Phase and Amplitude Calibration
in CASA for ALMA data

Adam Leroy

North American ALMA Science Center

Atacama Large Millimeter/submillimeter Array
Expanded Very Large Array
Robert C. Byrd Green Bank Telescope
Very Long Baseline Array

Outline

- **Basic Idea and Workflow (Review)**
- Gaincal
- Phase and Amplitude Calibration for NGC 3256
- Before and After for NGC 3256

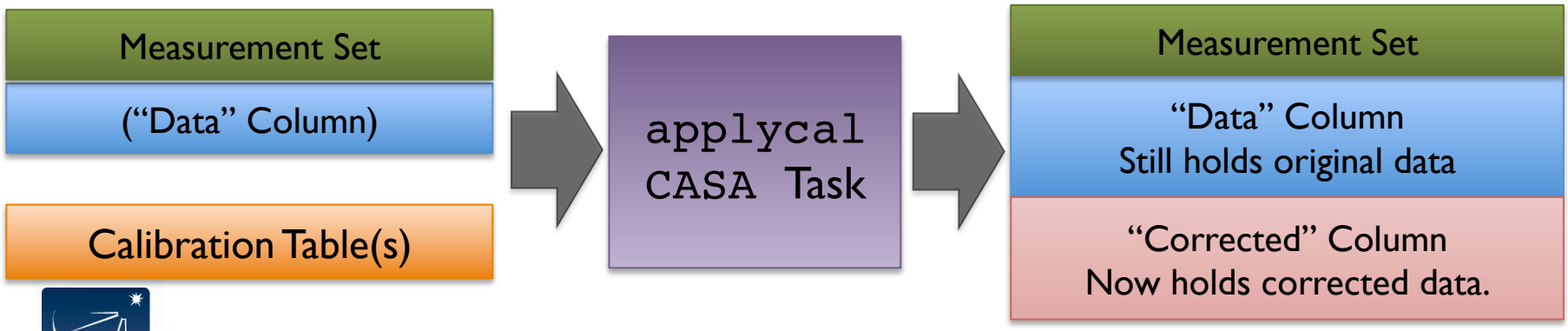
Calibration Tables

- Calibration yields estimates of phase and amplitude corrections.
  E.G., AS A FUNCTION OF TELESCOPE, TIME, FREQUENCY, POLARIZATION.

- CASA stores these corrections in directories called "calibration tables."
  TO DELETE THEM USE `rmtables`

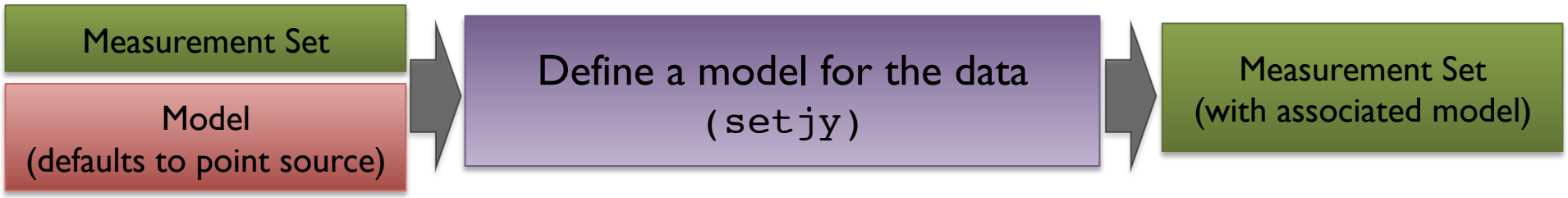
- These are created by calibration tasks:
  E.G., `gaincal, bandpass, gencal`

- Applied via "`applycal`" to the data column and saved as corrected.

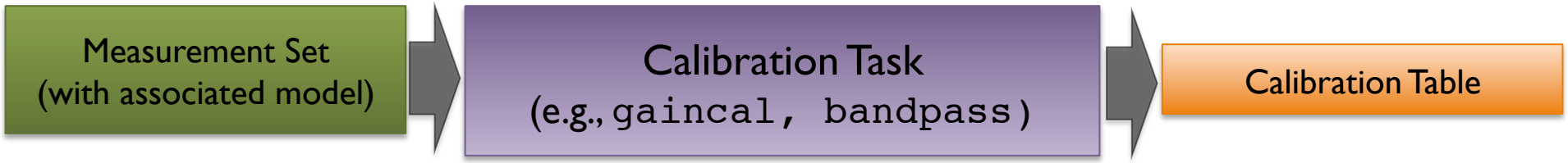


Reminder: Basic Calibration Flow

Define what the telescope SHOULD have seen.



Derive the corrections needed to make the data match the model.



Apply these corrections to derive the corrected (calibrated) data.

Schematic Calibration

DONE

Calibrate the Amplitude and Phase vs. Frequency of Each Antenna

ASSUME TIME & FREQUENCY RESPONSE SEPARABLE, REMOVE TIME VARIABILITY

DONE

Calibrate the Amplitude and Phase vs. Time of Each Antenna

ASSUME TIME & FREQUENCY RESPONSE SEPARABLE, REMOVE FREQ. VARIABILITY

Set the Absolute Amplitude Scale With Reference to a Known Source

PLANET (MODELLED), MONITORED QUASAR, ETC.

Apply all corrections to produce calibrated data

Schematic Calibration

DONE

Calibrate the Amplitude and Phase vs. Frequency of Each Antenna
`bandpass`

DONE

Bandpass Calibration Table

Calibrate the Amplitude and Phase vs. Time of Each Antenna
`gaincal`

Phase Calibration Table
Amplitude Calibration Table

Set the Absolute Amplitude Scale With Reference to a Known Source
`fluxscale`

Flux Calibration Table

Apply all corrections to produce calibrated data
`applycal`

Measurement Set

Corrected column now holds calibrated data.

Outline

- Basic Idea and Workflow (Review)
- **Gaincal**
- Phase and Amplitude Calibration for NGC 3256
- Before and After for NGC 3256

Key Tasks for Calibration

Derive Calibration Tables

- `setjy`: set "model" (correct) visibilities using known model for a calibrator
- `bandpass`: calculate bandpass calibration table (amp/phase vs frequency)
- `gaincal`: calculate temporal gain calibration table (amp/phase vs time)
- `fluxscale`: apply absolute flux scaling to calibration table from known source

Manipulate Your Measurement Set

- `flagdata`/`flagcmd`/`flagmanager`: flag (remove) bad data
- `applycal`: apply calibration table(s) from previous steps
- `split`: split off calibrated data from your ms (for imaging!)

Inspect Your Data and Results

- `plotms`: inspect your data interactively
- `plotcal`: examine a calibration table

gaincal

Measurement Set

Data column holds observations.

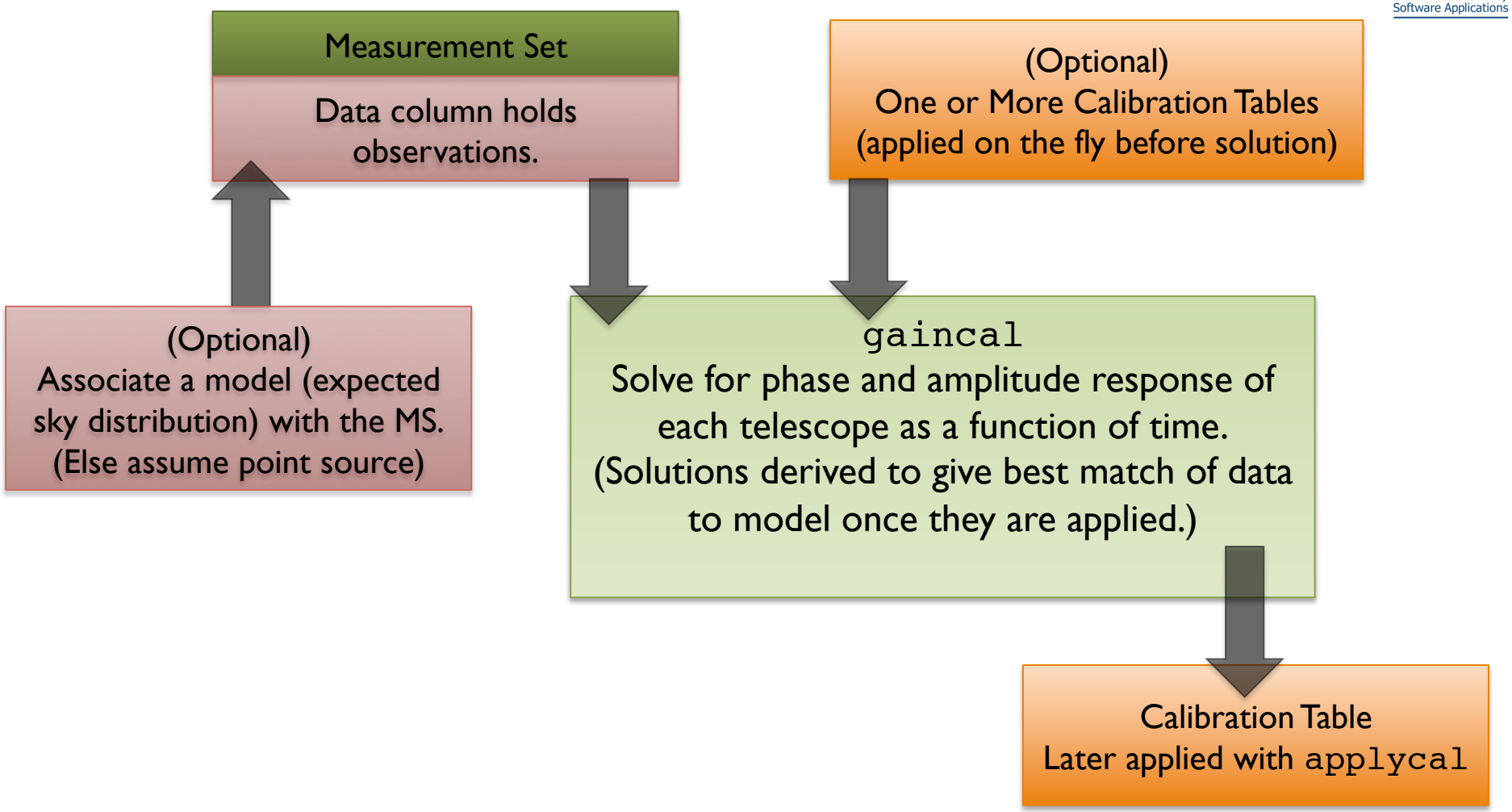
`gaincal`

Solve for phase and amplitude response of each telescope as a function of time. (Solutions derived to give best match of data to model once they are applied.)

