JIVE
Joint Institute for VLBI
ERIC

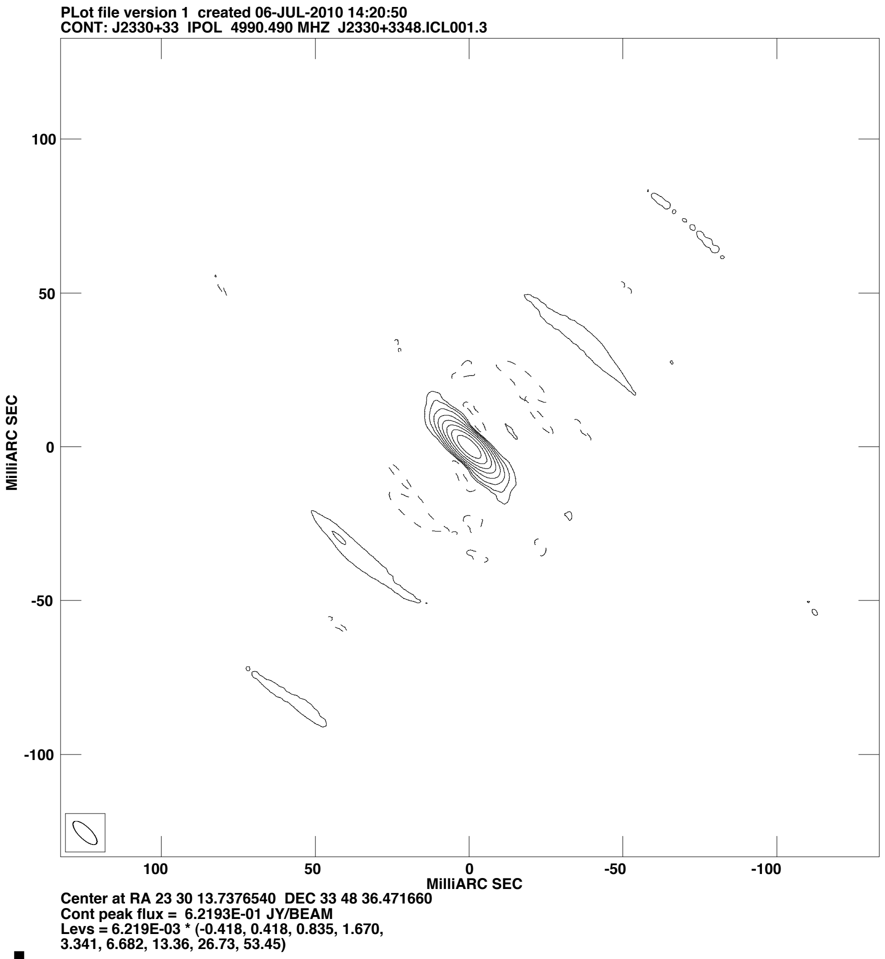
A Priori VLBI Calibration in CASA

Mark Kettenis
JIVE

EVN vs. VLBA

- VLBA: Homogeneous array, centrally controlled by NRAO
- EVN: Inhomogeneous array, loose collaboration supported by JIVE
- Model for EHT/GMVA?

EVN Pypeline



- ParselTongue/AIPS
- Focus on calibration
- Outputs lots of plots and crude images
- Written by Cormac Reynolds
- Derivatives used by eMERLIN, LBA, VERA

Pypeline Steps

1. Load and sort the data
2. A priori data flagging
3. Plot the raw data
4. Amplitude calibration and parallectic angle correction
5. Fringe fitting
6. Bandpass calibration

Load and sort the data

- FITS-IDI data format
- AIPS: FITLD
- CASA: importfitsidi
  - Creates MeasurementSet
  - Minor issues with observation/scan IDs

A priori data flagging

- AIPS: UVFLG
- CASA: flagdata
- flag.py: script to convert UVFLG input files into flagdata input files

Amplitude calibration

- AIPS: ANTAB and APCAL
- CASA: gencal
- Convert to physical units
- Compensate for antenna deformation

