ALMA Data Calibration

MC Toribio

ALLEGRO - CASA Training Day

Leiden Observatory - 2018/11/02

Based on / including slides from:

L.T. Maud + H. Nagai + G. Bendo + M. Zwaan+

+ G. Heald + A. Richards + A. Avison + R. Indebetouw +

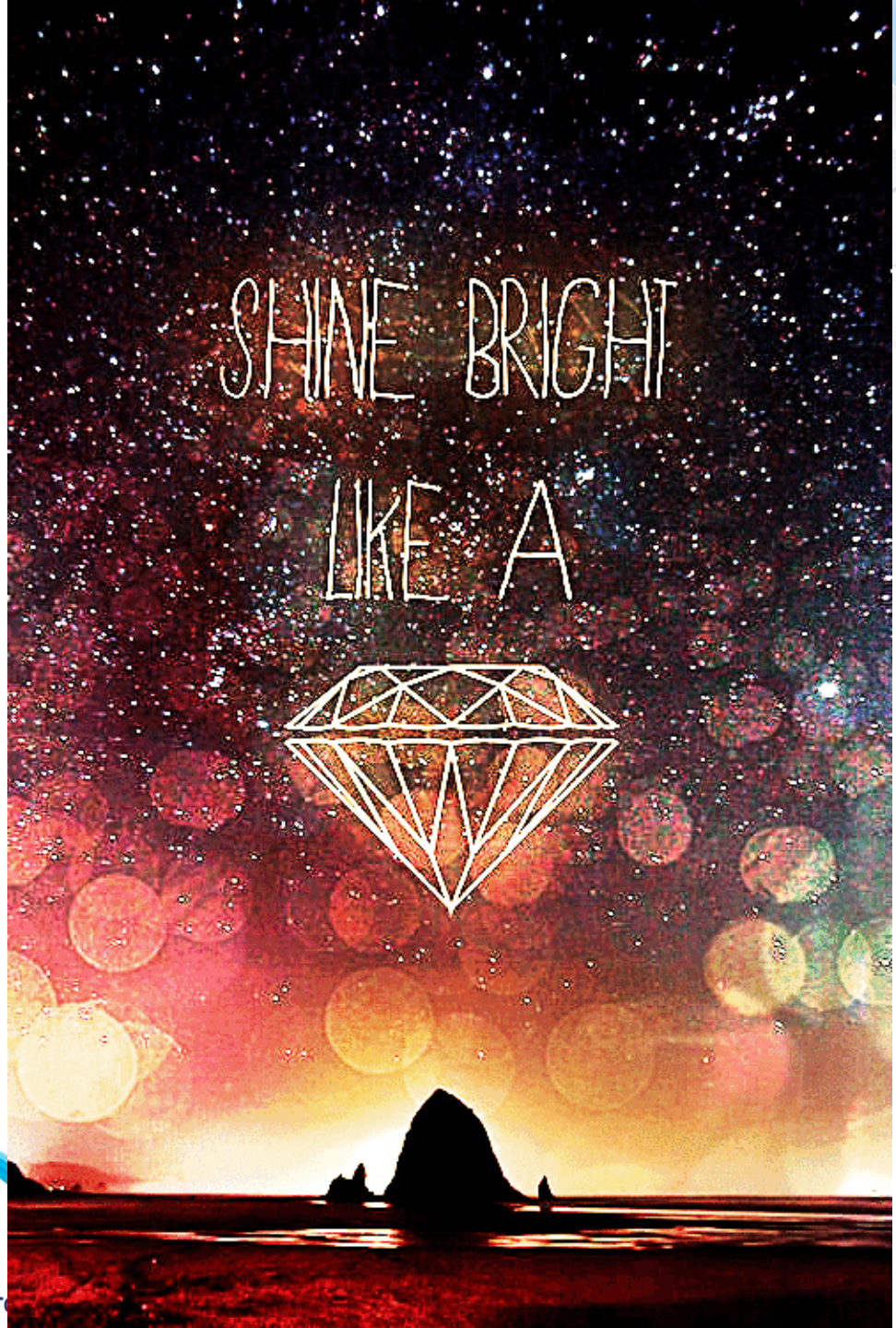
+ E. Fomalont + ALMA Technical Handbook +

+ CASA Guides + NRAO lectures + ERIS Lectures +

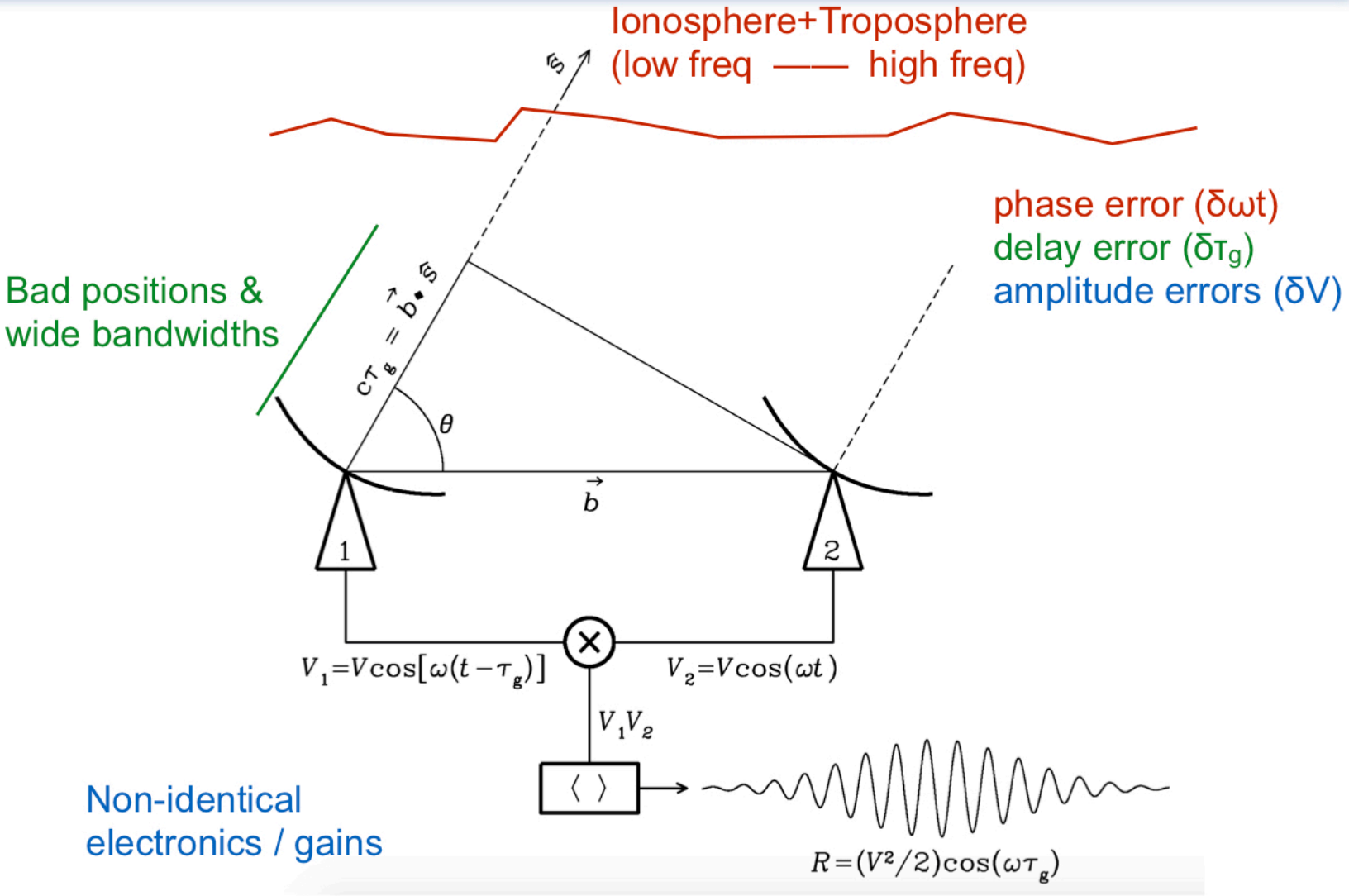
+ ALMA UK tutorials + IRAM School lectures.... ∞

EUROPEAN ARC

ALMA Regional Centre || Allegro


SHINE BRIGHT
LIKE A

Sketch of an ideal interferometer



Solve for these issues using calibration

CALIBRATION BASICS

$$V_{obs} = G_{ij} V_{sky}$$

HOW??

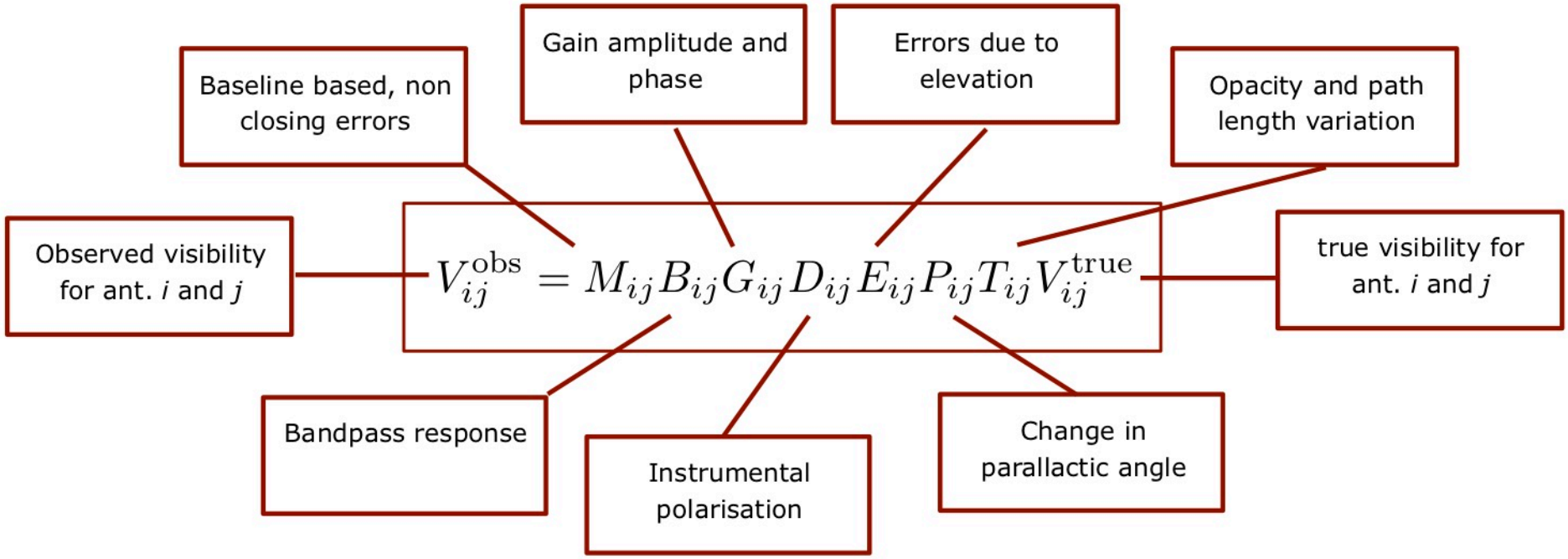
Making **A LOT OF ASSUMPTIONS**

- **time and frequency effects independent**
- **antenna based effects**
- **spatial structure of calibrators**
- **...**