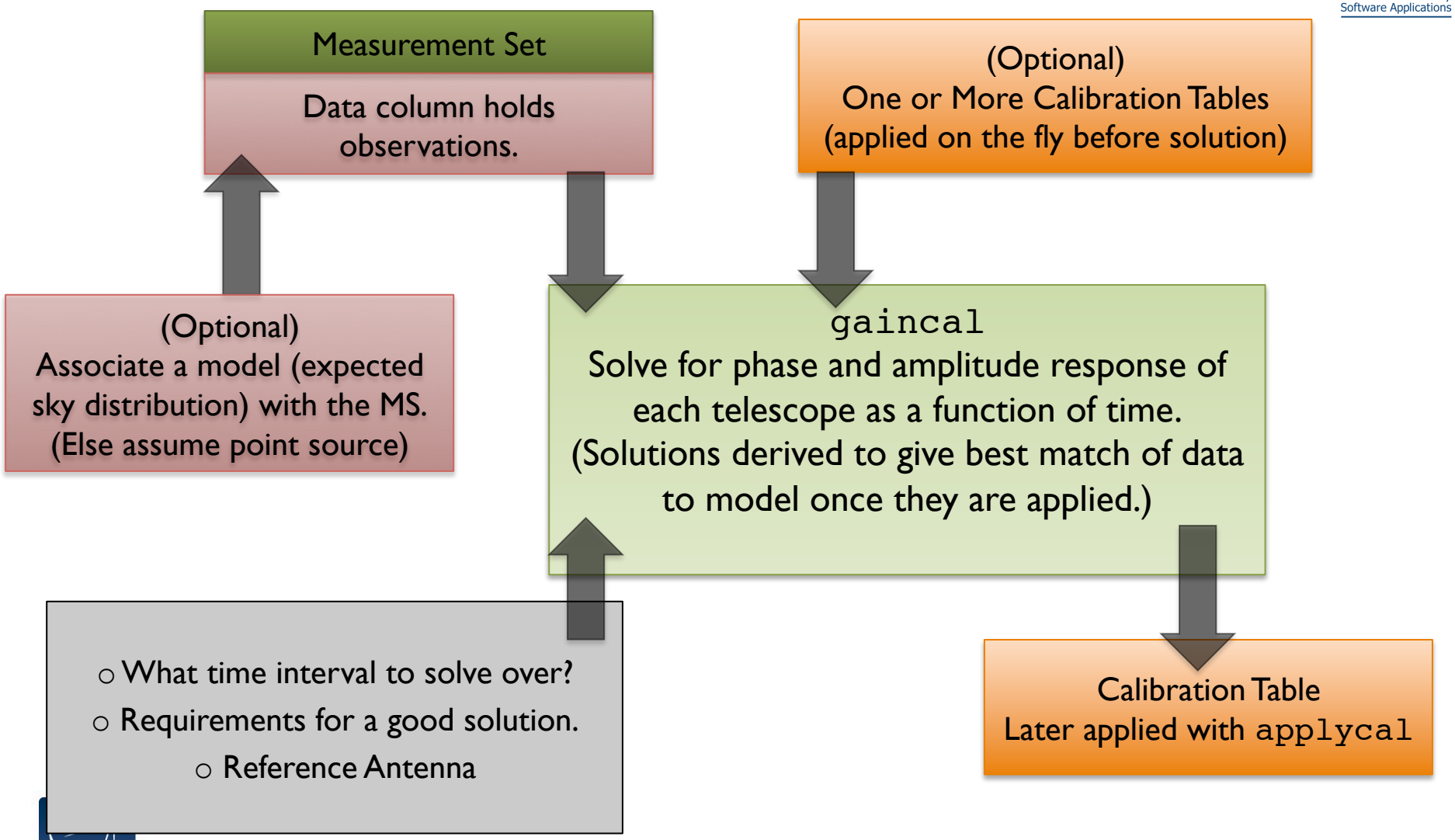
Calibration Table

Later applied with `applycal`

gaincal


Measurement Set
Data column holds observations.
(Optional)
One or More Calibration Tables
(applied on the fly before solution)
(Optional)
Associate a model (expected sky distribution) with the MS.
(Else assume point source)
gaincal
Solve for phase and amplitude response of each telescope as a function of time.
(Solutions derived to give best match of data to model once they are applied.)
Calibration Table
Later applied with applycal

gaincal


Measurement Set
Data column holds observations.
(Optional)
One or More Calibration Tables
(applied on the fly before solution)
(Optional)
Associate a model (expected sky distribution) with the MS.
(Else assume point source)
gaincal
Solve for phase and amplitude response of each telescope as a function of time.
(Solutions derived to give best match of data to model once they are applied.)
○ What time interval to solve over?
○ Requirements for a good solution.
○ Reference Antenna
Calibration Table
Later applied with applycal

gaincal

```
-------- inp(gaincal)
#  gaincal :: Determine temporal gains from calibrator observations
vis                 =         ''        #  Name of input visibility file
caltable            =         ''        #  Name of output gain calibration table
field               =         ''        #  Select field using field id(s) or field name(s)
spw                 =         ''        #  Select spectral window/channels
intent              =         ''        #  Select observing intent
selectdata          =      False        #  Other data selection parameters
solint              =      'inf'        #  Solution interval: egs. 'inf', '60s' (see help)
combine             =         ''        #  Data axes which to combine for solve (scan, spw,
                                        #   and/or field)
preavg              =       -1.0        #  Pre-averaging interval (sec) (rarely needed)
refant              =         ''        #  Reference antenna name(s)
minblperant         =          4        #  Minimum baselines _per antenna_ required for solve
minsnr              =        3.0        #  Reject solutions below this SNR
solnorm             =      False        #  Normalize average solution amplitudes to 1.0 (G, T
                                        #   only)
gaintype            =        'G'        #  Type of gain solution (G,T,GSPLINE,K,KCROSS)
smodel              =         []        #  Point source Stokes parameters for source model.
calmode             =       'ap'        #  Type of solution: ('ap', 'p', 'a')
append              =      False        #  Append solutions to the (existing) table
gaintable           =       ['']        #  Gain calibration table(s) to apply on the fly
gainfield           =       ['']        #  Select a subset of calibrators from gaintable(s)
interp              =       ['']        #  Temporal interpolation for each gaintable (=linear)
spwmap              =         []        #  Spectral windows combinations to form for
                                        #   gaintables(s)
gaincurve           =      False        #  Apply internal VLA antenna gain curve correction
opacity             =         []        #  Opacity correction to apply (nepers), per spw
parang              =      False        #  Apply parallactic angle correction on the fly
async               =      False        #  If true the taskname must be started using
                                        #   gaincal(...)
```

gaincal

```
--------) inp(gaincal)
#  gaincal :: Determine temporal gains from calibrator observations
vis                 =          ''        #  Name of input visibility file
caltable            =          ''        #  Name of output gain calibration table
field               =          ''        #  Select
spw                 =          ''        #  Select
intent              =          ''        #  Sele
selectdata          =       False        #  Other
solint              =       'inf'        #  Solutio
combine             =          ''        #  Data a
                                         #   and/or field)
preavg              =        -1.0        #  Pre-averaging interval (sec) (rarely needed)
refant              =          ''        #  Reference antenna name(s)
minblperant         =           4        #  Minimum baselines _per antenna_ required for solve
minsnr              =         3.0        #  Reject solutions below this SNR
solnorm             =       False        #  Normalize average solution amplitudes to 1.0 (G, T
                                         #   only)
gaintype            =         'G'        #  Type of gain solution (G,T,GSPLINE,K,KCROSS)
smodel              =          []        #  Point source Stokes parameters for source model.
calmode             =        'ap'        #  Type of solution: ('ap', 'p', 'a')
append              =       False        #  Append solutions to the (existing) table
gaintable           =        ['']        #  Gain calibration table(s) to apply on the fly
gainfield           =        ['']        #  Select a subset of calibrators from gaintable(s)
interp              =        ['']        #  Temporal interpolation for each gaintable (=linear)
spwmap              =          []        #  Spectral windows combinations to form for
                                         #   gaintables(s)
gaincurve           =       False        #  Apply internal VLA antenna gain curve correction
opacity             =          []        #  Opacity correction to apply (nepers), per spw
parang              =       False        #  Apply parallactic angle correction on the fly
async               =       False        #  If true the taskname must be started using
                                         #   gaincal(...)
```

