, you should have four processed measurement sets (*.dppp). Use **msoverview**, **casaplotms**, etc. to verify that everything was processed successfully.

10. Final remarks

- This tutorial has given a brief overview of some of the initial processing steps for a LOFAR data set. It is highly recommended to also review chapters 2-4 in the cookbook, where more information can be found about the tools used in this tutorial, as well as other software that is available. **(http://www.astron.nl/radio-observatory/lofar/lofar-imaging-cookbook)**

- Time permitting, you could, for example, try the following:

  (i) Simulating visibilities to predict A-team contributions: section 2.3.3 in the cookbook. *Make sure to run any tests in a separate sub-directory on a dummy copy of the compressed data!*

  (ii) Experimenting with **AOFlagger** settings: section 3 in the cookbook.

- Many/all of the steps in this tutorial can be carried out by the Radio Observatory as part of pre-processing. If in doubt about the best settings, always contact Science Support.

- *Always make sure to look at (at least some of) your data!!*

11. Extra slides / walkthrough

```
broderick@lhd002:/data/scratch/DATASCHOOL16/T1$ du -sh *uv.MS
29G     L456104_SAP000_SB000_uv.MS
29G     L456104_SAP000_SB001_uv.MS
29G     L456104_SAP000_SB002_uv.MS
29G     L456104_SAP000_SB003_uv.MS
```

* Raw data volumes very large!
* Total data rate can be ~few GB/s when using full 96 MHz bandwidth.
* Data volume calculator at

http://lofar.astron.nl/service/pages/storageCalculator/calculate.jsp

```
broderick@lhd002:/data/scratch/DATASCHOOL16/T1$ msoverview in=L456104_SAP000_SB001_uv.MS
msoverview: Version 20110407GvD
================================================================================
           MeasurementSet Name:  /data/scratch/DATASCHOOL16/T1/L456104_SAP000_SB001_uv.MS      MS Version 2
================================================================================
           This is a raw LOFAR MS (stored with LofarStMan)

   Observer: unknown     Project: LOFARSCHOOL
Observation: LOFAR
Antenna-set: HBA_DUAL_INNER

Data records: 14042070       Total elapsed time = 7199.99 seconds
   Observed from   30-Apr-2016/06:00:00.0   to   30-Apr-2016/08:00:00.0 (UTC)


Fields: 1
  ID   Code Name                RA               Decl           Epoch        nRows
  0    beam 0BEAM_0              21:00:00.000000 +52.54.00.00000 J2000     14042070

Spectral Windows:  (1 unique spectral windows and 1 unique polarization setups)
  SpwID  Name   #Chans   Frame    Ch0(MHz)  ChanWid(kHz)  TotBW(kHz) CtrFreq(MHz)  Corrs
  0      SB-1       64   TOPO     120.215          3.052       195.3    120.3110   XX  XY  YX  YY
```

= cannot load in Casa (solution: tutorial slide 4)

Array config. - all core sub-stations and inner 24 tiles on remote stations. Same FoV per station. Often gives better quality images.