e.g, A. Avison's introduction to calibration

Introduction to the RIME

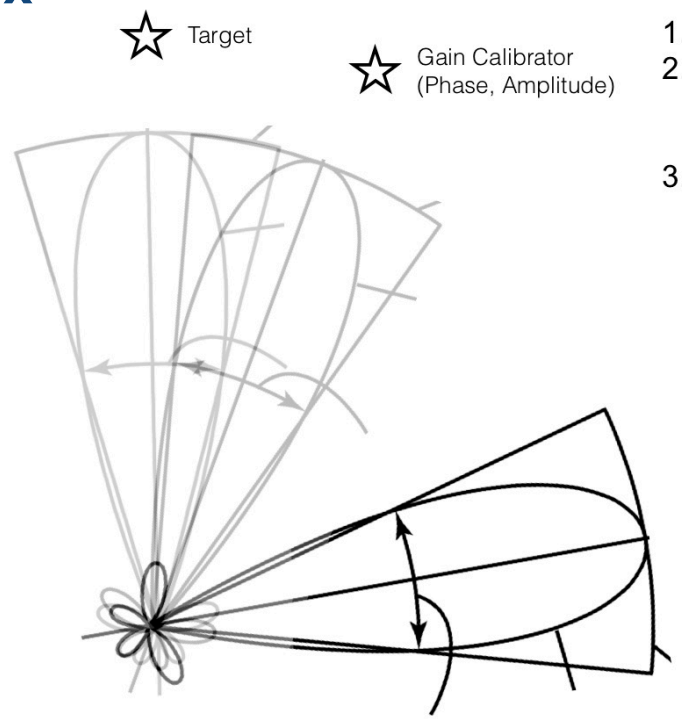
Within, for example **CASA**, the full radio interferometry measurement equation can be written as,



Calibration solves for each Jones matrix (when required) given a **model** for the sky.

Calibration - the basics

- **Remove effects of the instrument itself**
- **Remove effects of the atmosphere**
- **Scaling to the correct flux**



1. Observe **source**
2. Observe **calibrator** to measure gains (amplitude and phase) as a function of time.
3. Observe **bright calibrator** of known flux-density and spectrum to measure absolute flux calibration, band-pass and residual delays

Calibration - the basics

- Remove effects of the instrument itself
  - variations in frequency as a function of amplitude and phase
  - variation in receiver noise (Tsys)
  - effects of shadowing
  - antenna positions
- Remove effects of the atmosphere
  - atmospheric variability (phase)
  - atmospheric attenuation as function of time (Tsys)
- Scaling to the correct flux

-using an astronomical source

Calibration - the basics

- Remove effects of the instrument itself
  - variations in frequency as a function of amplitude and phase
  - variation in receiver noise (Tsys)
  - effects of shadowing
  - antenna positions

- Bandpass
- Tsys load
- Flagging
- All sky runs

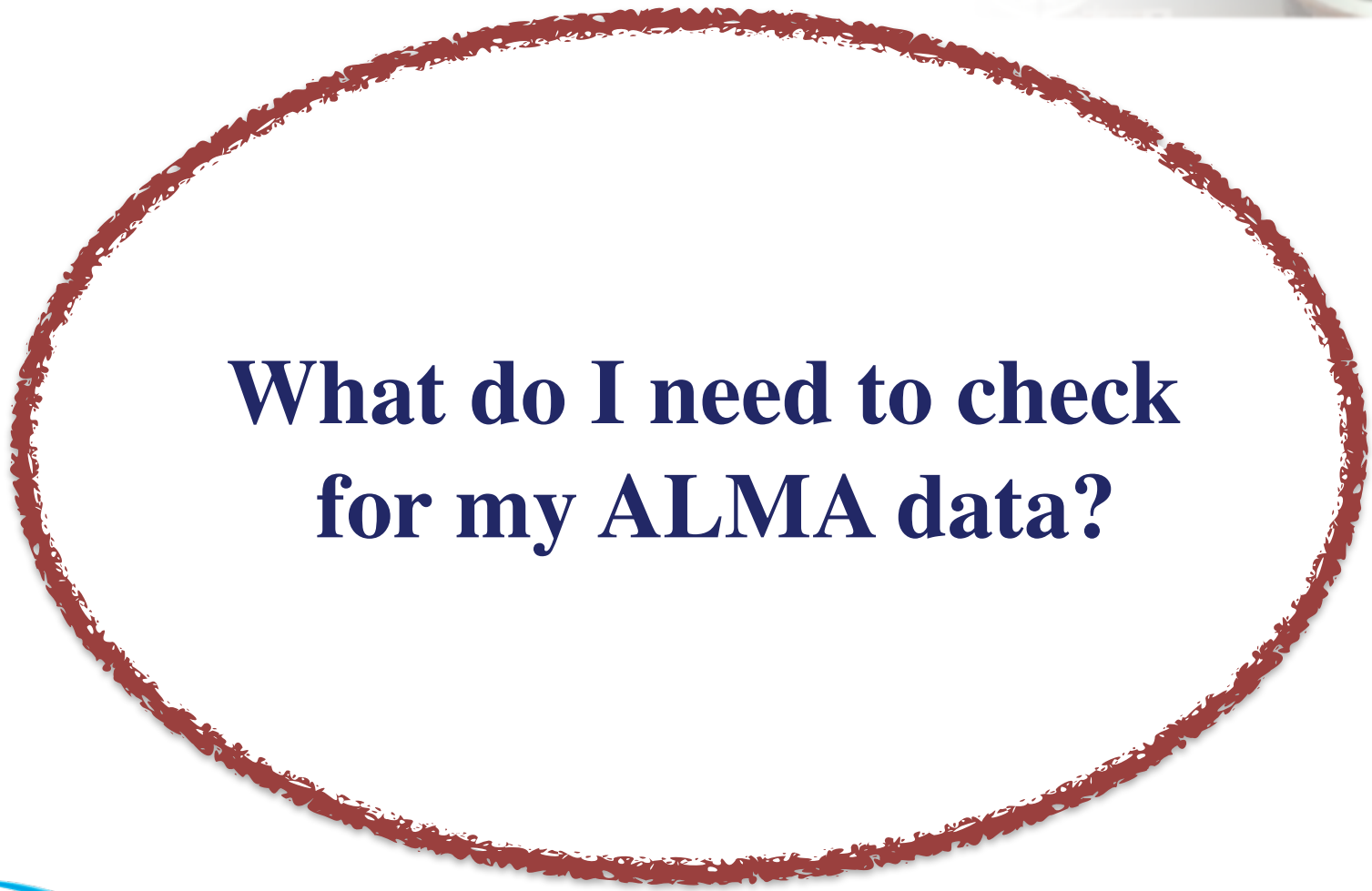
- Remove effects of the atmosphere
  - atmospheric variability (phase)
  - atmospheric attenuation as function of time (Tsys)

- WVRs + Gain Calibrator
- Tsys load

- Scaling to the correct flux
  - using an astronomical source

- Flux Calibrator

Calibration - the basics



Calibration - the basics

Bandpass

QSO -freq. response

Flux

planet/QSO

Temporal Gains

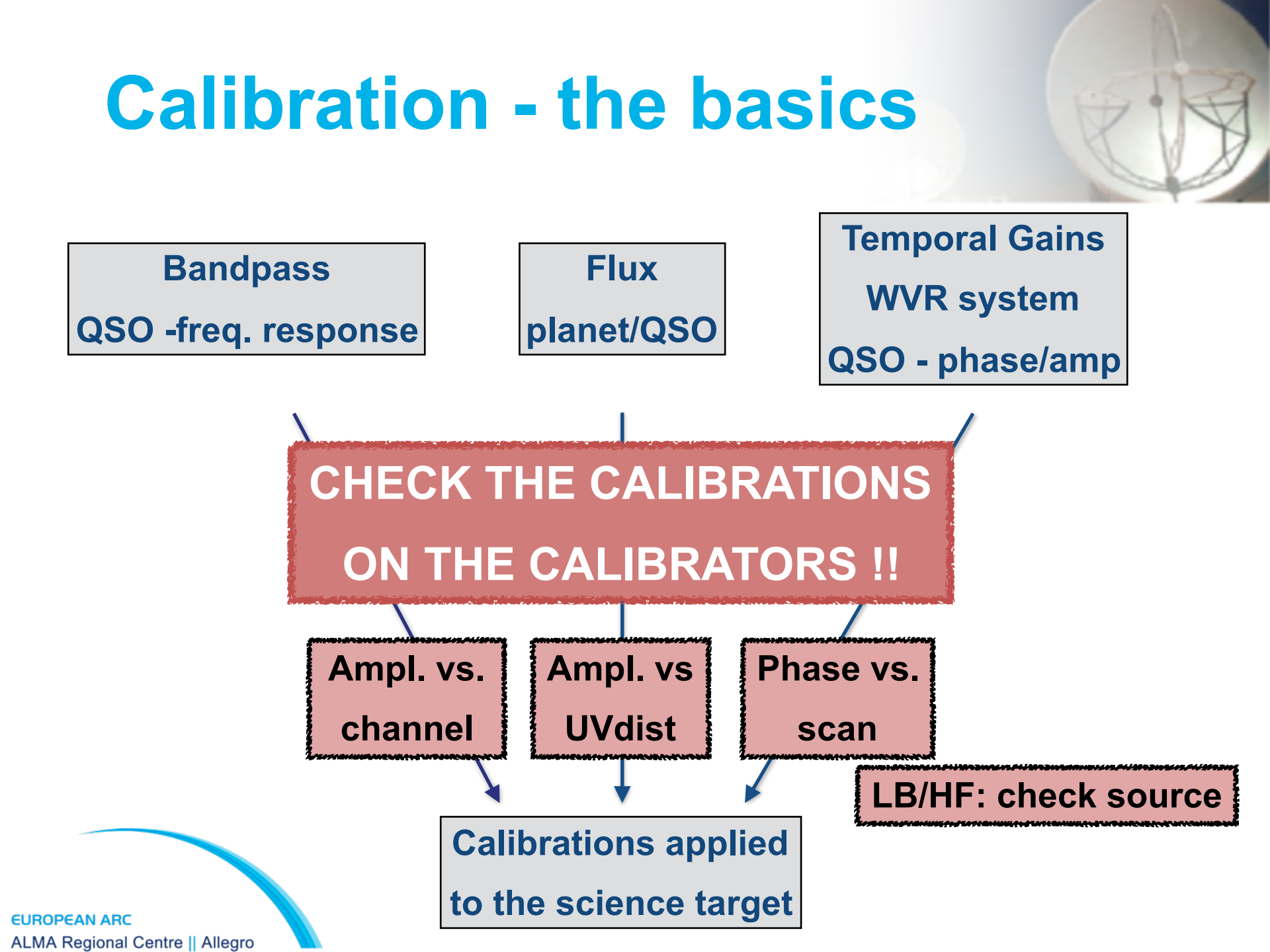
WVR system

QSO - phase/amp

CHECK THE CALIBRATIONS ON THE CALIBRATORS !!