Input Measurement Set
(with model set, if needed)

gaincal

```
-------- inp(gaincal)
#  gaincal :: Determine temporal gains from calibrator observations
vis                 =          ''          #  Name of input visibility file
caltable            =          ''          #  Name of output gain calibration table
field               =          ''          #  Select
spw                 =          ''          #  Select
intent              =          ''          #  Sel
selectdata          =       False          #  Other
solint              =       'inf'          #  Solutio
combine             =          ''          #  Data a
                                           #   and/or field)
preavg              =        -1.0          #  Pre-averaging interval (sec) (rarely needed)
refant              =          ''          #  Reference antenna name(s)
minblperant         =           4          #  Minimum baselines _per antenna_ required for solve
minsnr              =         3.0          #  Reject solutions below this SNR
solnorm             =       False          #  Normalize average solution amplitudes to 1.0 (G, T
                                           #   only)
gaintype            =         'G'          #  Type of gain solution (G,T,GSPLINE,K,KCROSS)
smodel              =          []          #  Point source Stokes parameters for source model.
calmode             =        'ap'          #  Type of solution: ('ap', 'p', 'a')
append              =       False          #  Append solutions to the (existing) table
gaintable           =        ['']          #  Gain calibration table(s) to apply on the fly
gainfield           =        ['']          #  Select a subset of calibrators from gaintable(s)
interp              =        ['']          #  Temporal interpolation for each gaintable (=linear)
spwmap              =          []          #  Spectral windows combinations to form for
                                           #   gaintables(s)
gaincurve           =       False          #  Apply internal VLA antenna gain curve correction
opacity             =          []          #  Opacity correction to apply (nepers), per spw
parang              =       False          #  Apply parallactic angle correction on the fly
async               =       False          #  If true the taskname must be started using
                                           #   gaincal(...)
```

Output Calibration Table

(apply later with `applycal`)

gaincal

```
--------> inp(gaincal)
#  gaincal :: Determine temporal gains from calibrator observations
vis                 =          ''        #  Name of input visibility file
caltable            =          ''        #  Name of output gain calibration table
field               =          ''        #  Select
spw                 =          ''        #  Select
intent              =          ''        #  Select
selectdata          =       False        #  Other
solint              =       'inf'        #  solutio
combine             =          ''        #  Data a
                                         #   and/or
preavg              =        -1.0        #  Pre-ave
refant              =          ''        #  Referen
minblperant         =           4        #  Minimum
minsnr              =         3.0        #  Reject solutions below this SNR
solnorm             =       False        #  Normalize average solution amplitudes to 1.0 (G, T
                                         #   only)
gaintype            =         'G'        #  Type of gain solution (G,T,GSPLINE,K,KCROSS)
smodel              =          []        #  Point source Stokes parameters for source model.
calmode             =        'ap'        #  Type of solution: ('ap', 'p', 'a')
append              =       False        #  Append solutions to the (existing) table
gaintable           =        ['']        #  Gain calibration table(s) to apply on the fly
gainfield           =        ['']        #  Select a subset of calibrators from gaintable(s)
interp              =        ['']        #  Temporal interpolation for each gaintable (=linear)
spwmap              =          []        #  Spectral windows combinations to form for
                                         #   gaintables(s)
gaincurve           =       False        #  Apply internal VLA antenna gain curve correction
opacity             =          []        #  Opacity correction to apply (nepers), per spw
parang              =       False        #  Apply parallactic angle correction on the fly
async               =       False        #  If true the taskname must be started using
                                         #   gaincal(...)
```

Options to select which data to consider:

e.g., select calibrator fields

gaincal

```
--------> inp(gaincal)
#  gaincal :: Determine temporal gains from calibrator observations
vis                 =          ''        #  Name of input visibility file
caltable            =          ''        #  Name of output gain calibration table
field               =          ''        #  Select
spw                 =          ''        #  Select
intent              =          ''        #  Select
selectdata          =       False        #  Other
solint              =       'inf'        #  Solutio
combine             =          ''        #  Data a
                                         #   and/or
preavg              =        -1.0        #  Pre-ave
refant              =          ''        #  Referen
minblperant         =           4        #  Minimum
minsnr              =         3.0        #  Reject
solnorm             =       False        #  Normalize average solution amplitudes to 1.0 (G, T
                                         #   only)
gaintype            =         'G'        #  Type of gain solution (G,T,GSPLINE,K,KCROSS)
smodel              =          []        #  Point source Stokes parameters for source model.
calmode             =        'ap'        #  Type of solution: ('ap', 'p', 'a')
append              =       False        #  Append solutions to the (existing) table
gaintable           =        ['']        #  Gain calibration table(s) to apply on the fly
gainfield           =        ['']        #  Select a subset of calibrators from gaintable(s)
interp              =        ['']        #  Temporal interpolation for each gaintable (=linear)
spwmap              =          []        #  Spectral windows combinations to form for
                                         #   gaintables(s)
gaincurve           =       False        #  Apply internal VLA antenna gain curve correction
opacity             =          []        #  Opacity correction to apply (nepers), per spw
parang              =       False        #  Apply parallactic angle correction on the fly
async               =       False        #  If true the taskname must be started using
                                         #   gaincal(...)
```

Time interval over which to solve.

(Only cross scan or spw boundaries with "combine")

gaincal

```
--------) inp(gaincal)
#  gaincal :: Determine temporal gains from calibrator observations
vis                 =          ''        #  Name of input visibility file
caltable            =          ''        #  Name of output gain calibration table
field               =          ''        #  Select
spw                 =          ''        #  Select
intent              =          ''        #  Select
selectdata          =       False        #  Other
solint              =       'inf'        #  Solutio
combine             =          ''        #  Data
                                         #  and/or
preavg              =        -1.0        #  Pre-ave
refant              =          ''        #  Referen
minblperant         =           4        #  Minimum
minsnr              =         3.0        #  Reject
solnorm             =       False        #  Normalize average solution amplitudes to 1.0 (G, T
                                         #   only)
gaintype            =         'G'        #  Type of gain solution (G,T,GSPLINE,K,KCROSS)
smodel              =          []        #  Point source Stokes parameters for source model.
calmode             =        'ap'        #  Type of solution: ('ap', 'p', 'a')
append              =       False        #  Append solutions to the (existing) table
gaintable           =        ['']        #  Gain calibration table(s) to apply on the fly
gainfield           =        ['']        #  Select a subset of calibrators from gaintable(s)
interp              =        ['']        #  Temporal interpolation for each gaintable (=linear)
spwmap              =          []        #  Spectral windows combinations to form for
                                         #   gaintables(s)
gaincurve           =       False        #  Apply internal VLA antenna gain curve correction
opacity             =          []        #  Opacity correction to apply (nepers), per spw
parang              =       False        #  Apply parallactic angle correction on the fly
async               =       False        #  If true the taskname must be started using
                                         #   gaincal(...)
```

Reference Antenna

(pick a central one with little or no flagging)

gaincal

```
--------> inp(gaincal)
#  gaincal :: Determine temporal gains from calibrator observations
vis                 =          ''        #  Name of input visibility file
caltable            =          ''        #  Name of output gain calibration table
field               =          ''        #  Select
spw                 =          ''        #  Select
intent              =          ''        #  Select
selectdata          =       False        #  Other
solint              =       'inf'        #  Solutio
combine             =          ''        #  Data
                                         #  and/or
preavg              =        -1.0        #  Pre-ave
refant              =          ''        #  Referen
minblperant         =           4        #  Minimum
minsnr              =         3.0        #  Reject solutions below this SNR
solnorm             =       False        #  Normalize average solution amplitudes to 1.0 (G, T
                                         #   only)
gaintype            =         'G'        #  Type of gain solution (G,T,GSPLINE,K,KCROSS)
smodel              =          []        #  Point source Stokes parameters for source model.
calmode             =        'ap'        #  Type of solution: ('ap', 'p', 'a')
append              =       False        #  Append solutions to the (existing) table
gaintable           =        ['']        #  Gain calibration table(s) to apply on the fly
gainfield           =        ['']        #  Select a subset of calibrators from gaintable(s)
interp              =        ['']        #  Temporal interpolation for each gaintable (=linear)
spwmap              =          []        #  Spectral windows combinations to form for
                                         #   gaintables(s)
gaincurve           =       False        #  Apply internal VLA antenna gain curve correction
opacity             =          []        #  Opacity correction to apply (nepers), per spw
parang              =       False        #  Apply parallactic angle correction on the fly
async               =       False        #  If true the taskname must be started using
                                         #   gaincal(...)
```

Requirements for a solution in terms of S/N and # of baselines contributing

gaincal

```
--------) inp(gaincal)
#  gaincal :: Determine temporal gains from calibrator observations
vis                 =          ''        #  Name of input visibility file
caltable            =          ''        #  Name of output gain calibration table
field               =          ''        #  Select
spw                 =          ''        #  Select
intent              =          ''        #  Select
selectdata          =       False        #  Other
solint              =       'inf'        #  Solutio
combine             =          ''        #  Data
                                         #   and/or
preavg              =        -1.0        #  Pre-ave
refant              =          ''        #  Referen
minblperant         =           4        #  Minimum
minsnr              =         3.0        #  Reject solutions below this SNR
solnorm             =       False        #  Normalize average solution amplitudes to 1.0 (G, T
                                         #   only)
gaintype            =         'G'        #  Type of gain solution (G,T,GSPLINE,K,KCROSS)
smodel              =          []        #  Point source Stokes parameters for source model.
calmode             =        'ap'        #  Type of solution: ('ap', 'p', 'a')
append              =       False        #  Append solutions to the (existing) table
gaintable           =        ['']        #  Gain calibration table(s) to apply on the fly
gainfield           =        ['']        #  Select a subset of calibrators from gaintable(s)
interp              =        ['']        #  Temporal interpolation for each gaintable (=linear)
spwmap              =          []        #  Spectral windows combinations to form for
                                         #   gaintables(s)
gaincurve           =       False        #  Apply internal VLA antenna gain curve correction
opacity             =          []        #  Opacity correction to apply (nepers), per spw
parang              =       False        #  Apply parallactic angle correction on the fly
async               =       False        #  If true the taskname must be started using
                                         #   gaincal(...)
```

Normalize solutions?

gaincal

```
--------> inp(gaincal)
#  gaincal :: Determine temporal gains from calibrator observations
vis                 =          ''        #  Name of input visibility file
caltable            =          ''        #  Name of output gain calibration table
field               =          ''        #  Select
spw                 =          ''        #  Select
intent              =          ''        #  Select
selectdata          =       False        #  Other
solint              =       'inf'        #  Solutio
combine             =          ''        #  Data a
                                         #   and/or
preavg              =        -1.0        #  Prea
refant              =          ''        #  Referen
minblperant         =           4        #  Minimum
minsnr              =         3.0        #  Reject
solnorm             =       False        #  Normalize average solution amplitudes to 1.0 (G, T
                                         #   only)
gaintype            =         'G'        #  Type of gain solution (G,T,GSPLINE,K,KCROSS)
smodel              =          []        #  Point source Stokes parameters for source model.
calmode             =        'ap'        #  Type of solution: ('ap', 'p', 'a')
append              =       False        #  Append solutions to the (existing) table
gaintable           =        ['']        #  Gain calibration table(s) to apply on the fly
gainfield           =        ['']        #  Select a subset of calibrators from gaintable(s)
interp              =        ['']        #  Temporal interpolation for each gaintable (=linear)
spwmap              =          []        #  Spectral windows combinations to form for
                                         #   gaintables(s)
gaincurve           =       False        #  Apply internal VLA antenna gain curve correction
opacity             =          []        #  Opacity correction to apply (nepers), per spw
parang              =       False        #  Apply parallactic angle correction on the fly
async               =       False        #  If true the taskname must be started using
                                         #   gaincal(...)
```

What to solve for?