Sub-array pointing

```
  34   CS101HBA0LOFAR     31.3 m   +006.52.51.7  +52.44.11.4           0.0058        -0.1524  6364567.1930  3825899.977000   461698.906000  5065
339.205000
  35   CS101HBA1LOFAR     31.3 m   +006.52.49.9  +52.44.15.4           0.0055        -0.1514  6364566.8126  3825805.288000   461653.869000  5065
414.350000
  36   CS103HBA0LOFAR     31.3 m   +006.53.45.2  +52.43.48.8           0.0134        -0.1577  6364569.7367  3826331.590000   462759.074000  5064
919.620000
  37   CS103HBA1LOFAR     31.3 m   +006.53.50.1  +52.43.51.8           0.0141        -0.1570  6364569.4814  3826248.441000   462840.933000  5064
974.634000
  38   CS201HBA0LOFAR     31.3 m   +006.52.55.0  +52.43.39.2           0.0062        -0.1600  6364571.2825  3826679.281000   461855.243000  5064
741.380000
  39   CS201HBA1LOFAR     31.3 m   +006.53.01.7  +52.43.38.1           0.0072        -0.1602  6364571.4109  3826690.821000   461982.139000  5064
721.249000
  40   CS301HBA0LOFAR     31.3 m   +006.52.07.4  +52.43.12.4          -0.0006        -0.1663  6364574.8646  3827442.564000   461050.814000  5064
242.391000
  41   CS301HBA1LOFAR     31.3 m   +006.52.00.7  +52.43.13.5          -0.0015        -0.1661  6364574.7517  3827431.025000   460923.919000  5064
262.521000
  42   CS302HBA0LOFAR     31.3 m   +006.50.53.7  +52.42.57.6          -0.0111        -0.1698  6364585.8501  3827973.226000   459728.624000  5063
975.300000
  43   CS302HBA1LOFAR     31.3 m   +006.50.58.6  +52.43.00.6          -0.0104        -0.1691  6364585.5254  3827890.077000   459810.483000  5064
030.313000
  44   CS401HBA0LOFAR     31.3 m   +006.51.24.1  +52.43.42.8          -0.0068        -0.1591  6364572.1931  3826795.752000   460158.894000  5064
808.929000
  45   CS401HBA1LOFAR     31.3 m   +006.51.17.4  +52.43.43.9          -0.0077        -0.1589  6364572.1017  3826784.211000   460031.993000  5064
829.062000
  46   CS501HBA0LOFAR     31.3 m   +006.51.57.9  +52.44.29.9          -0.0019        -0.1480  6364565.6054  3825568.820000   460647.620000  5065
683.028000
  47   CS501HBA1LOFAR     31.3 m   +006.51.59.7  +52.44.25.8          -0.0017        -0.1489  6364565.9714  3825663.508000   460692.658000  5065
607.883000