Calibrations applied to the science target

Calibration - the basics

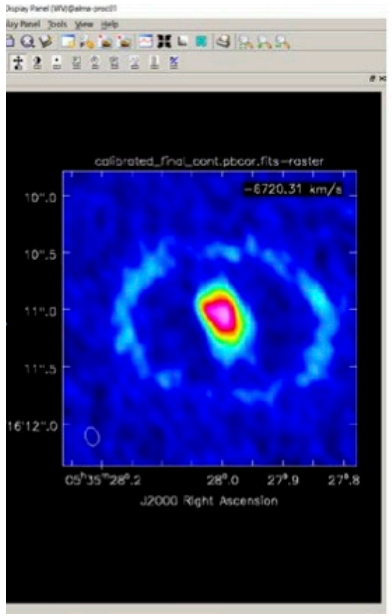


TUTORIAL DATA

home.strw.leidenuniv.nl/~toribio/CASAtutorial2018_data

- ✓ manual calibration: IRAS16293 Science Verification Band 4

- ✓ pipeline calibration: SN1987a in the Archive
  - 2013.1.00063.S/
    - science_goal.uid___A001_X12e_X27e/
      - group. uid___A001_X12e_X27f/
        - member.uid___A001_X12e_X280



Calibration - with CASA

All run from the scripts/scriptForPI.py

- ✓ manual calibration
- - uid__XXXXX_scriptForCalibration.py

- ✓ pipeline calibration
- - casa_piperestorescript.py
- - casa_pipescript.py

Calibration - with CASA

All run from the scriptForPI.py

- manual calibration
- uid__XXXXX_scriptForCalibration.py
  - + scriptForImagingPrep.py
  - + scriptForImaging.py

- pipeline calibration
- casa_piperestorescript.py
- casa_pipescript.py
  - + scriptForImagingPrep.py
  - + scriptForImaging.py

Calibration - with CASA

All run from the scriptForPI.py

- manual calibration
- uid__XXXXX_scriptForCalibration.py
  - + scriptForImagingPrep.py
  - + scriptForImaging.py

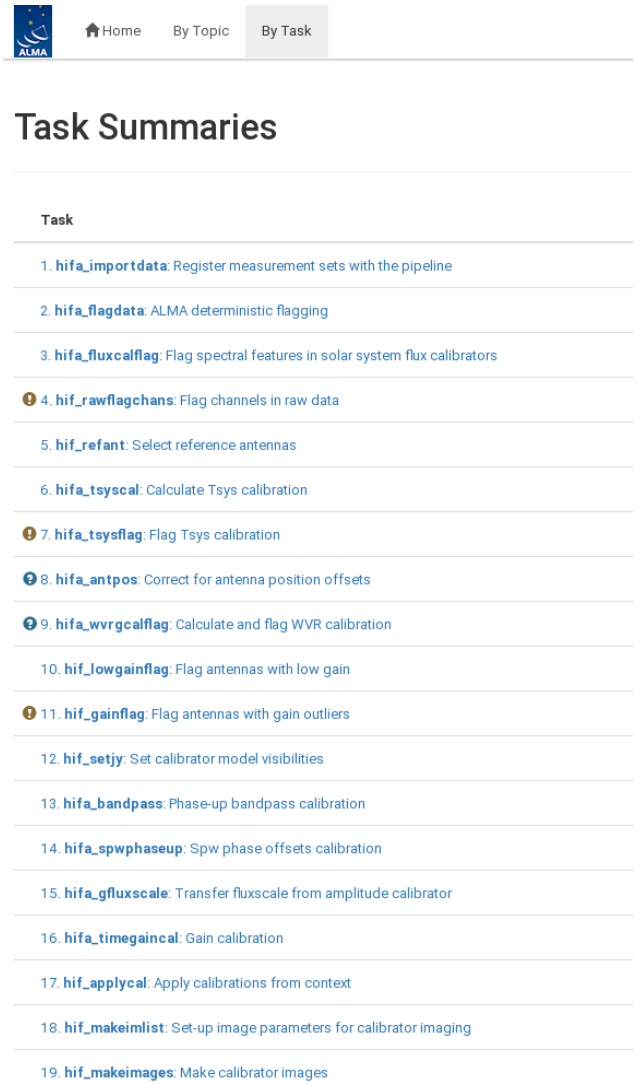
- pipeline calibration
- casa_piperestorescript.py
- casa_pipescript.py
  - + scriptForImagingPrep.py
  - + scriptForImaging.py

Calibration - with CASA

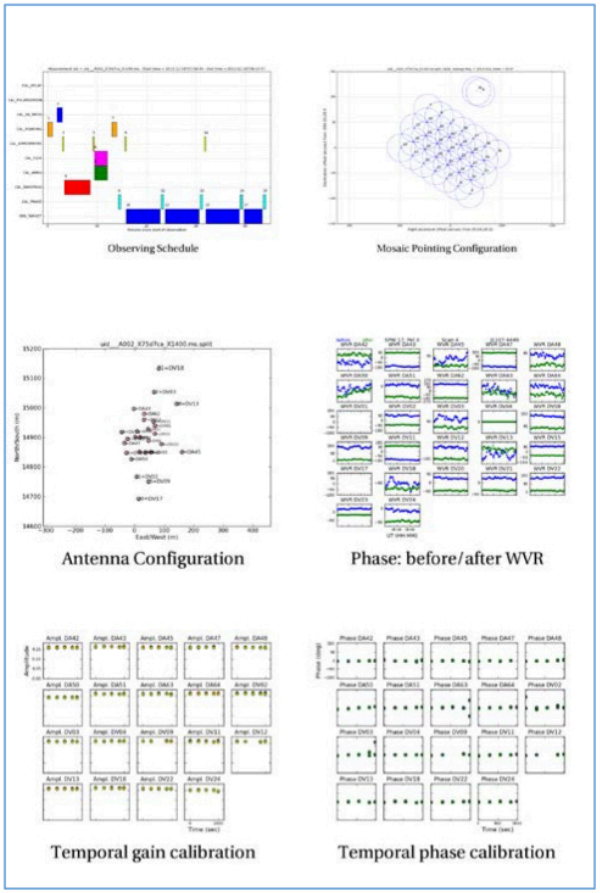
MANUAL

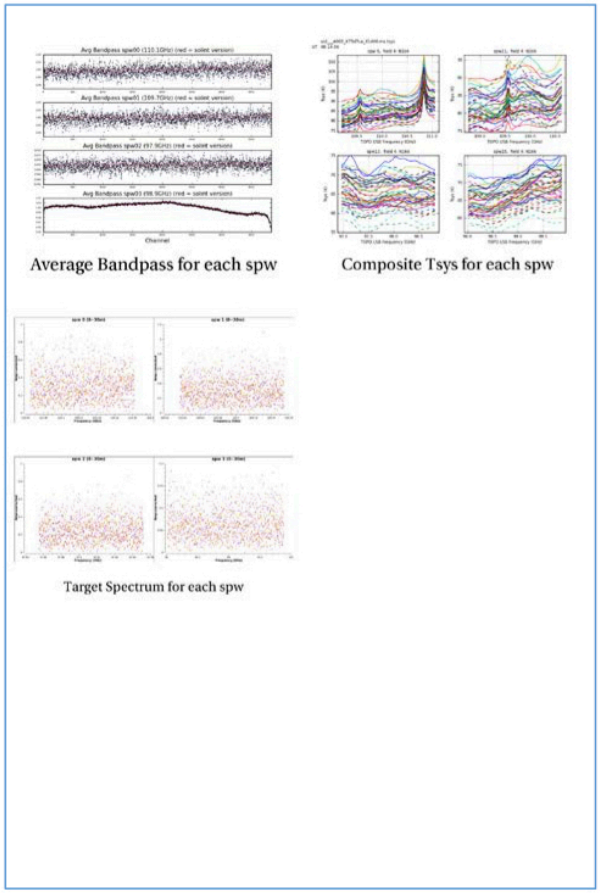
```
# ALMA Data Reduction Script

# Calibration

thesteps = []
step_title = {0: 'Import of the ASDM',
              1: 'Fix of SYSCAL table times',
              2: 'listobs',
              3: 'A priori flagging',
              4: 'Generation and time averaging of the WVR cal table',
              5: 'Generation of the Tsys cal table',
              6: 'Generation of the antenna position cal table',
              7: 'Application of the WVR, Tsys and antpos cal tables',
              8: 'Split out science SPWs and time average',
              9: 'Listobs, and save original flags',
              10: 'Initial flagging',
              11: 'Putting a model for the flux calibrator(s)',
              12: 'Save flags before bandpass cal',
              13: 'Bandpass calibration',
              14: 'Save flags before gain cal',
              15: 'Gain calibration',
              16: 'Save flags before applycal',
              17: 'Application of the bandpass and gain cal tables',
              18: 'Split out corrected column',
              19: 'Save flags after applycal'}

if 'applyonly' not in globals(): applyonly = False
try:
  print 'List of steps to be executed ...', mysteps
  thesteps = mysteps
except:
  print 'global variable mysteps not set.'
if (thesteps==[]):
  thesteps = range(0,len(step_title))
  print 'Executing all steps: ', thesteps
```