'a'mplitude

'p'hase

'ap' - both

gaincal

```
--------> inp(gaincal)
#  gaincal :: Determine temporal gains from calibrator observations
vis                 =          ''        #  Name of input visibility file
caltable            =          ''        #  Name of output gain calibration table
field               =          ''        #  Select
spw                 =          ''        #  Select
intent              =          ''        #  Select
selectdata          =       False        #  Other
solint              =       'inf'        #  Solutio
combine             =          ''        #  Data a
                                         #   and/or
preavg              =        -1.0        #  Preave
refant              =          ''        #  Referen
minblperant         =           4        #  Minimum
minsnr              =         3.0        #  Reject solutions below this SNR
solnorm             =       False        #  Normalize average solution amplitudes to 1.0 (G, T
                                         #   only)
gaintype            =         'G'        #  Type of gain solution (G,T,GSPLINE,K,KCROSS)
smodel              =          []        #  Point source Stokes parameters for source model.
calmode             =        'ap'        #  Type of solution: ('ap', 'p', 'a')
append              =       False        #  Append solutions to the (existing) table
gaintable           =        ['']        #  Gain calibration table(s) to apply on the fly
gainfield           =        ['']        #  Select a subset of calibrators from gaintable(s)
interp              =        ['']        #  Temporal interpolation for each gaintable (=linear)
spwmap              =          []        #  Spectral windows combinations to form for
                                         #   gaintables(s)
gaincurve           =       False        #  Apply internal VLA antenna gain curve correction
opacity             =          []        #  Opacity correction to apply (nepers), per spw
parang              =       False        #  Apply parallactic angle correction on the fly
async               =       False        #  If true the taskname must be started using
                                         #   gaincal(...)
```

Calibration tables to apply before solution:

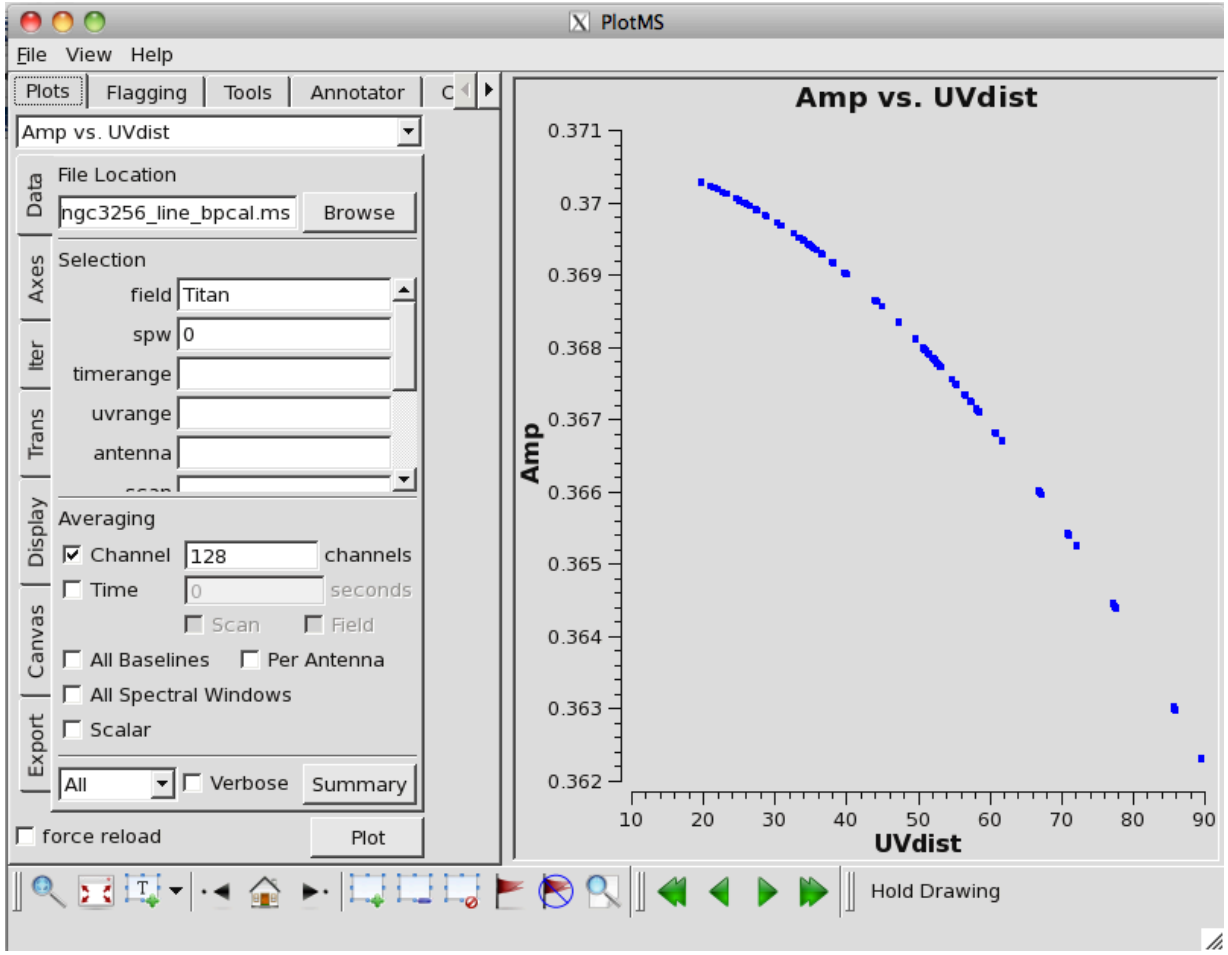
e.g., apply bandpass calibration before `gaincal`

Outline

- Basic Idea and Workflow (Review)
- Gaincal
- **Phase and Amplitude Calibration for NGC 3256**
- Before and After for NGC 3256

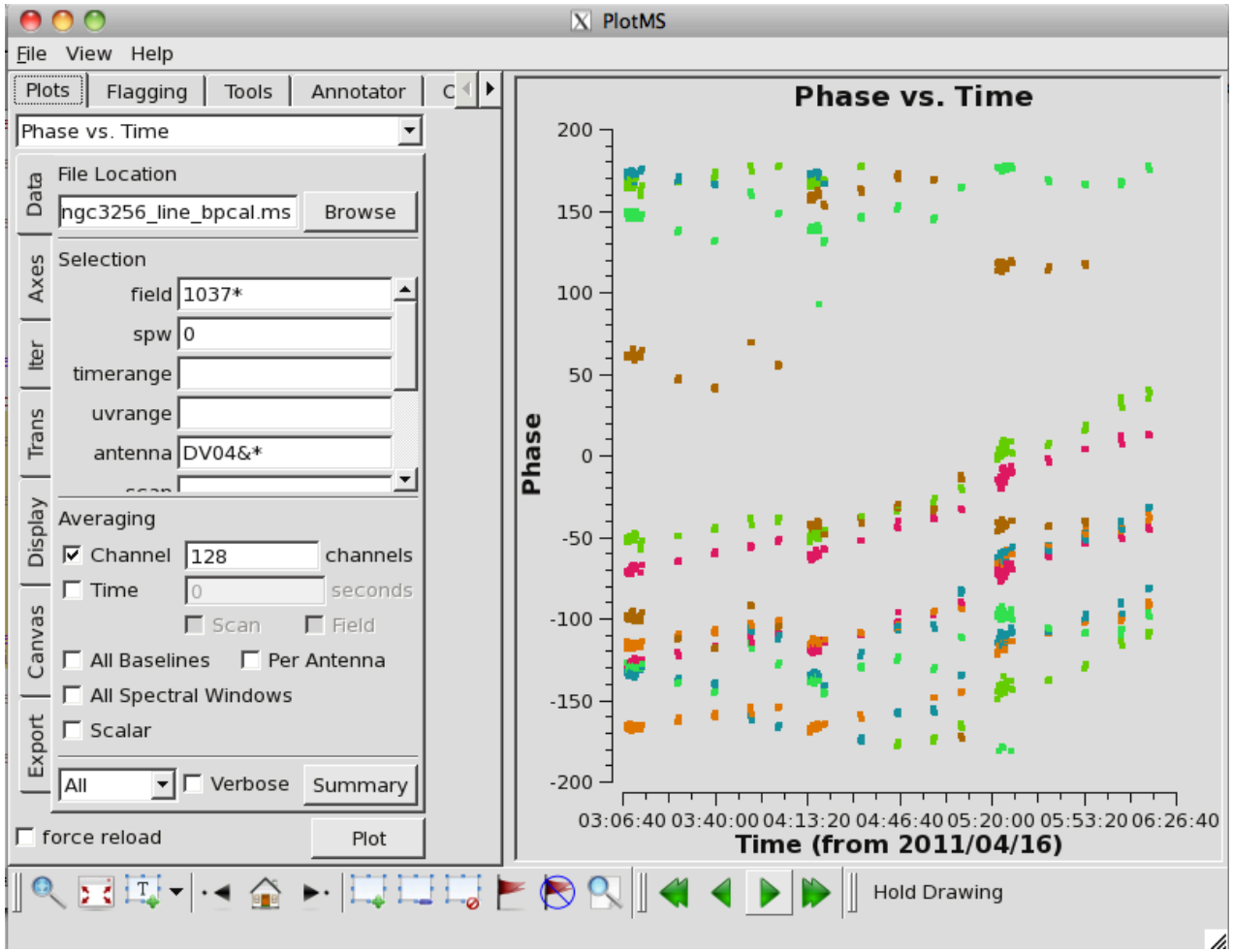
Read in a Model for Titan

Titan is resolved by ALMA and CASA includes a model. Read it in with setjy.