  48   RS106HBALOFAR     31.3 m   +006.59.05.6  +52.41.21.6          0.0592        -0.1926  6364586.7503  3829205.598000   469142.533000  50621
81.002000
  49   RS205HBALOFAR     31.3 m   +006.53.50.8  +52.40.17.6          0.0142        -0.2078  6364593.7597  3831479.670000   463487.529000  50609
89.903000
  50   RS208HBALOFAR     31.3 m   +006.55.10.4  +52.29.03.1          0.0256        -0.3676  6364664.3511  3847753.310000   466962.809000  50483
97.244000
  51   RS210HBALOFAR     31.3 m   +006.52.29.2  +52.08.40.9          0.0025        -0.6573  6364795.2167  3877827.561860   467536.604956  50254
45.584000
  52   RS305HBALOFAR     31.3 m   +006.46.21.4  +52.42.51.1         -0.0500        -0.1714  6364581.4811  3828732.721000   454692.421000  50638
50.334000
  53   RS306HBALOFAR     31.3 m   +006.44.32.3  +52.42.18.5         -0.0656        -0.1791  6364585.6319  3829771.249000   452761.702000  50632
43.181000
  54   RS307HBALOFAR     31.3 m   +006.40.54.8  +52.37.02.6         -0.0967        -0.2540  6364619.3796  3837964.520000   449627.261000  50573
57.585000
  55   RS310HBALOFAR     31.3 m   +006.08.21.3  +52.34.46.0         -0.3758        -0.2863  6364616.3511  3845376.290000   413616.564000  50547
96.341000
  56   RS406HBALOFAR     31.3 m   +006.45.04.3  +52.49.59.6         -0.0610        -0.0698  6364542.6649  3818424.939000   452020.269000  50718
17.644000
  57   RS407HBALOFAR     31.3 m   +006.47.03.9  +52.54.25.1         -0.0439        -0.0069  6364499.3582  3811649.455000   453459.894000  50767
28.952000
  58   RS409HBALOFAR     31.3 m   +006.21.26.0  +52.47.41.8         -0.2637        -0.1025  6364547.8842  3824812.621000   426130.330000  50692
51.754000
  59   RS503HBALOFAR     31.3 m   +006.51.04.8  +52.45.33.2         -0.0095        -0.1330  6364557.2108  3824138.566000   459476.972000  50668
58.578000
  60   RS508HBALOFAR     31.3 m   +006.57.13.3  +53.03.21.7          0.0431         0.1202  6364441.8110  3797136.484000   463114.447000  50866
51.286000
  61   RS509HBALOFAR     31.3 m   +006.47.04.7  +53.13.30.1         -0.0438         0.2644  6364384.5199  3783537.525000   450130.064000  50978
66.146000

The MS is fully regular, thus suitable for BBS
   nrows=14042070   ntimes=7190   nbands=1   nbaselines=1953 (62 autocorr)
```