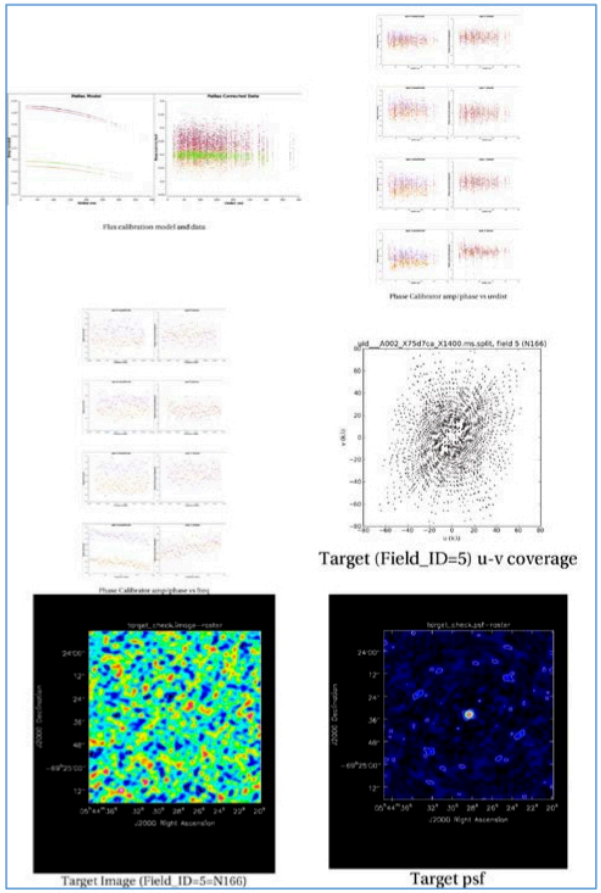
PIPELINE



MANUAL:QA2 REPORT







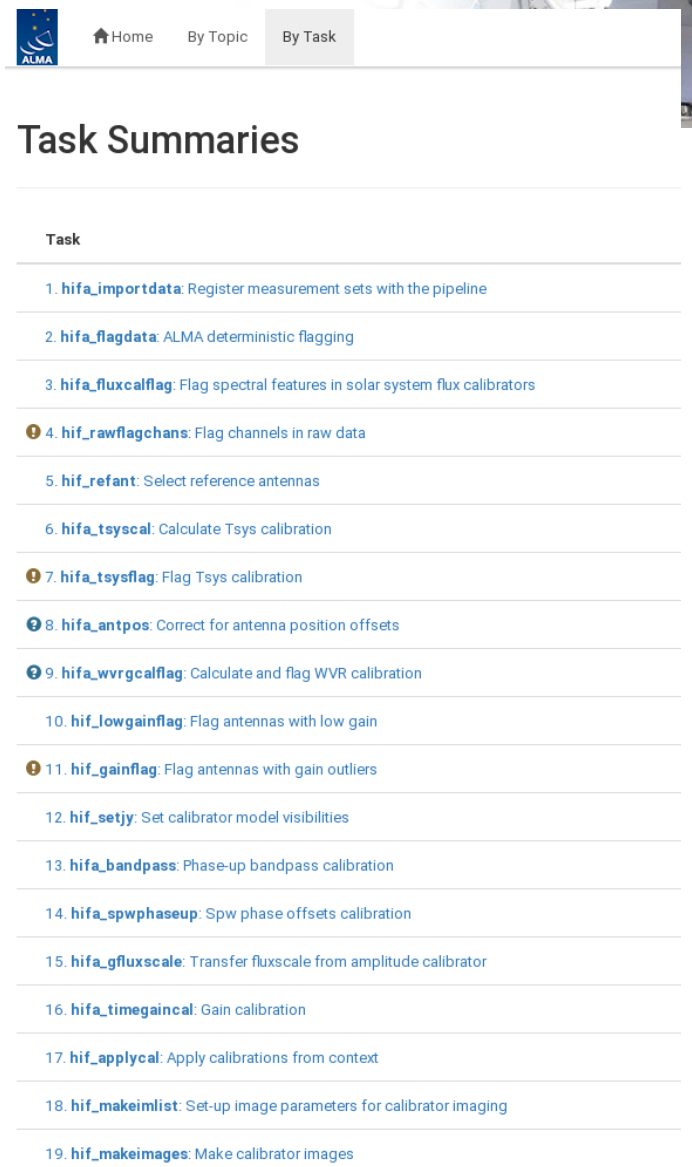
Basic png diagnostic figures

EUROPEAN ARC
ALMA Regional Centre || Allegro

H. Nagai

PIPELINE: WEBLOG

Webpages with all diagnostic plots and other information generated by the pipeline



WEBLOG INSPECTION

UNDERSTAND YOUR OBSERVATIONS!

- ✓ Check observation summary
- ✓ Check flagging summaries
- ✓ Check calibrated data
  - If any suspicion, check relevant calibrations
- ✓ Check flux consistency
  - Search calibrator fluxes in ALMA database:

    aU.getALMAFlux/ Source Catalogue

Calibration - with CASA

> cd $ALLEGRO/home/$USER/

open_CASA_training_2018/analysis/$USER

Please navigate to your sn1987a/

✓check the README.txt

✓qa/pipeline-XXXweblog/

->...-> firefox index.html &

home.strw.leidenuniv.nl/~toribio/CASAtutorial2018_data/sn1987a/qa/pipeline-20151016T113145/html/

WEBLOG INSPECTION

Observation Overview

Project	uid://A001/X10a/X6a
Principal Investigator	ri3e
OUS Status Entity id	uid://A001/X12e/X280
Observation Start	2015-06-28 13:56:22 UTC
Observation End	2015-09-22 11:00:50 UTC

Pipeline Summary

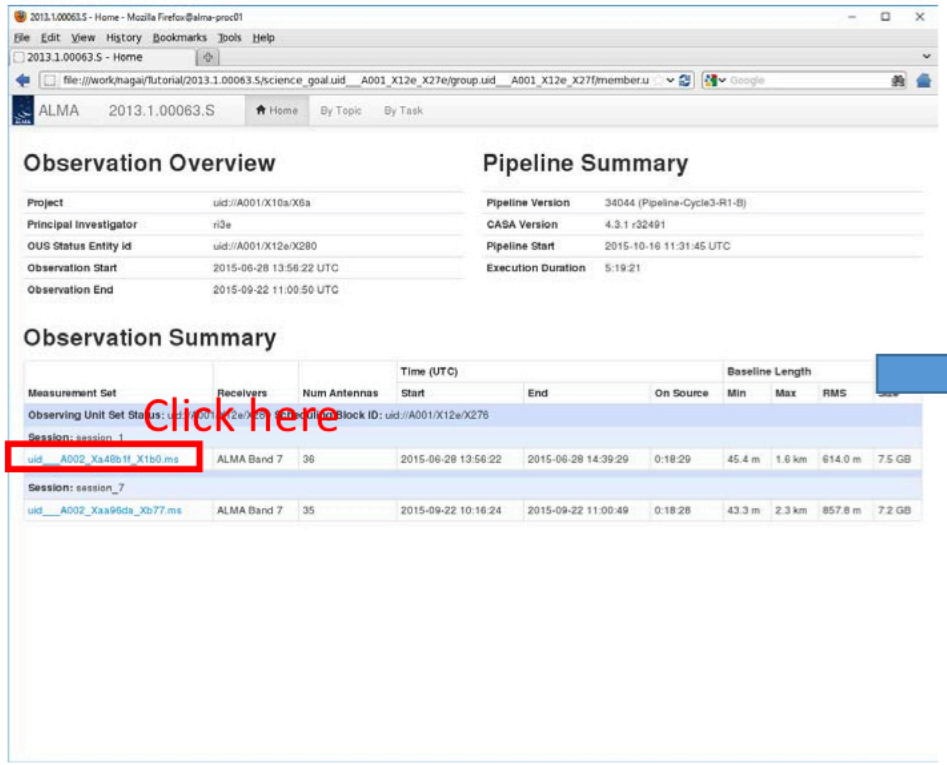
Pipeline Version	34044 (Pipeline-Cycle3-R1-B)
CASA Version	4.3.1 r32491
Pipeline Start	2015-10-16 11:31:45 UTC
Execution Duration	5:19:21

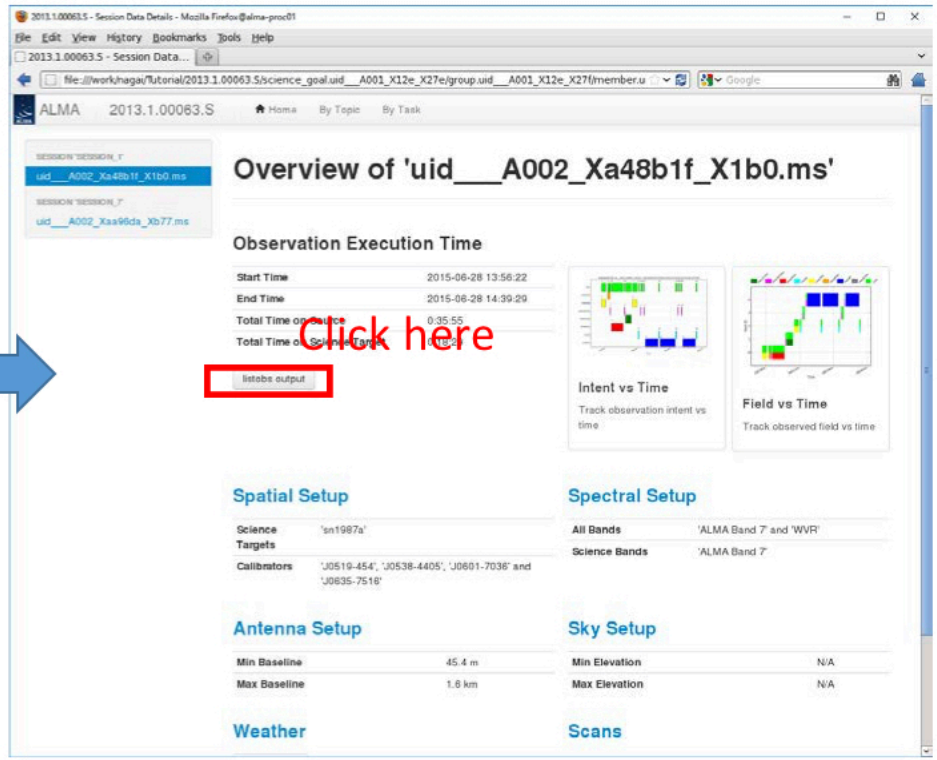
Observation Summary

			Time (UTC)			Baseline Length	
Measurement Set	**Receivers**	**Num Antennas**	**Start**	**End**	**On Source**	**Min**	**Max**
Observing Unit Set Status: uid://A001/X12e/X280 **Scheduling Block ID:** uid://A001/X12e/X276							
Session: session_1							
uid___A002_Xa48b1f_X1b0.ms	ALMA Band 7	36	2015-06-28 13:56:22	2015-06-28 14:39:29	0:18:29	45.4 m	1.6 km
Session: session_7							
uid___A002_Xaa96da_Xb77.ms	ALMA Band 7	35	2015-09-22 10:16:24	2015-09-22 11:00:49	0:18:28	43.3 m	2.3 km

For this data, there are two execution blocks.