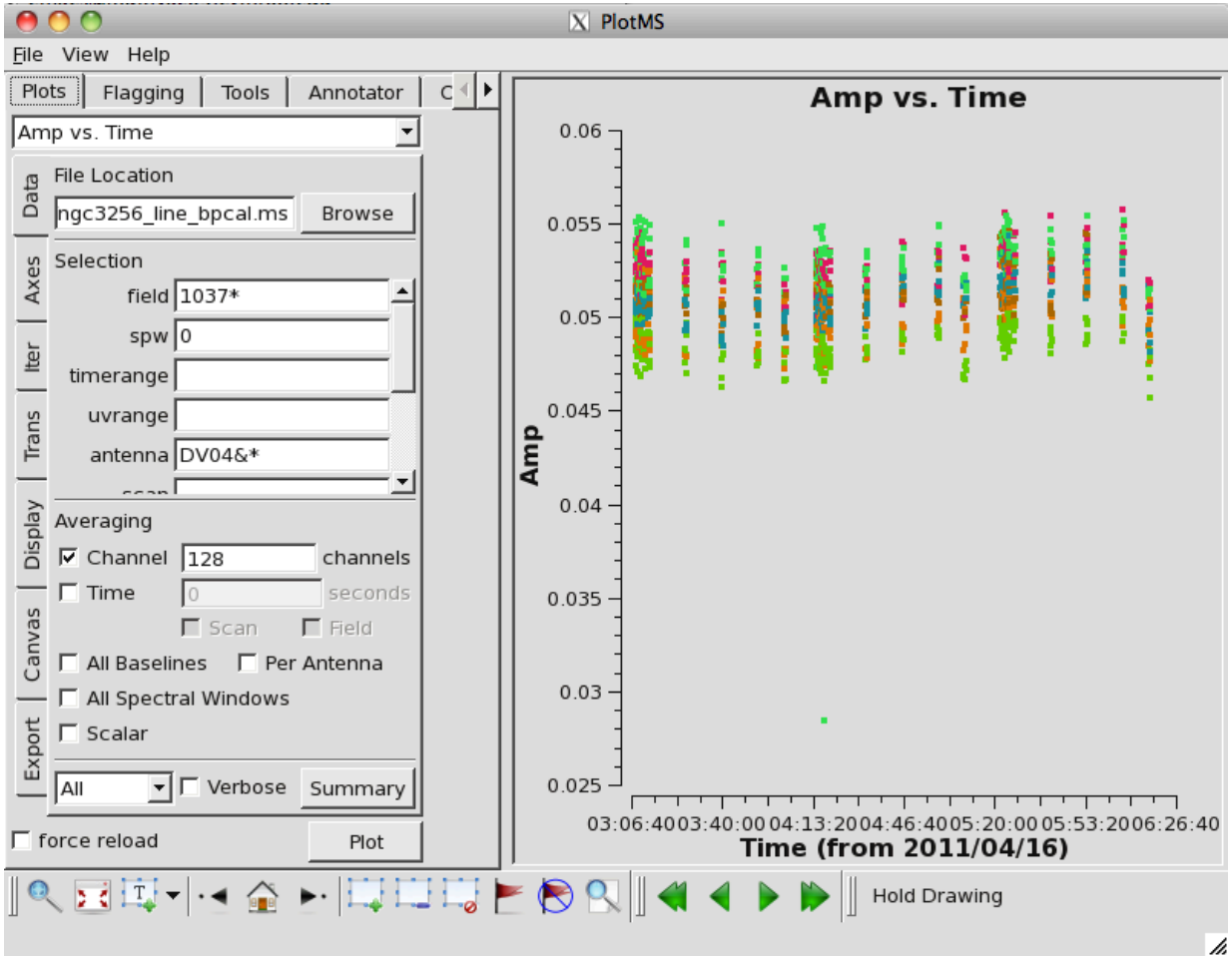
Phase vs. Time

Look at the phase vs. time on baselines with DV04.



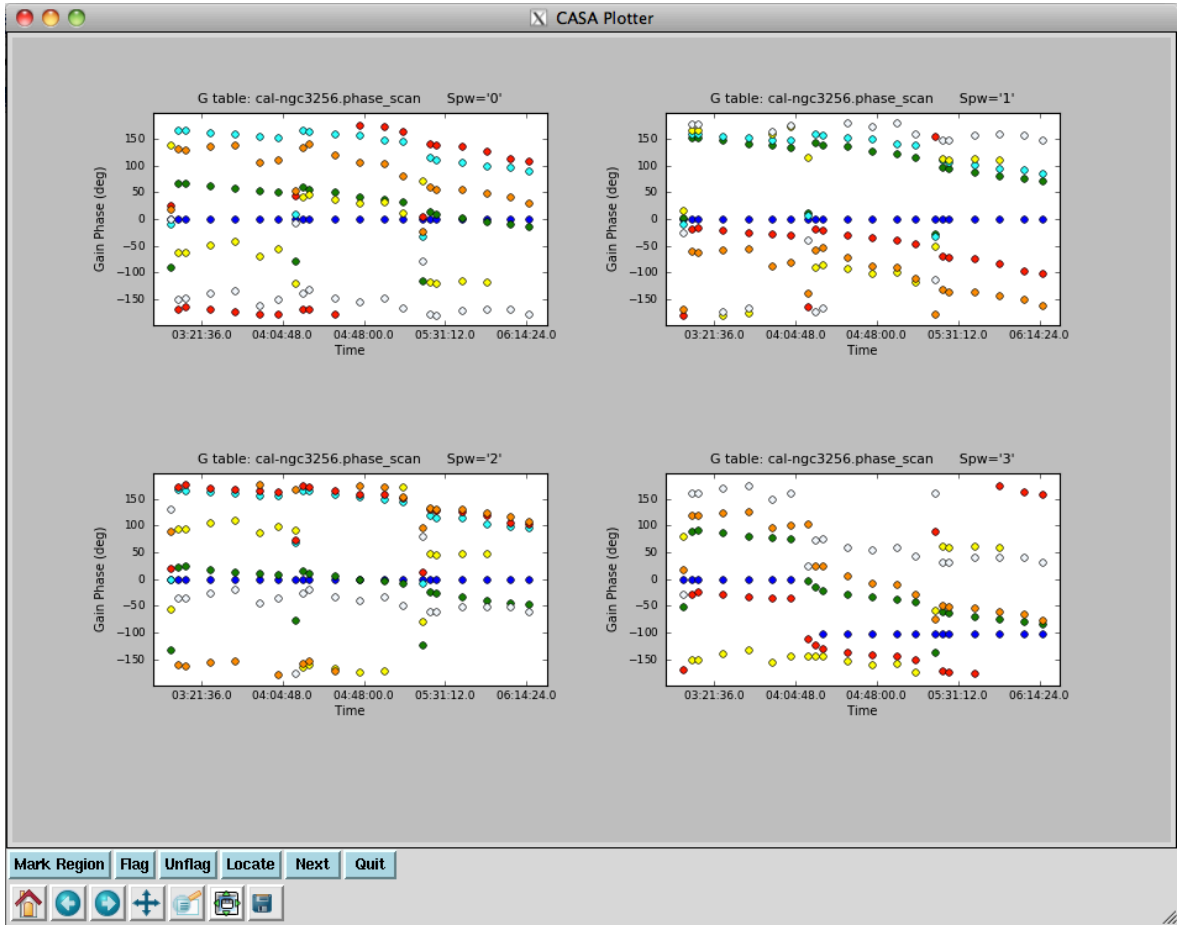
Amp vs. Time

Look at the amp vs. time on baselines with DV04.