CS – core station
RS – remote station

(note: no international stations in this data set)

HBA 0/1 – core station 'ears'

(62*61)/2 cross-correlation baselines
= 1891
= 1953-62
(note first station = no. 0 in **msoverview** output)

* Some of the verbose=T output from **msoverview**.
* Note that the same sized diameter for core/remote stations (i.e. 'INNER' for the remote station tiles).

```
broderick@lof012:/data/scratch/broderick/lofar_school/T1$ DPPP DPPP-makecopy.parset
 copying info and subtables ...
Finished preparing output MS
MSReader
  input MS:       /data/scratch/broderick/lofar_school/T1/L456104_SAP000_SB001_uv.MS
  band            0
  startchan:      0  (0)
  nchan:          64  (0)
  ncorrelations:  4
  nbaselines:     1953
  ntimes:         7190
  time interval:  1.00139
  DATA column:    DATA
  WEIGHT column:  WEIGHT_SPECTRUM
  autoweight:     true
MSWriter msout.
  output MS:      /data/scratch/broderick/lofar_school/T1/L456104_SAP000_SB001_uv_copy.MS
  nchan:          64
  ncorrelations:  4
  nbaselines:     1953
  ntimes:         7190
  time interval:  1.00139
  DATA column:    DATA
  WEIGHT column:  WEIGHT_SPECTRUM

Processing 7190 time slots ...

0%....10....20....30....40....50....60....70....80....90....100%
Finishing processing ...

NaN/infinite data flagged in reader
===================================

Percentage of flagged visibilities detected per correlation:
  [0,0,0,0] out of 898692480 visibilities   [0%, 0%, 0%, 0%]
0 missing time slots were inserted



Total NDPPP time     268.64 real      202.58 user       45.62 system
   59.2% MSReader
   40.8% MSWriter msout.
```

N.B. only do this once at this stage!!
Otherwise always = F (default)

This simple parset also automatically removes clearly bad data if necessary (not here, though).

LOFAR ASTRON

article | edit this page | old revisions | manage subscriptions | export: pdf

navigation
Wiki Startpage

Visitor Sections:

- User and LTA access
- Meetings, Workshops, Reports, docs
- LOFAR Science
- International partners
- User Software
- Important dates (stopdays, rollouts)

Projects:

- Imaging Tiger Team
- CEP3 cluster
- CEP4 cluster
- RRR project
- Dragnet project
- LCU update project

Local Sections:

NDPPP

NDPPP is the LOFAR data pipelined processing program. It can be used to do all kind of operations on the data in a pipelined way, so the data are read and written only once.

NDPPP started as a new and faster version of IDPPP. The original differences can be seen here.

NDPPP preprocesses the data of a LOFAR observation by executing steps like flagging or averaging. Such steps can be used for the raw data as well as the calibrated data by defining the data column to use. One or more of the following steps can be defined as a pipeline. NDPPP has an implicit input and output step. It is also possible to have intermediate output steps.

NDPPP comes with quite some predefined steps, but it is possible to plugin arbitrary steps, either implemented in C++ or Python.