WEBLOG INSPECTION





H. Nagai

LISTOBS

ALMA 2013.1.00063.S | Home | By Topic | By Task

SESSION 'SESSION_1'

uid___A002_Xa48b1f_X1b0.ms

SESSION 'SESSION_7'

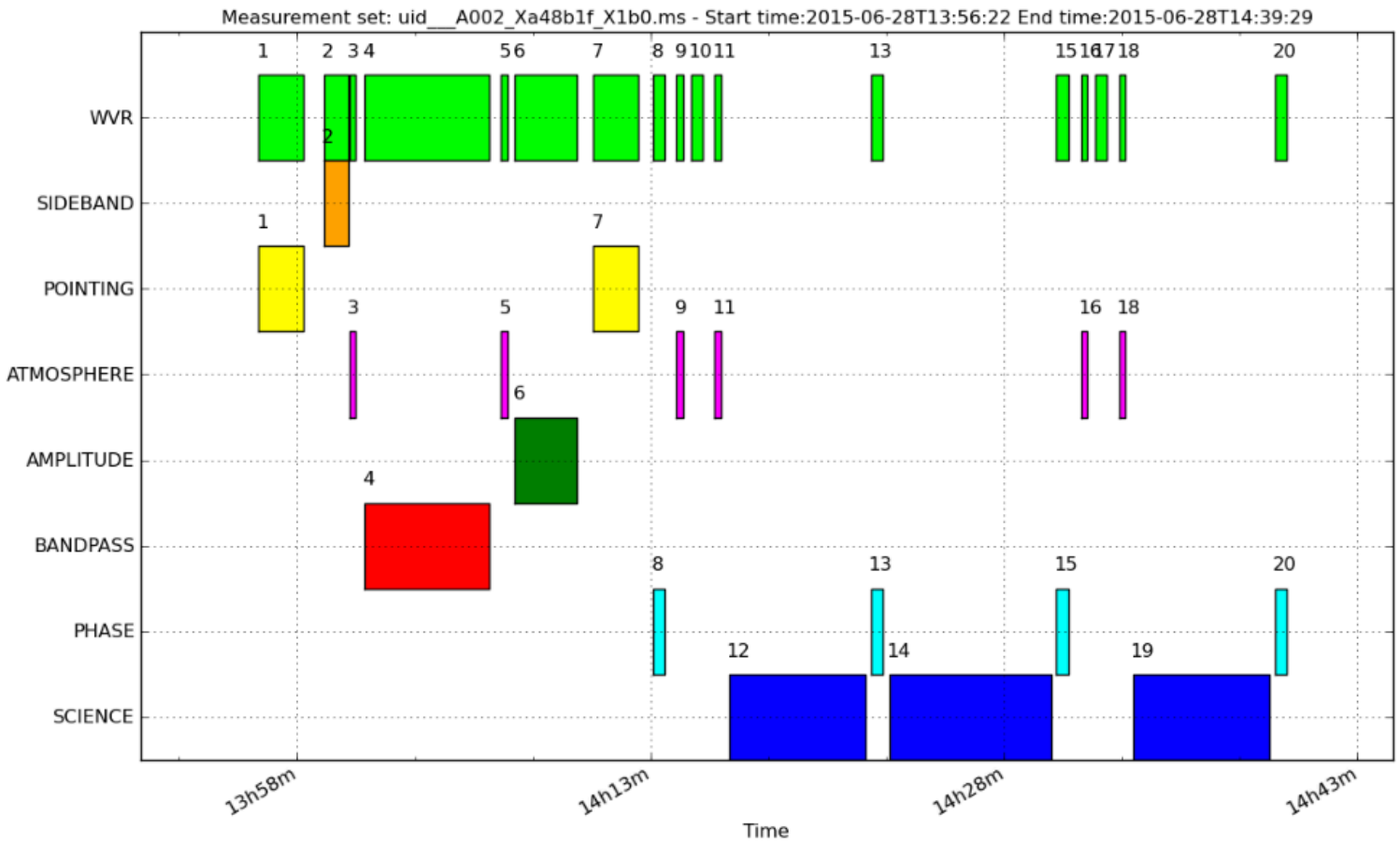
uid___A002_Xaa96da_Xb77.ms

listobs.txt

Back

```
================================================================================
           MeasurementSet Name:  /mnt/jaosco/pipeline/data/2013.1.00063.S_2015_10_16T10_42_17.689/SOUS_uid___A001_X12e_X27e/GOUS_uid___A001_X12e_X27f/MOUS_uid___A001_X12e_X28
0/working/uid___A002_Xa48b1f_X1b0.ms      MS Version 2
================================================================================
   Observer: ri3e     Project: uid://A001/X10a/X6a
Observation: ALMA
Data records: 9014220       Total elapsed time = 2617.68 seconds
   Observed from   28-Jun-2015/13:56:22.3   to   28-Jun-2015/14:40:00.0 (UTC)

   ObservationID = 0         ArrayID = 0
  Date        Timerange (UTC)          Scan  FldId FieldName             nRows     SpwIds   Average Interval(s)    ScanIntent
  28-Jun-2015/13:56:22.3 - 13:58:17.4     1      0 J0538-4405              402732  [0,1,2,3,4,5,6,7,8]  [1.15, 2.02, 1.01, 2.02, 1.01, 2.02, 1.01, 2.02, 1.01] [CALIBRATE_POINT
ING#ON_SOURCE,CALIBRATE_WVR#ON_SOURCE]
              13:59:09.3 - 14:00:11.6     2      0 J0538-4405              673236  [0,9,10,11,12,13,14,15,16]  [1.15, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48] [CALIBRAT
E_SIDEBAND_RATIO#OFF_SOURCE,CALIBRATE_SIDEBAND_RATIO#ON_SOURCE,CALIBRATE_WVR#OFF_SOURCE,CALIBRATE_WVR#ON_SOURCE]
              14:00:13.8 - 14:00:30.0     3      0 J0538-4405              128268  [0,9,10,11,12,13,14,15,16]  [1.15, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48] [CALIBRAT
E_ATMOSPHERE#OFF_SOURCE,CALIBRATE_ATMOSPHERE#ON_SOURCE,CALIBRATE_WVR#OFF_SOURCE,CALIBRATE_WVR#ON_SOURCE]
              14:00:51.8 - 14:06:08.7     4      0 J0538-4405             1208232  [0,17,18,19,20,21,22,23,24]  [1.15, 2.02, 1.01, 2.02, 1.01, 2.02, 1.01, 2.02, 1.01] [CALIBRA
TE_BANDPASS#ON_SOURCE,CALIBRATE_WVR#ON_SOURCE]
              14:06:39.7 - 14:06:55.5     5      1 J0519-454               128196  [0,9,10,11,12,13,14,15,16]  [1.15, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48] [CALIBRAT
E_ATMOSPHERE#OFF_SOURCE,CALIBRATE_ATMOSPHERE#ON_SOURCE,CALIBRATE_WVR#OFF_SOURCE,CALIBRATE_WVR#ON_SOURCE]
              14:07:15.5 - 14:09:52.8     6      1 J0519-454               604116  [0,17,18,19,20,21,22,23,24]  [1.15, 2.02, 1.01, 2.02, 1.01, 2.02, 1.01, 2.02, 1.01] [CALIBRA
TE_AMPLI#ON_SOURCE,CALIBRATE_FLUX#ON_SOURCE,CALIBRATE_WVR#ON_SOURCE]
              14:10:34.8 - 14:12:29.5     7      2 J0635-7516              402768  [0,1,2,3,4,5,6,7,8]  [1.15, 2.02, 1.01, 2.02, 1.01, 2.02, 1.01, 2.02, 1.01] [CALIBRATE_POINT
ING#ON_SOURCE,CALIBRATE_WVR#ON_SOURCE]
              14:13:06.8 - 14:13:37.2     8      2 J0635-7516              120816  [0,17,18,19,20,21,22,23,24]  [1.15, 2.02, 1.01, 2.02, 1.01, 2.02, 1.01, 2.02, 1.01] [CALIBRA
TE_PHASE#ON_SOURCE,CALIBRATE_WVR#ON_SOURCE]
              14:14:06.7 - 14:14:22.8     9      3 J0601-7036              128196  [0,9,10,11,12,13,14,15,16]  [1.15, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48] [CALIBRAT
E_ATMOSPHERE#OFF_SOURCE,CALIBRATE_ATMOSPHERE#ON_SOURCE,CALIBRATE_WVR#OFF_SOURCE,CALIBRATE_WVR#ON_SOURCE]
              14:14:42.4 - 14:15:13.0    10      3 J0601-7036              120816  [0,17,18,19,20,21,22,23,24]  [1.15, 2.02, 1.01, 2.02, 1.01, 2.02, 1.01, 2.02, 1.01] [CALIBRA
TE_DELAY#ON_SOURCE,CALIBRATE_WVR#ON_SOURCE]
              14:15:43.5 - 14:15:59.3    11      4 sn1987a                 128196  [0,9,10,11,12,13,14,15,16]  [1.15, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48] [CALIBRAT
E_ATMOSPHERE#OFF_SOURCE,CALIBRATE_ATMOSPHERE#ON_SOURCE,CALIBRATE_WVR#OFF_SOURCE,CALIBRATE_WVR#ON_SOURCE]
              14:16:19.2 - 14:22:07.3    12      4 sn1987a                1329084  [0,17,18,19,20,21,22,23,24]  [1.15, 2.02, 1.01, 2.02, 1.01, 2.02, 1.01, 2.02, 1.01] [OBSERVE
_TARGET#ON_SOURCE]
              14:22:20.9 - 14:22:51.4    13      2 J0635-7516              120816  [0,17,18,19,20,21,22,23,24]  [1.15, 2.02, 1.01, 2.02, 1.01, 2.02, 1.01, 2.02, 1.01] [CALIBRA
TE_PHASE#ON_SOURCE,CALIBRATE_WVR#ON_SOURCE]
              14:23:08.2 - 14:30:00.5    14      4 sn1987a                1570716  [0,17,18,19,20,21,22,23,24]  [1.15, 2.02, 1.01, 2.02, 1.01, 2.02, 1.01, 2.02, 1.01] [OBSERVE
_TARGET#ON_SOURCE]
              14:30:13.2 - 14:30:44.4    15      2 J0635-7516              120852  [0,17,18,19,20,21,22,23,24]  [1.15, 2.02, 1.01, 2.02, 1.01, 2.02, 1.01, 2.02, 1.01] [CALIBRA
TE_PHASE#ON_SOURCE,CALIBRATE_WVR#ON_SOURCE]
              14:31:15.5 - 14:31:31.7    16      3 J0601-7036              128268  [0,9,10,11,12,13,14,15,16]  [1.15, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48] [CALIBRAT
E_ATMOSPHERE#OFF_SOURCE,CALIBRATE_ATMOSPHERE#ON_SOURCE,CALIBRATE_WVR#OFF_SOURCE,CALIBRATE_WVR#ON_SOURCE]
              14:31:51.2 - 14:32:22.0    17      3 J0601-7036              120816  [0,17,18,19,20,21,22,23,24]  [1.15, 2.02, 1.01, 2.02, 1.01, 2.02, 1.01, 2.02, 1.01] [CALIBRA
TE_DELAY#ON_SOURCE,CALIBRATE_WVR#ON_SOURCE]
              14:32:52.2 - 14:33:08.0    18      4 sn1987a                 128196  [0,9,10,11,12,13,14,15,16]  [1.15, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48, 0.48] [CALIBRAT
E_ATMOSPHERE#OFF_SOURCE,CALIBRATE_ATMOSPHERE#ON_SOURCE,CALIBRATE_WVR#OFF_SOURCE,CALIBRATE_WVR#ON_SOURCE]
              14:33:27.9 - 14:39:16.0    19      4 sn1987a                1329084  [0,17,18,19,20,21,22,23,24]  [1.15, 2.02, 1.01, 2.02, 1.01, 2.02, 1.01, 2.02, 1.01] [OBSERVE
_TARGET#ON_SOURCE]
              14:39:29.7 - 14:40:00.0    20      2 J0635-7516              120816  [0, 17, 18, 19, 20, 21, 22, 23, 24]  [1.15, 2.02, 1.01, 2.02, 1.01, 2.02, 1.01, 2.02, 1.01]
[CALIBRATE_PHASE#ON_SOURCE,CALIBRATE_WVR#ON_SOURCE]
          (nRows = Total number of rows per scan)
Fields: 5
  ID   Code Name                RA               Decl           Epoch   SrcId      nRows
  0    none J0538-4405          05:38:50.361560 -44.05.08.93890 J2000   0         2412468
```