Tsys measurements

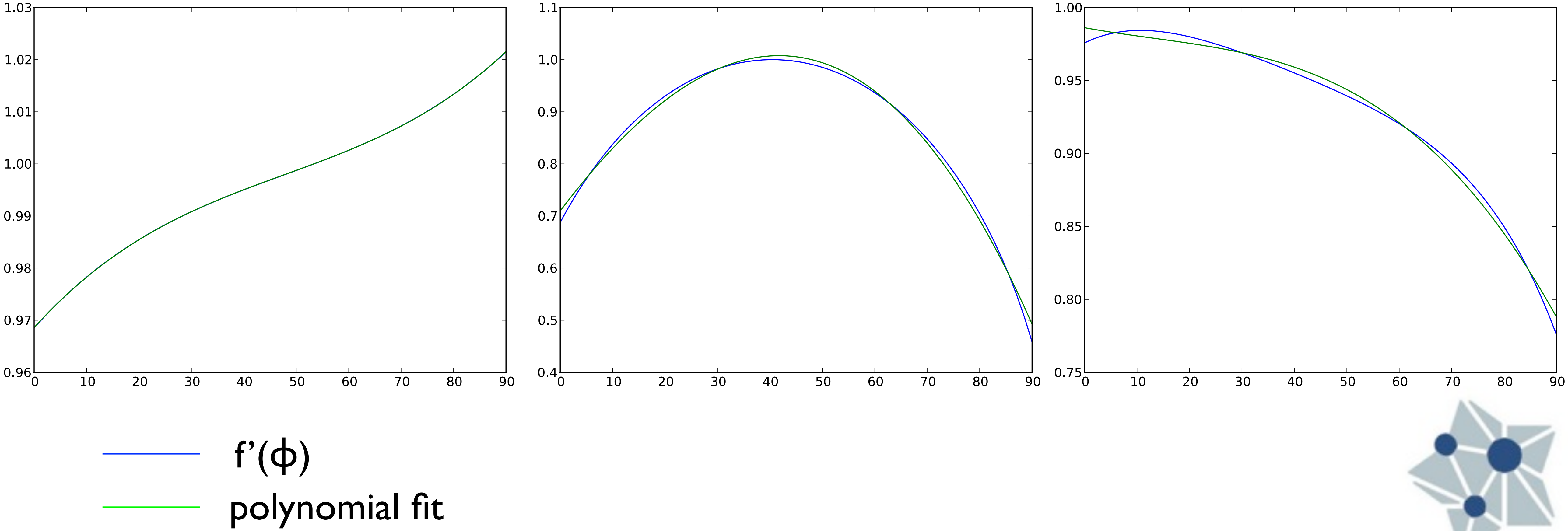
- AIPS: ANTAB
- CASA: gencal(caltype='tsys')
- MeasurementSet SYSCAL table:
  - mandatory TSYS column
  - optional TSYS_SPECTRUM column
- gencal task only uses TSYS_SPECTRUM
- expects synchronous measurements for antennas
- tsys.py script reads ANTAB file inter/extrapolates

Gain curves

- VLBI: DPFU (degrees per flux unit) × polynomial as function of elevation/zenith angle
- VLA: √polynomial of zenith angle × efficiency
- AIPS: ANTAB
- CASA: gencal(type='gc')
    - No support in MeasurementSet
    - Internal tables with VLA-type gain curves
- gc.py script converts ANTAB to VLA-type gain curves
- gencal modified to take external table