The following steps are possible:

- **Flagging** and **Filtering**
  - **AOFlagger** for automatic flagging in time/freq windows using Andre Offringa's advanced aoflagger.
  - **Preflagger** to flag given baselines, time slots, etc.
  - **UVWFlagger** to flag based on UVW coordinates, possibly in the direction of another source.
  - **MADFlagger** for automatic flagging in time/freq windows based on median filtering.
  - **Filter** to filter on baseline and/or channel (only the given baselines/channels are kept). The reader step has an implicit filter.
- **Averaging**
  - **Averager** to average data in time and/or freq.
- **Phase Shifting**
  - **PhaseShift** to shift data to another phase center.
- **Demixing** to remove strong sources (A-team) from the data.

Table of Contents –

http://www.lofar.org/wiki/doku.php?id=public:user_software:ndppp

* DPPP.log on Tutorial slide 7 – 98% flagged is the max. because chan 0 already flagged by default. General RFI occupancy quite low (~few per cent).

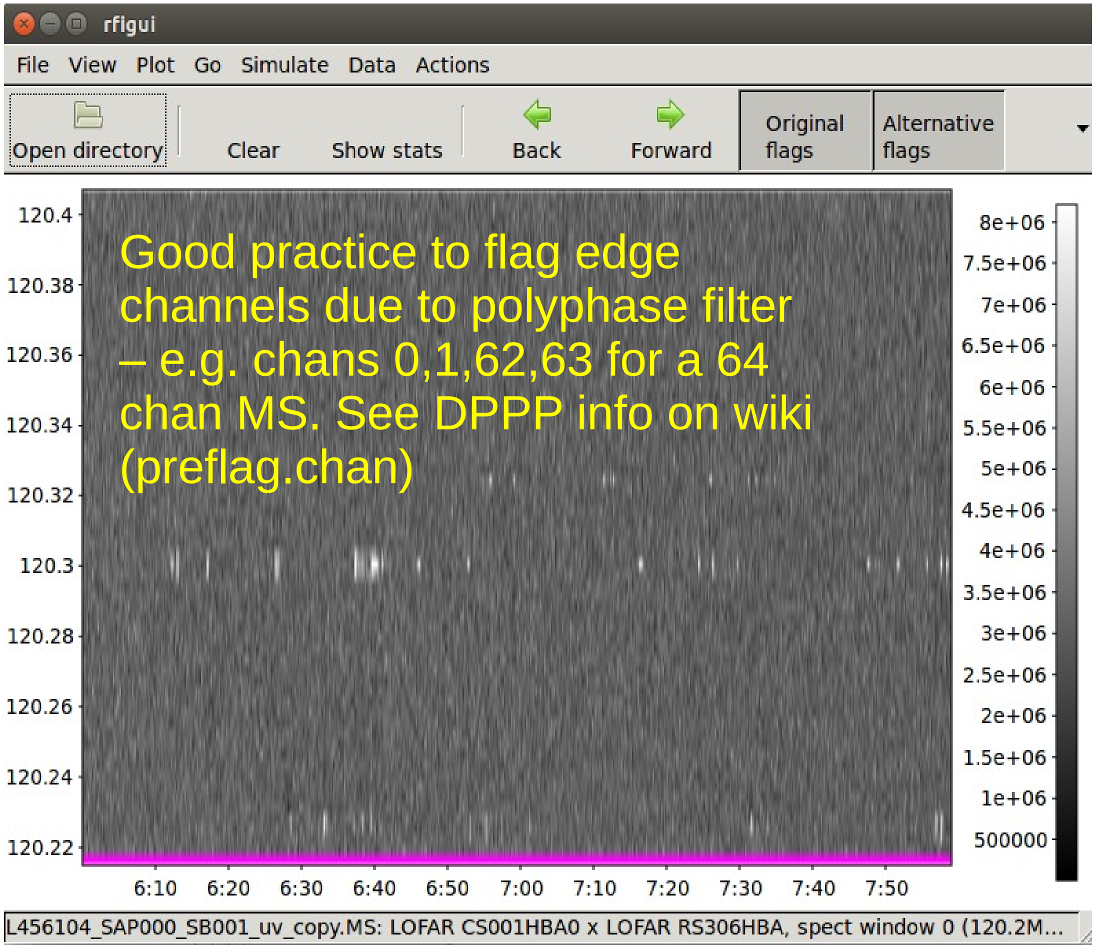


Table 1. Short list of allocated frequencies in The Netherlands in the range 10–250 MHz (source: Agentschap Telecom).