INTENTS


Measurement set: uid___A002_Xa48b1f_X1b0.ms - Start time:2015-06-28T13:56:22 End time:2015-06-28T14:39:29
WVR
SIDEBAND
POINTING
ATMOSPHERE
AMPLITUDE
BANDPASS
PHASE
SCIENCE
13h58m
14h13m
14h28m
14h43m
Time

2013.1.00063.S - Topic Summary - Mozilla Firefox@alma-proc01

File Edit View History Bookmarks Tools Help

2013.1.00063.S - Topic Summary

file:///work/nagai/Tutorial/2013.1.00063.S/science_goal.uid___A001_X12e_X27e/group.uid___A001_X12e_X27f/member.u

Google

ALMA 2013.1.00063.S Home By Topic By Task

Flagging Summaries

uid___A002_Xa48b1f_X1b0.ms

spw	DA41	DA42	DA43	DA45	DA47	DA49	DA52	DA53	DA54	DA55	DA57	DA58	DA59	DA60	DA61	DA62	DA64	DV01	DV02	DV03	DV04	DV07
17	23.54	23.54	23.54	23.54	23.54	23.54	23.54	100.00	23.54	23.54	23.54	23.54	23.54	23.54	23.53	23.54	23.54	23.54	23.54	25.04	23.54	23.54
19	23.54	23.54	23.54	23.54	23.54	23.54	23.54	100.00	23.54	23.54	23.54	23.54	23.54	23.54	23.53	23.54	23.54	23.54	23.54	25.04	23.54	23.54
21	23.54	23.54	23.54	100.00	23.54	23.54	23.54	23.54	23.54	23.54	23.54	23.54	23.54	23.54	23.53	23.54	23.54	23.54	23.54	25.04	23.54	23.54
23	23.54	23.54	23.54	100.00	23.54	23.54	23.54	23.54	23.54	23.54	23.54	23.54	23.54	23.54	23.53	23.54	23.54	23.54	23.54	25.04	23.54	23.54

Flagging percentages for Source name: J0538-4405, Intents: WVR,ATMOSPHERE,SIDEBAND,BANDPASS,POINTING

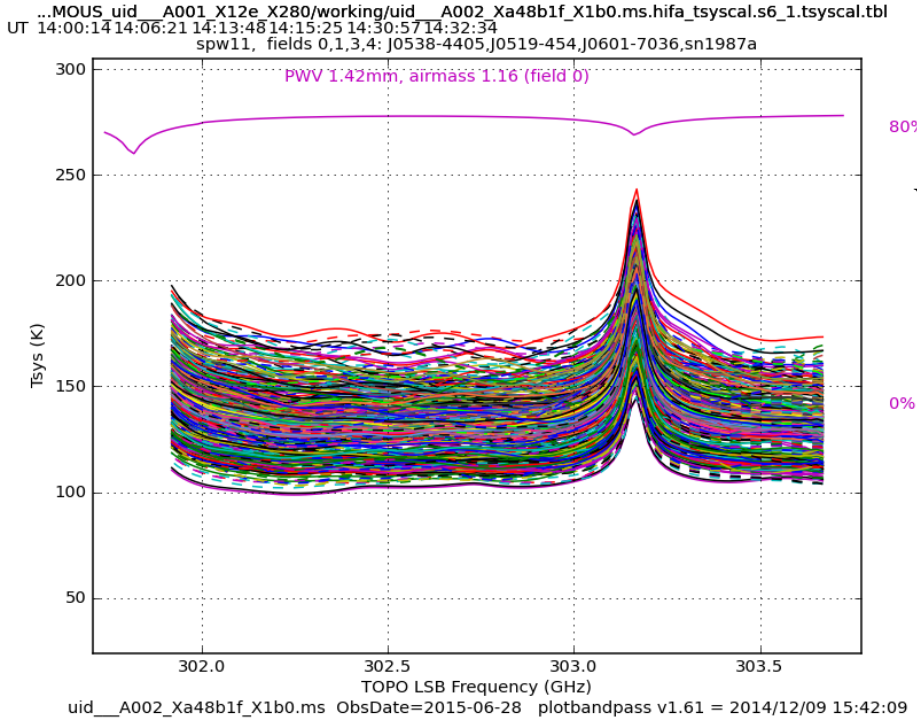
spw	DA41	DA42	DA43	DA45	DA47	DA49	DA52	DA53	DA54	DA55	DA57	DA58	DA59	DA60	DA61	DA62	DA64	DV01	DV02	DV03	DV04	DV07
17	23.07	23.07	23.07	23.07	23.07	23.07	23.07	100.00	23.07	23.07	23.07	23.21	23.07	23.07	23.07	23.07	23.07	23.07	23.07	23.07	23.07	23.07
19	23.07	23.07	23.07	23.07	23.07	23.07	23.07	100.00	23.07	23.07	23.07	23.21	23.07	23.07	23.07	23.07	23.07	23.07	23.07	23.07	23.07	23.07
21	23.07	23.07	23.07	100.00	23.07	23.07	23.07	23.07	23.07	23.07	23.07	23.21	23.07	23.07	23.07	23.07	23.07	23.07	23.07	23.07	23.07	23.07
23	23.07	23.07	23.07	100.00	23.07	23.07	23.07	23.07	23.07	23.07	23.07	23.21	23.07	23.07	23.07	23.07	23.07	23.07	23.07	23.07	23.07	23.07

Flagging percentages for Source name: sn1987a, Intents: WVR,ATMOSPHERE,TARGET

spw	DA41	DA42	DA43	DA45	DA47	DA49	DA52	DA53	DA54	DA55	DA57	DA58	DA59	DA60	DA61	DA62	DA64	DV01	DV02	DV03	DV04	DV07
17	22.92	22.92	22.92	22.92	22.92	22.92	22.92	100.00	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92
19	22.92	22.92	22.92	22.92	22.92	22.92	22.92	100.00	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92
21	22.92	22.92	22.92	100.00	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92
23	22.92	22.92	22.92	100.00	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92	22.92

Flagging percentages for Source name: J0635-7516, Intents: WVR,PHASE,POINTING

Tsys - WVR correction

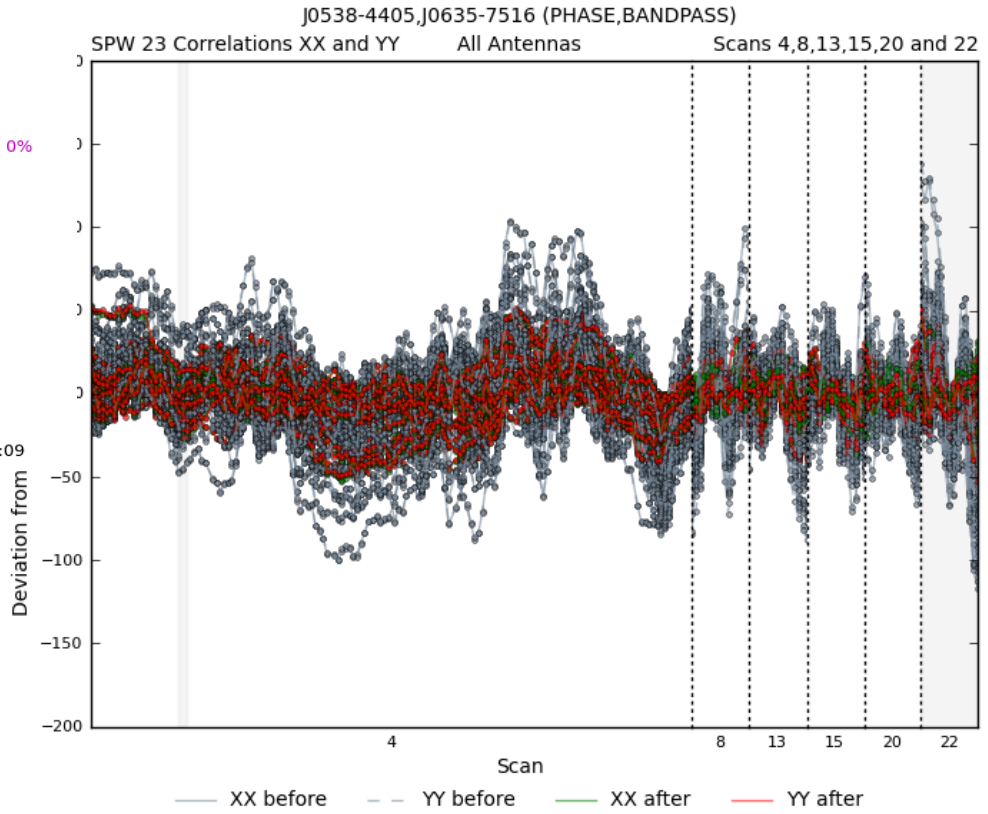


Tsys - sensitivity of each antenna with time (atmosphere & receivers)