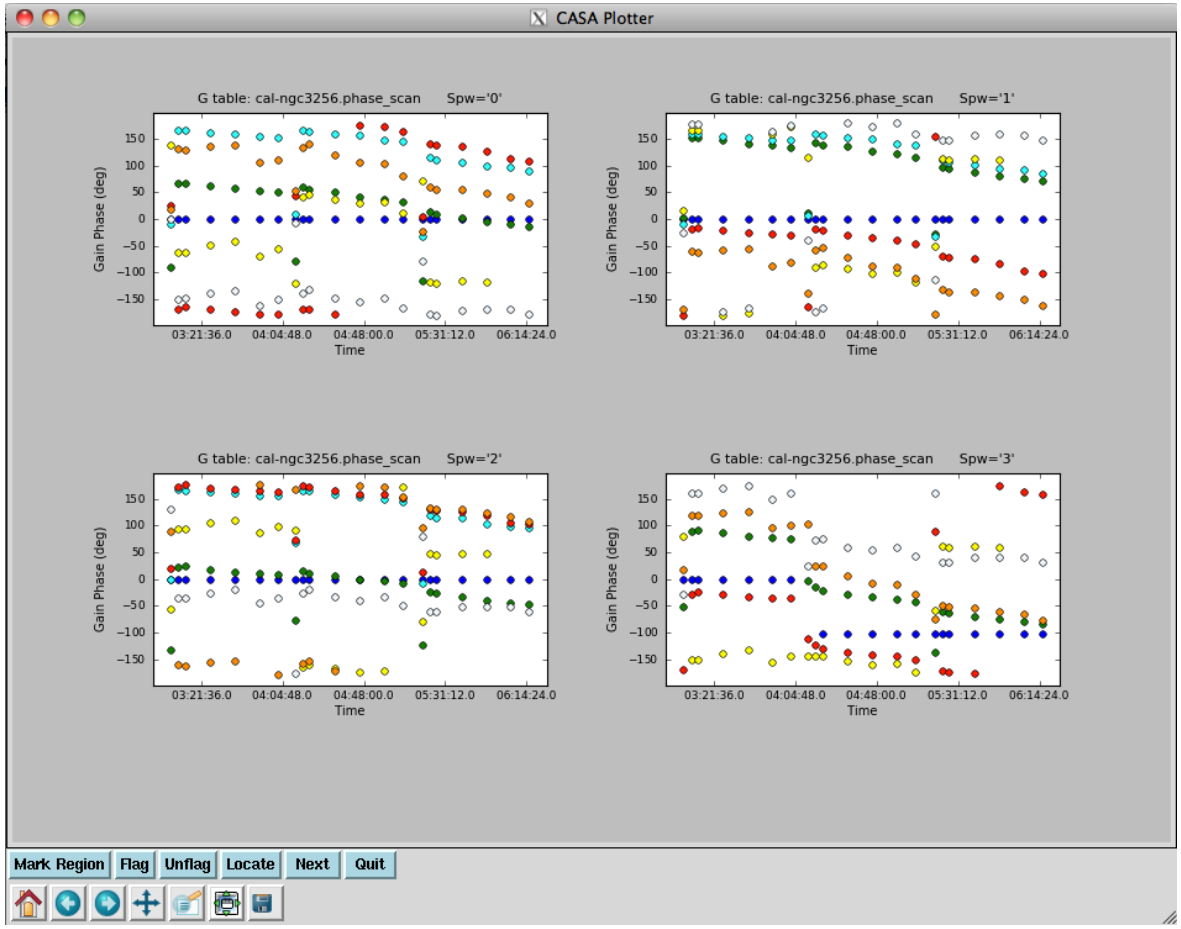
Solve for Phase vs. Time

Get a solution for each scan.



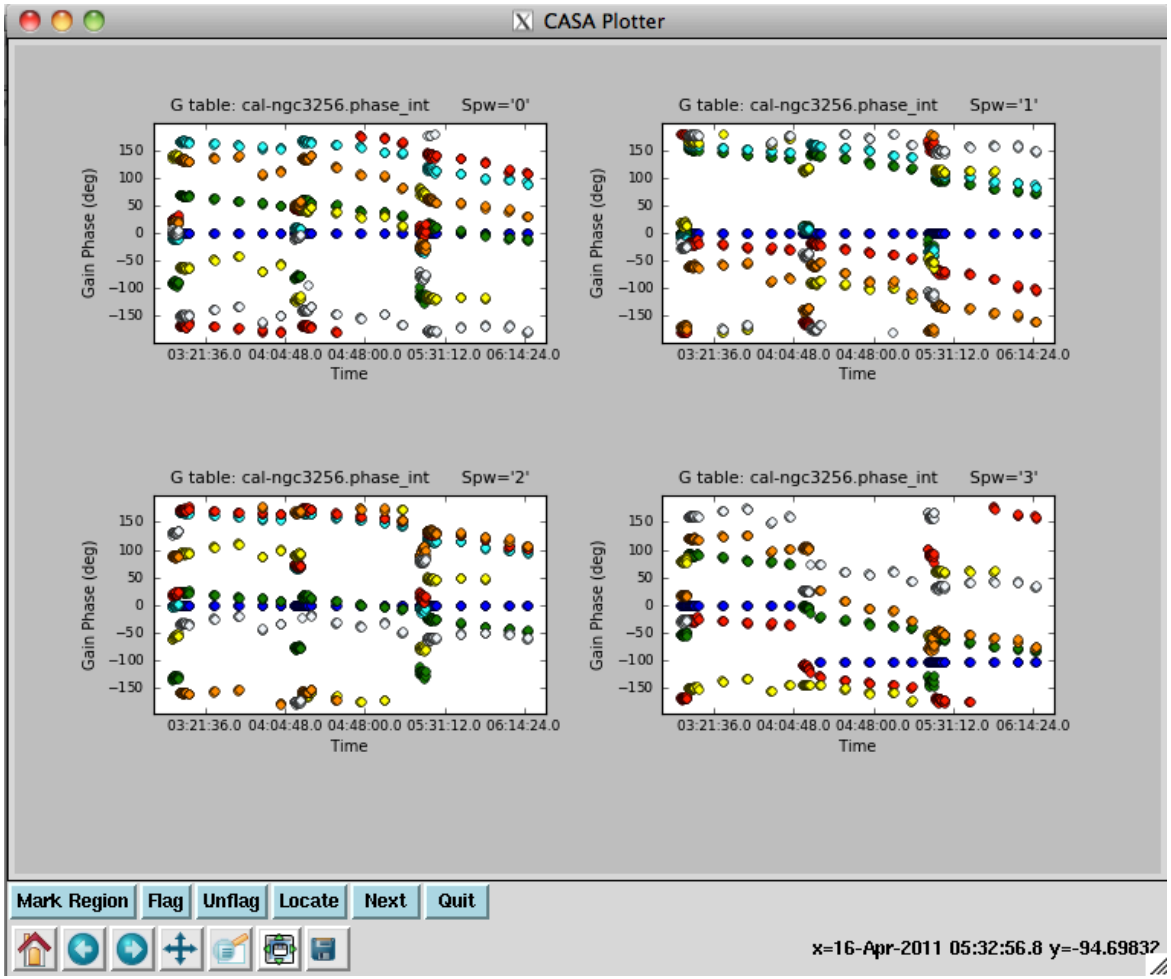
Solve for Phase vs. Time

Get a solution for each scan.



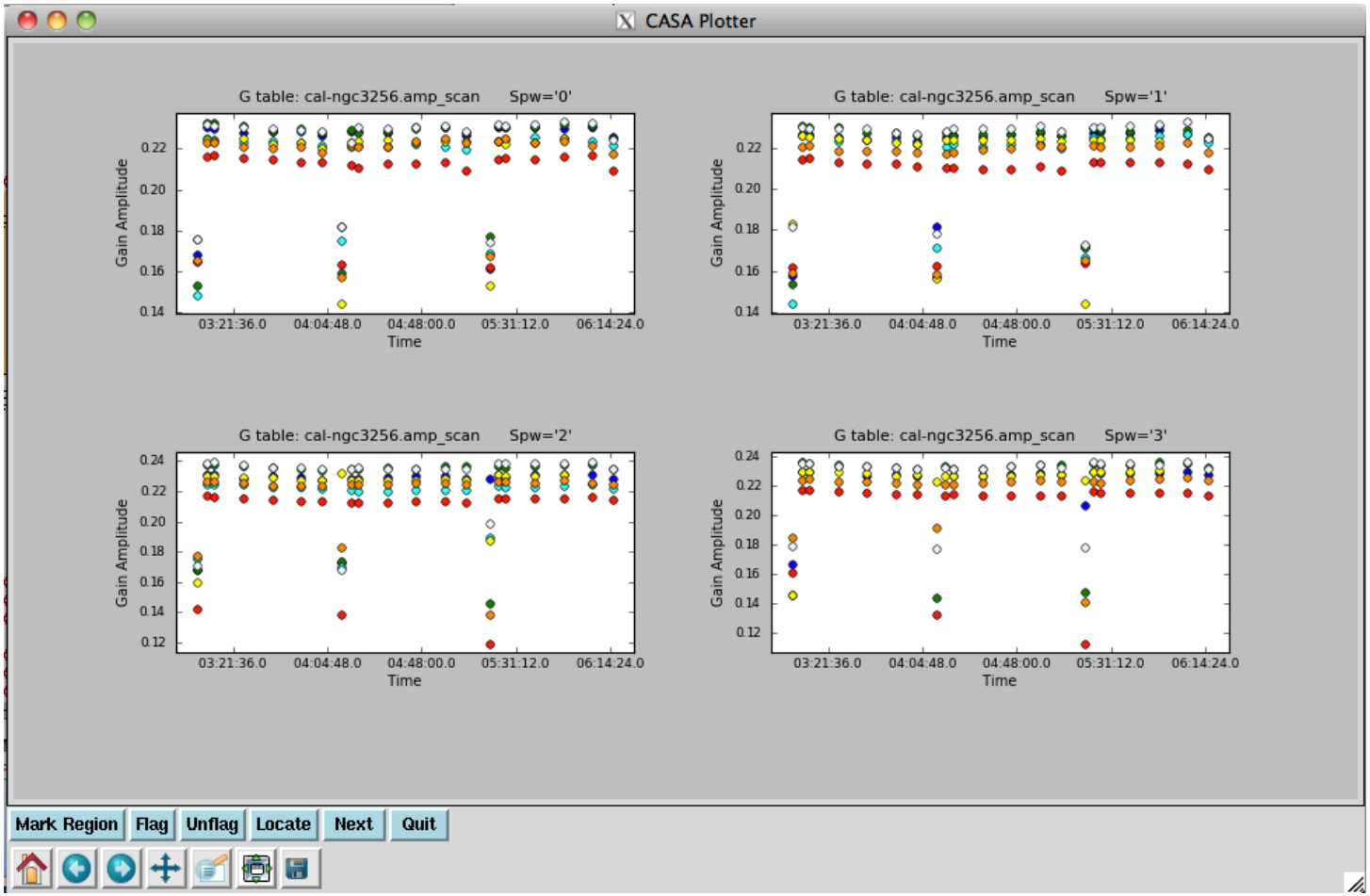
Short-timescale Phase

Get a solution for each integration (used to aid amplitude calibration).