Service type	Frequency range(s) in MHz
Time signal	10, 15, 20
Air traffic	10–22, 118–137, 138–144
Short-wave radio broadcasting	11–26
Military, maritime, mobile	12–26, 27–61, 68–88, 138–179
Amateur	14, 50–52, 144–146
CB radio	27-28
Modelling control	27–30, 35, 40–41
Microphones	36–38, 173–175
Radio astronomy	13, 26, 38, 150–153
Baby monitor (portophone)	39–40
Broadcasting	61–88
Emergency	74, 169–170
Air navigation	75, 108–118
FM radio	87–108
Satellites	137–138, 148–150
Navigation	150
Remote control	154
T-DAB	174–230
Intercom	202–209

* Core-remote baseline – extended emission from target field resolved out (cf. tutorial slide 6). But RFI remains in this case...

* Offringa et al. 2013, A&A, 549, A11
NOTE: Significant RFI from DAB inter-modulation products above 150 MHz! See LOFAR technical webpages.

rfigui - also see http://www.astro.rug.nl/rfi-software/gui-tutorial.html

* The **Drawer** - http://www.lofar.org/wiki/lib/exe/fetch.php?media=commissioning:pres_drawer.pdf

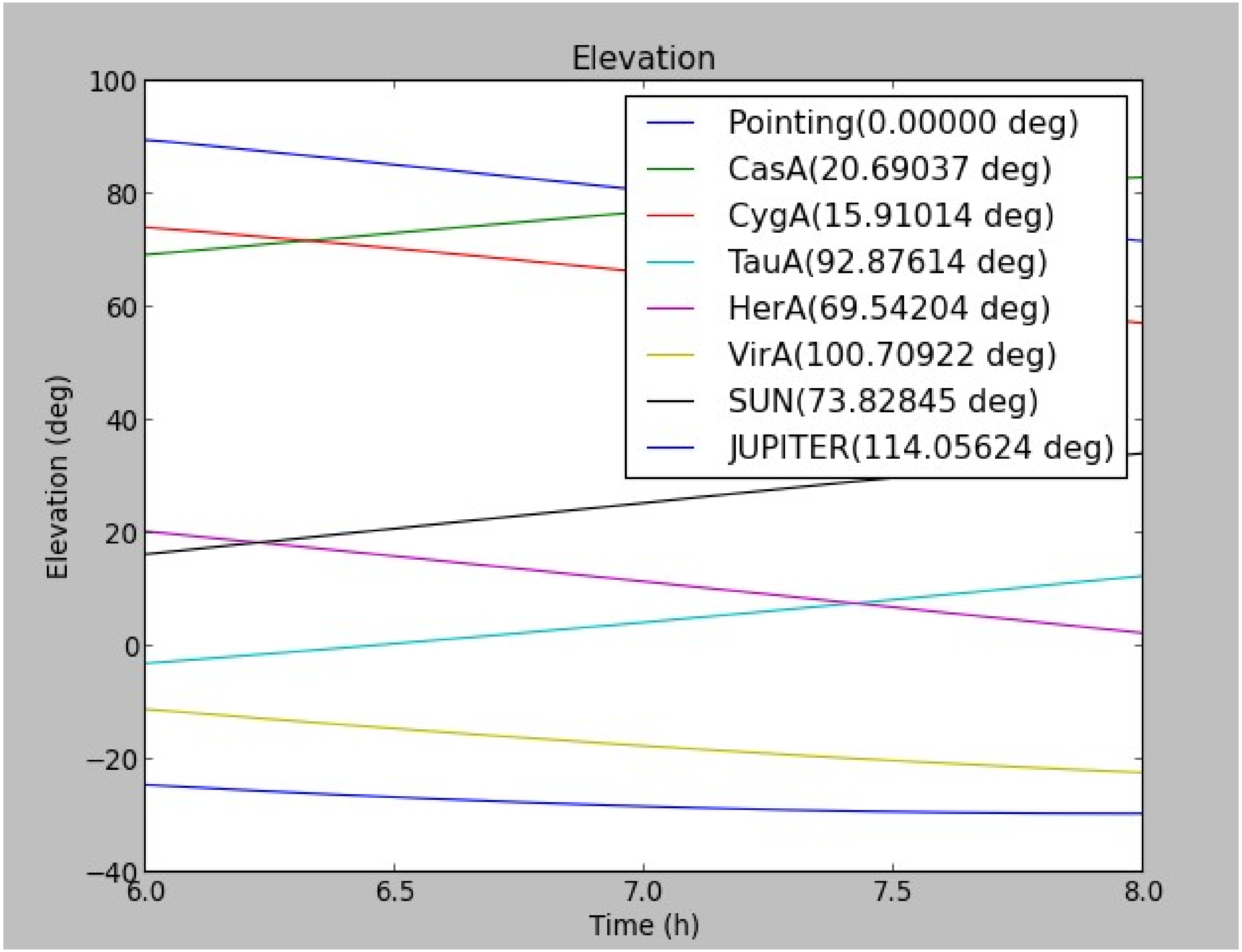
* Demixing; usually default settings are fine. Look out for 'smart demixing' from LOFAR Cycle 8 onwards. Demixing paper: http://adsabs.harvard.edu/abs/2007ITSP...55.4497V (van der Tol et al. 2007)