80%

0%

WVR correction of atmospheric phase fluctuations

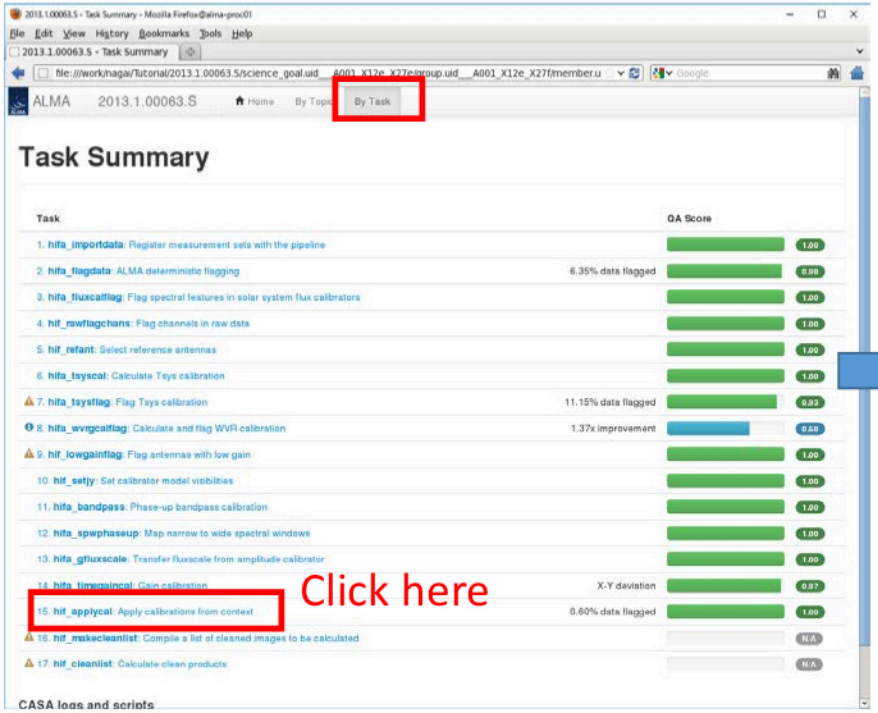


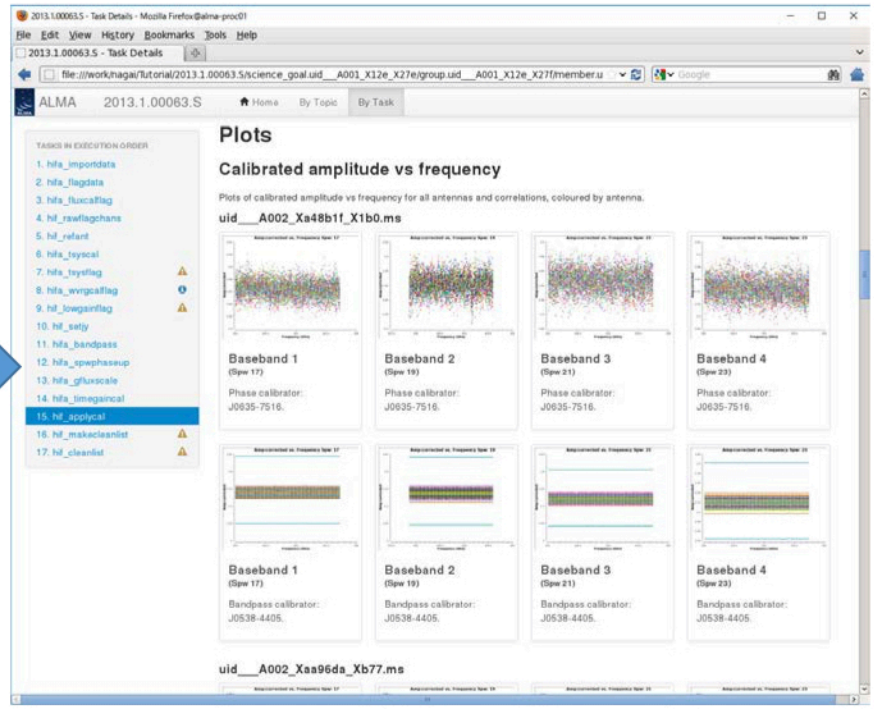
(see also www.alma-allegro.nl/wvr-and-phase-metrics/wvr-scaling/)

EUROPEAN ARC
ALMA Regional Centre || Allegro

CHECK CALIBRATED DATA

- Go to “By Task”, click 15. hifa_applycal

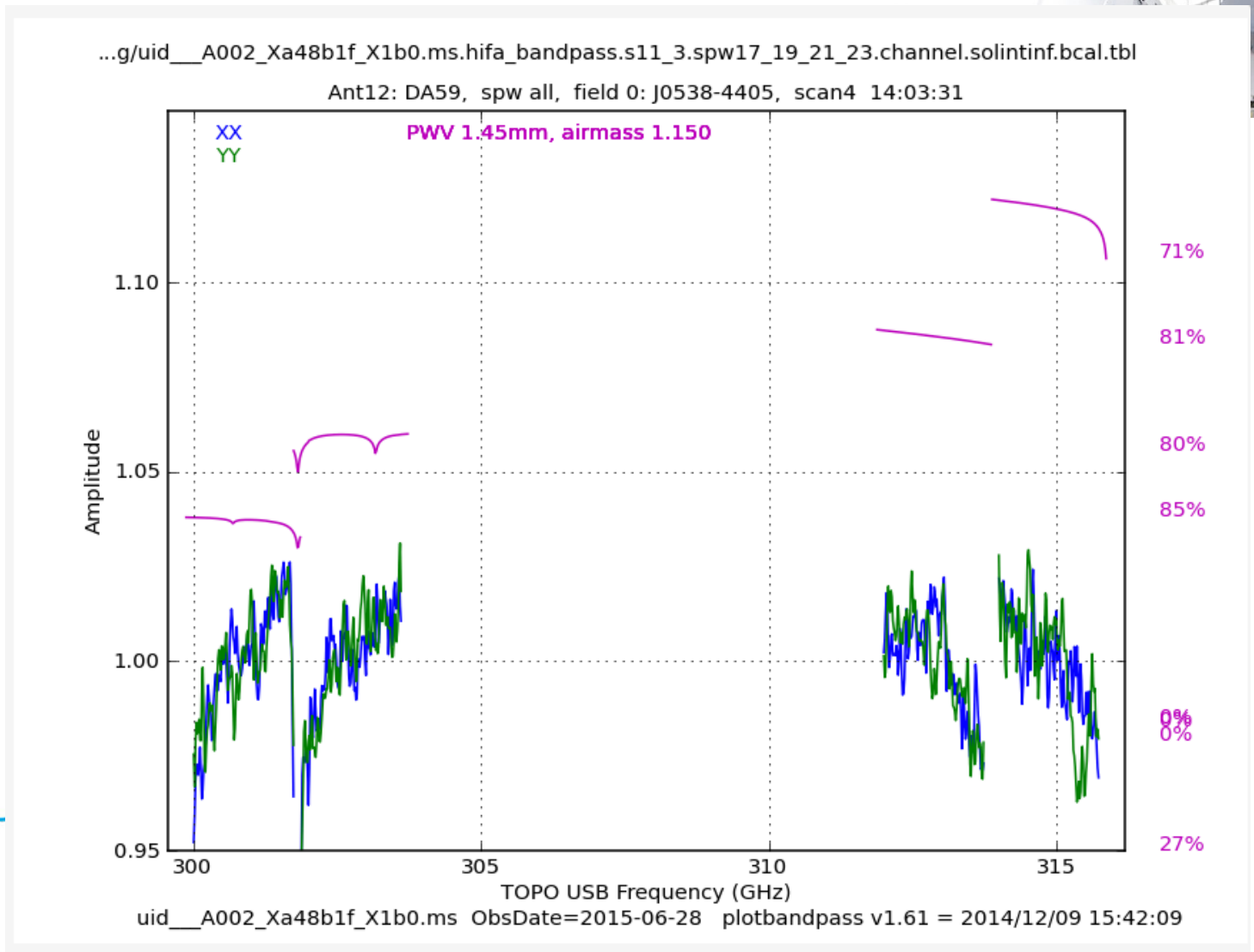




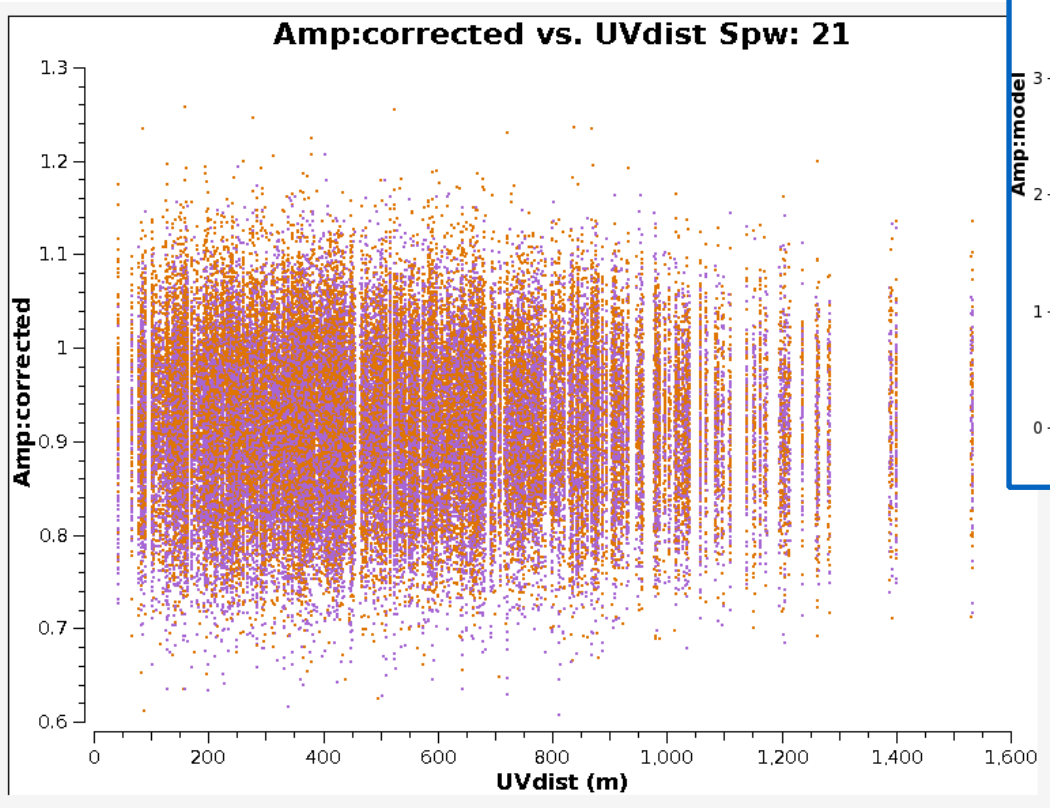
- If any suspicious feature, check relevant calibrations.
  - e.g.,
    - strange bandpass shape -> check bandpass/Tsys calibration
    - Time variable phase -> check gain/wvr calibration