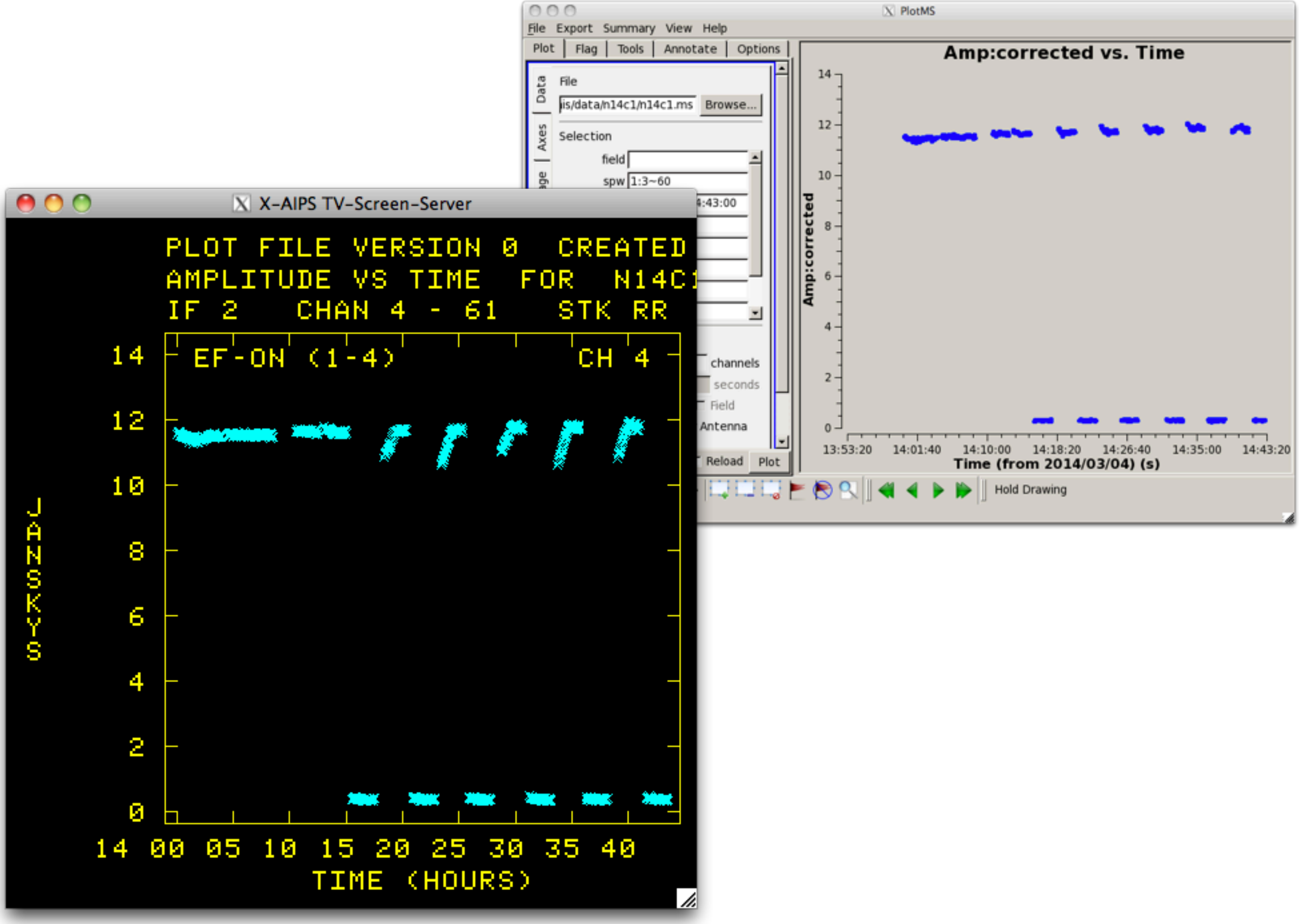
Gain curves

Third order polynomial fit of f’(ϕ) = √f(90-ϕ)


Ef

Jb

On



File
jis/data/n14c1/n14c1.ms

Selection
field
spw 1:3~60
:43:00

Parallectic angle conversion

- AIPS: PANG
- CASA: parang parameter
    - Needs to be set for all invocations?

Fringe fitting

- AIPS: FRINGE
- CASA can only solve for simple antenna-based delays (‘K’ solutions)
    - Only delays, no rates
    - Insufficient for VLBI
- See presentation by Des Small

Bandpass Calibration

- AIPS: BPASS
- CASA: bandpass
- Solutions found for strong calibrators, except for longer baselines. Needs fringe fit?

Calibration script

```
importfitsidi(vis='n14c1.ms', fitsidifile=[
        '/scratch/kettenis/data/n14c1/n14c1_1_1.IDI1',
        '/scratch/kettenis/data/n14c1/n14c1_1_1.IDI2’)

execfile('tsys.py')
gencal(vis='n14c1.ms', caltable='n14c1.tsys', caltype='tsys')
execfile('gc.py')
gencal(vis='n14c1.ms', caltype='gc', caltable='n14c1.gain',
        infile='EVN.gc')

applycal(vis='n14c1.ms', gaintable=['n14c1.tsys', 'n14c1.gain'])
plotms(vis='n14c1.ms', spw='1:3~60', antenna='EF&ON',
        correlation='RR', avgchannel='64', ydatacolumn='corrected')
```

Additional calibration?

- Opacity corrections
  - Ties in with gain curves?
- Digital corrections
  - Somehwhat correlator dependent
- Pulse-cal
- WVR

Gripes

- Creating CORRECTED_DATA column takes ridiculously long.
- Logger GUI slow when using remotely