* Worth putting in the time to make sure that you have some ideas of the possible effects of the A-team on your data (and also in terms of the processing / observing time ratio). In the LBA you demix both Cyg A and Cas A by default, because the antenna elements can see the whole sky accessible from the LOFAR site. In the HBA, simulations can really help, but you are very likely to need demixing if an A-team source is less than about 20 deg from your target.

* Be a little bit careful with demix ignoretarget = F/T. F is generally best (and the default). If in doubt contact Science Support.

* Data averaging: what are your science goals? How much bandwidth/time smearing can you tolerate? How often should you be solving for the gain phases? Do you need to transfer large amounts of data to an external cluster first?
Useful links:
http://adsabs.harvard.edu/abs/1999ASPC..180..371B (Bridle & Schwab 1999)
http://adsabs.harvard.edu/abs/2015A%26A...582A.123H (Heald et al. 2015; LOFAR MSSS)

One recommendation is to keep the time resolution to 10s maximum after averaging, and the minimum number of channels to 4 per sub-band (and quite possibly 8). Native time/frequency resolutions for the actual observations are typically 1 or 2s, and 64 chans/sub-band.



plot_Ateam_elevation.py output

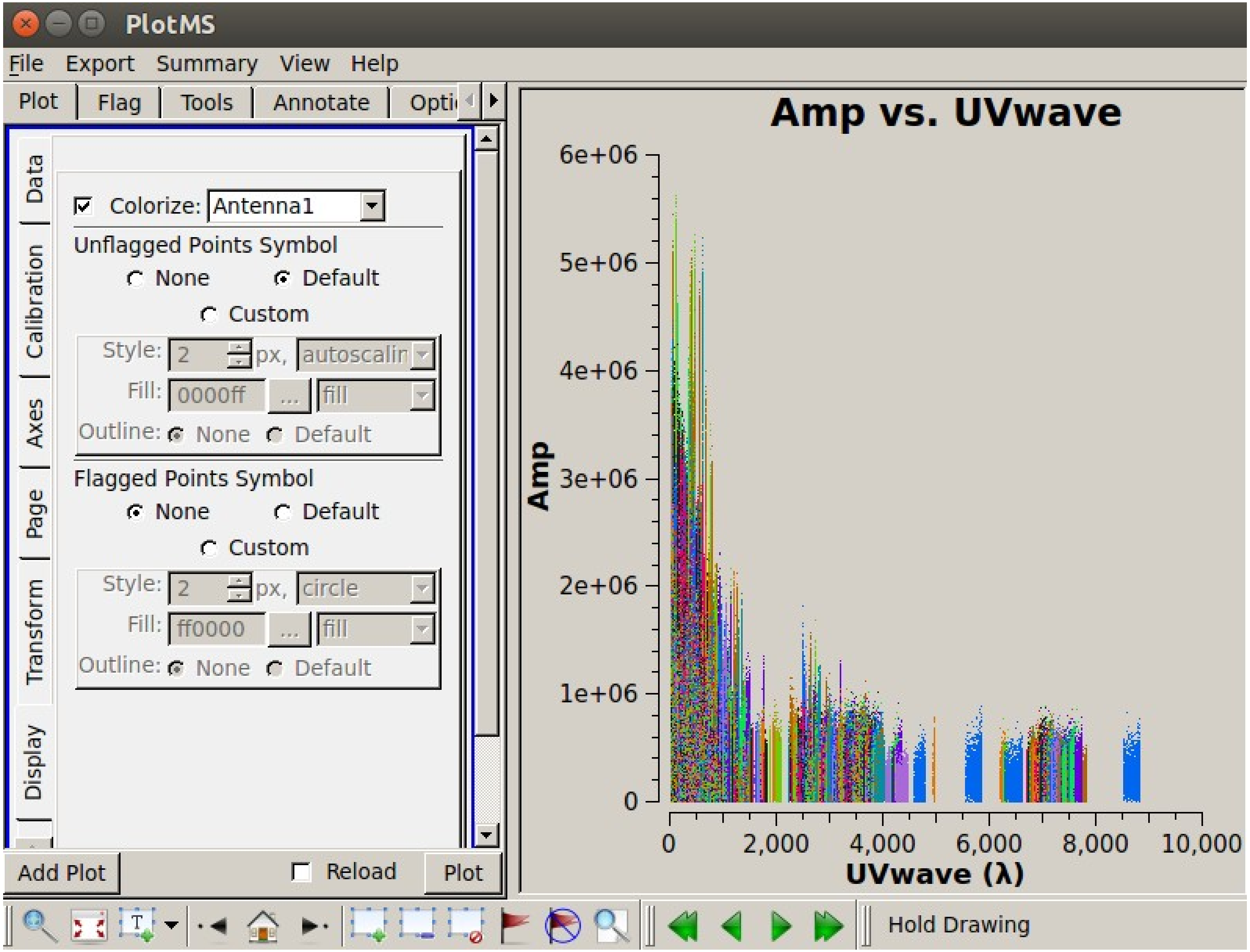
* Scripting – learning Python is a highly valuable skill!

* **Important**: <step>.type in **DPPP** - this defaults to the standard process name. But if you e.g. had custom step names like the following:

steps = [process1, process2, process3]