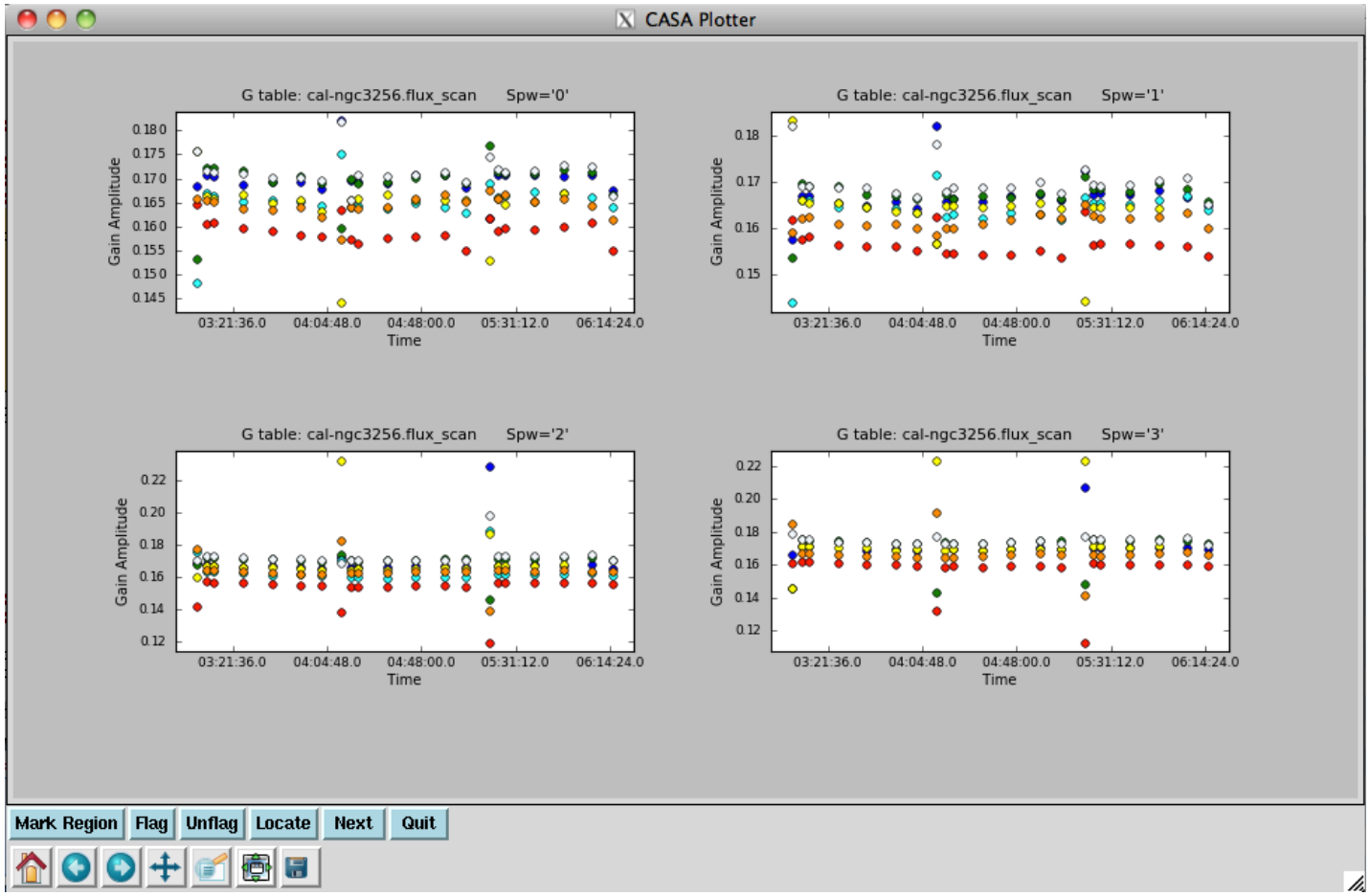
Per-Scan Amplitude

Amplitude solution per scan (note low-lying bad data).



Per-Scan Flux (after scaling)

Flux solution per scan (note scattered bad data).

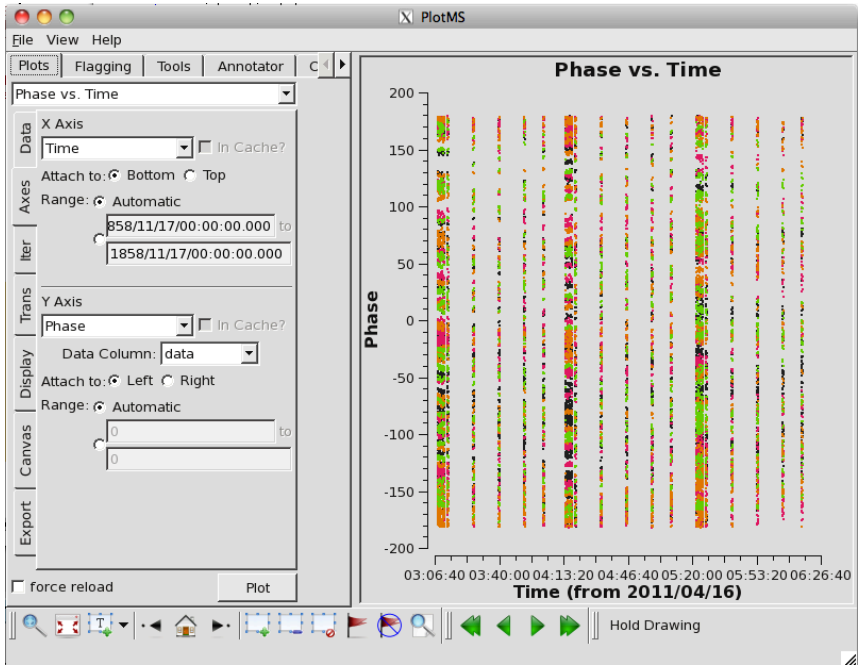


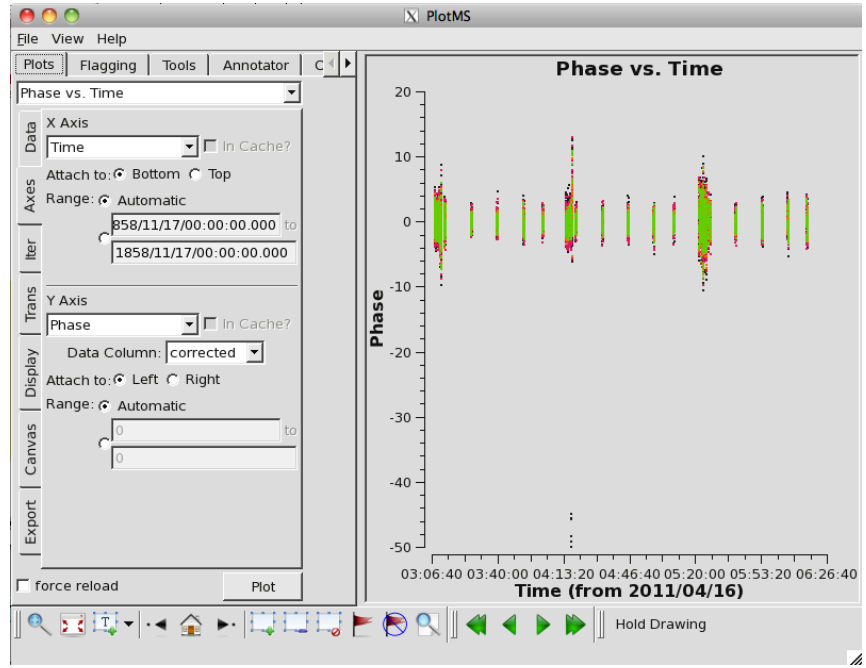
Outline

- Basic Idea and Workflow (Review)
- Gaincal
- Phase and Amplitude Calibration for NGC 3256
- **Before and After for NGC 3256**

Phase vs. time Before/After

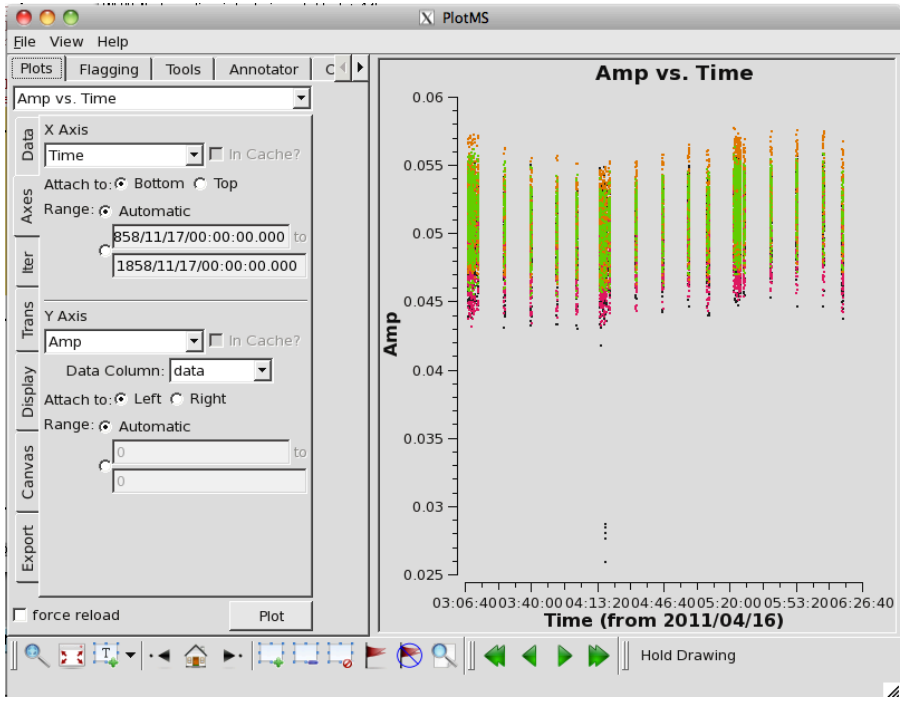
Phase on calibrator set to 0 at all times after calibration.