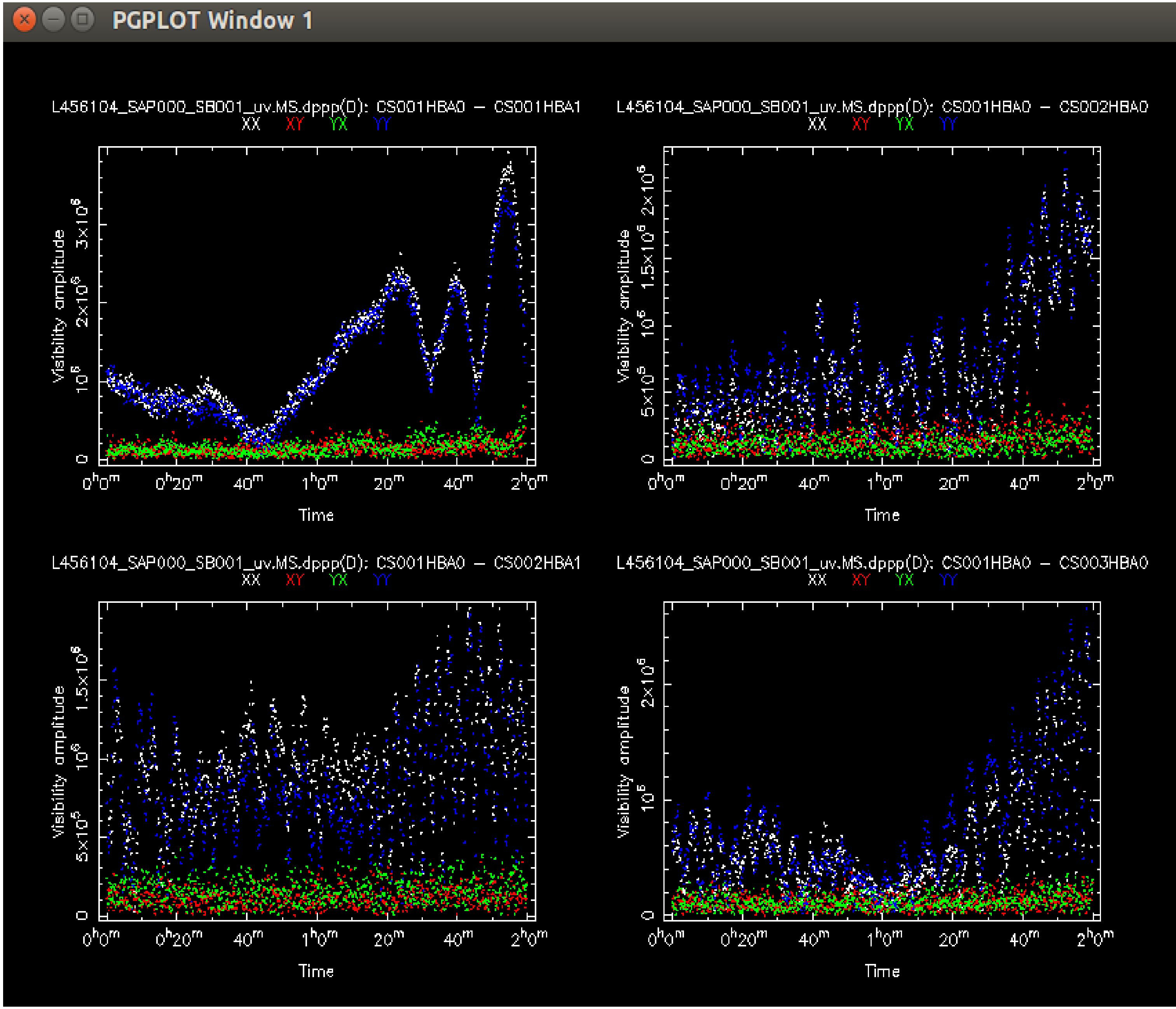
Then your parset needs extra lines such as
process1.type=preflagger
process2.type=aoflagger
and so on.... See the LOFAR wiki for more information. And the example below.

```
broderick@lof012:/data/scratch/broderick/lofar_school/T1$ more DPPP-all2.parset
msin=L456104_SAP000_SB001_uv.MS
msin.autoweight=true
msout=L456104_SAP000_SB001_uv.MS.dppp
steps=[lhee,dwingeloo,exloo]
lhee.type=preflagger
lhee.baseline=*&&&;RS208HBA;RS210HBA;RS310HBA;RS406HBA;RS407HBA;RS409HBA;RS508HBA;RS509HBA;RS205HBA&RS
307HBA
dwingeloo.type=aoflagger
exloo.type=averager
exloo.timestep=10
exloo.freqstep=16
```



Not a surprise that amplitudes much higher on short baselines: Galactic Plane field, extended emission.
Redo plot after calibration.

https://casa.nrao.edu/docs/taskref/plotms-task.html



Can be much faster than **casaplotms**!

> uvplot.py -h
> uvplot.py - L456104_SAP000_SB001_uv.MS.dppp -n 2