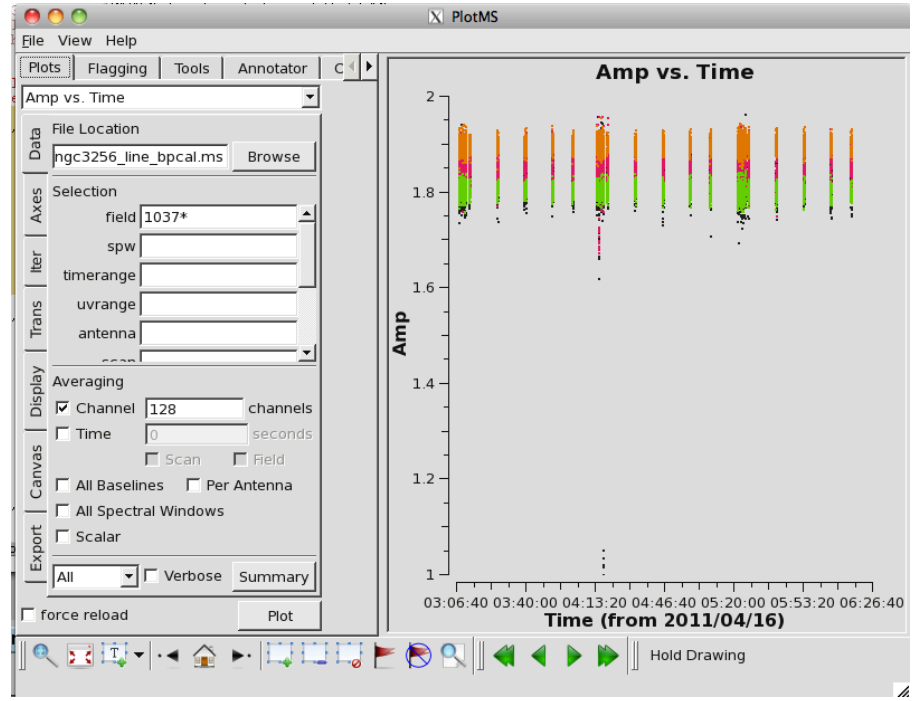




Amplitude vs. Time Before/After

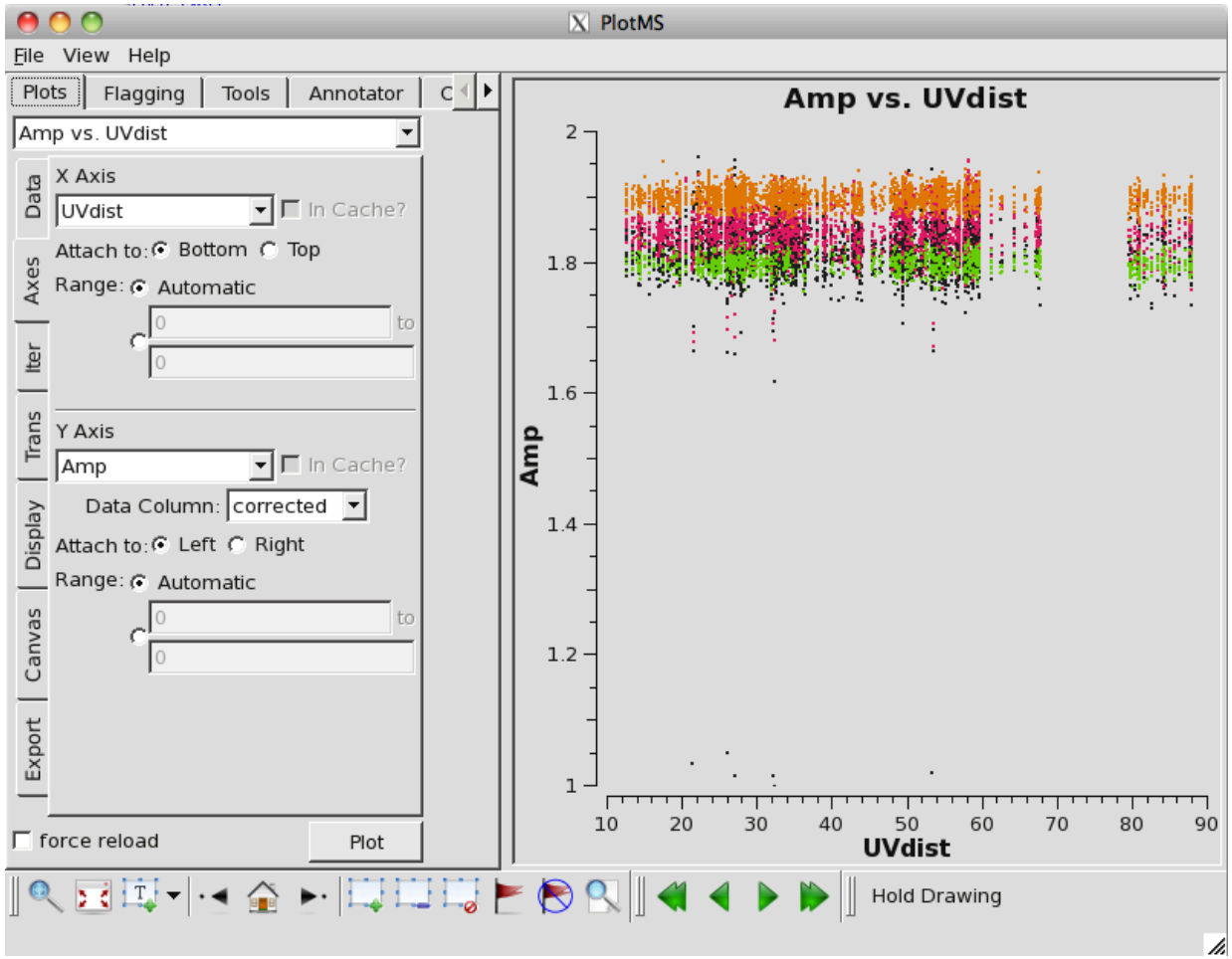
Calibration brings amplitude vs. time up to common, flat ~1.8 Jy.





Amp. vs uv Distance for Phasecal

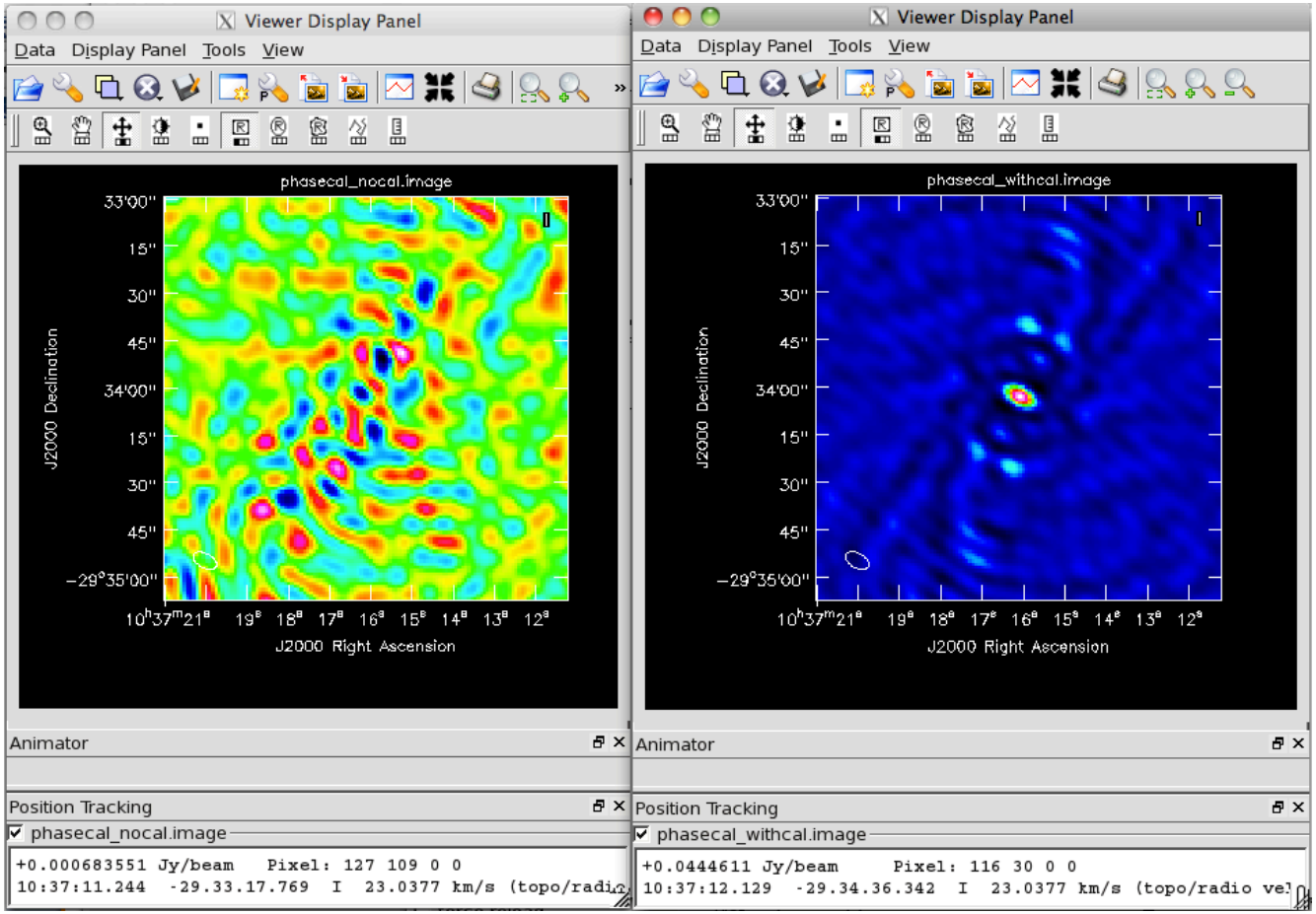
Point source has flat amplitude vs. uv distance.



Calibrator Image

imview(raster='phasecal_nocal.image')

imview(raster='phasecal_withcal.image')



Target Spectrum

Amplitude vs. frequency for the source after calibration.