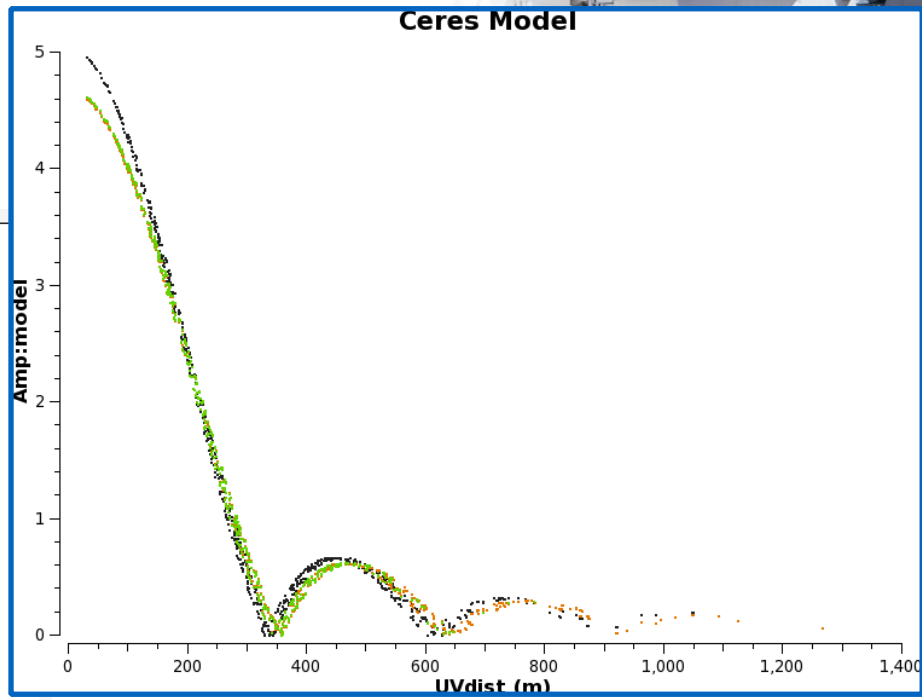
H. Nagai

BANDPASS

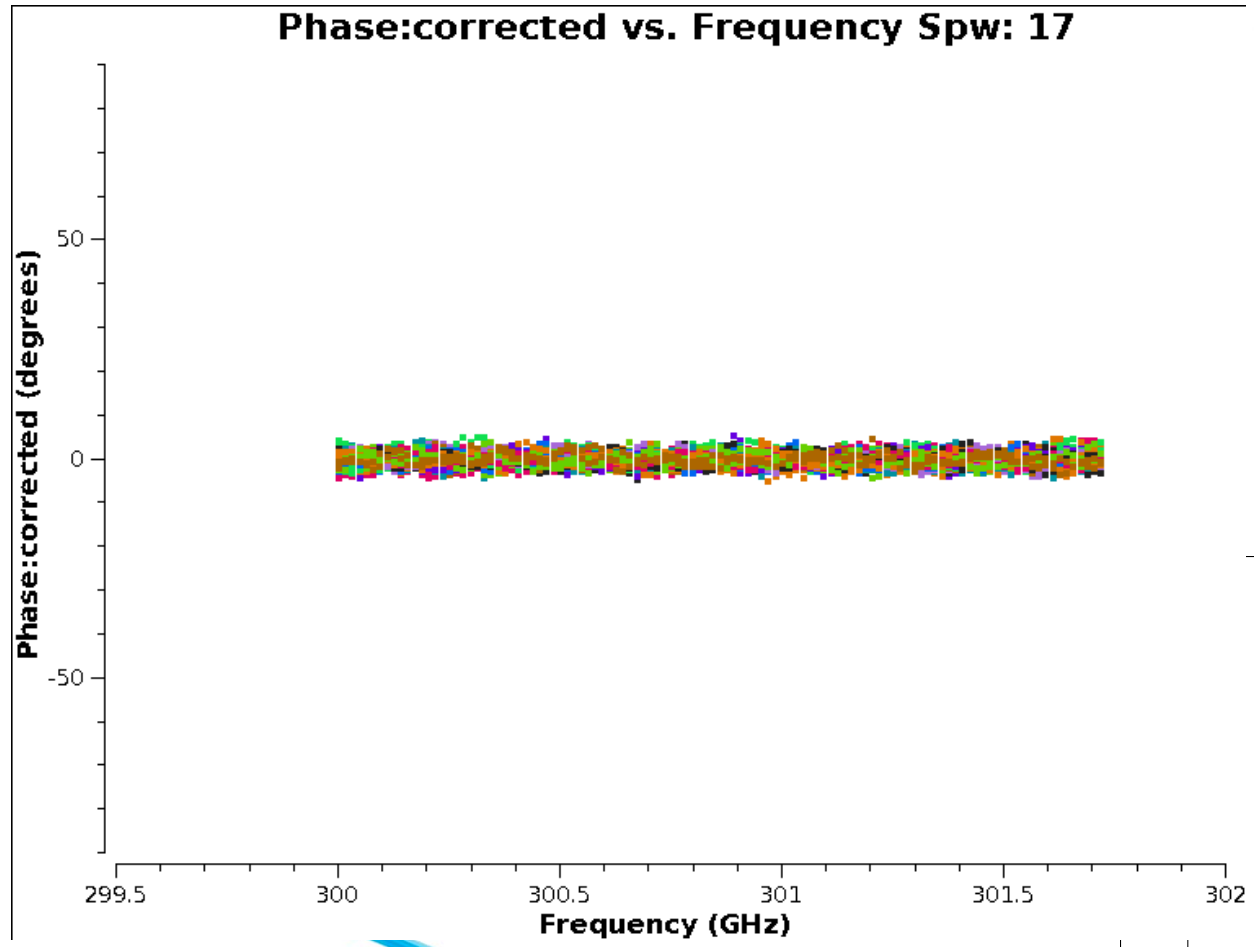

...g/uid___A002_Xa48b1f_X1b0.ms.hifa_bandpass.s11_3.spw17_19_21_23.channel.solintinf.bcal.tbl
Ant12: DA59, spw all, field 0: J0538-4405, scan4 14:03:31
XX
YY
PWV 1.45mm, airmass 1.150
71%
81%
80%
85%
0%
0%
27%
Amplitude
1.10
1.05
1.00
0.95
300
305
310
315
TOPO USB Frequency (GHz)
uid___A002_Xa48b1f_X1b0.ms ObsDate=2015-06-28 plotbandpass v1.61 = 2014/12/09 15:42:09

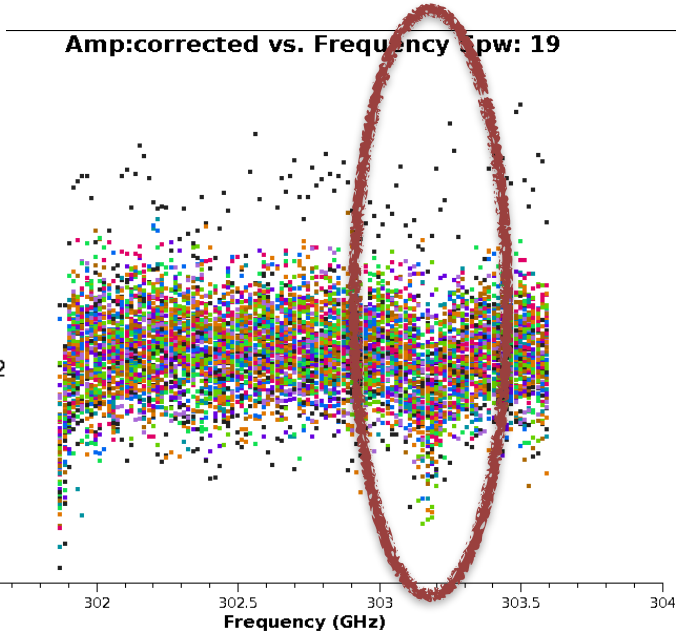
FLUX CALIBRATOR





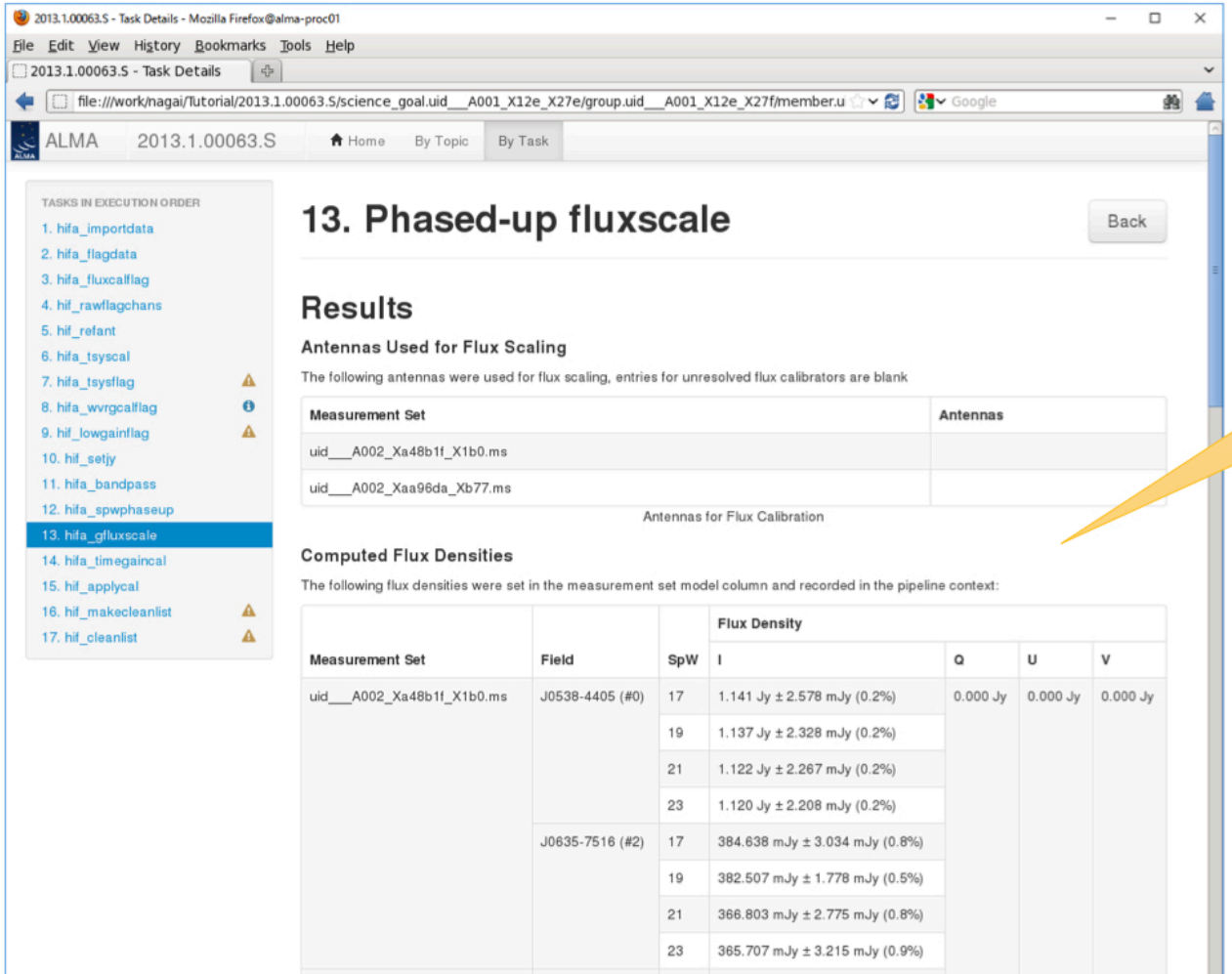
PHASE CAL +CHECK SOURCE





CHECK FLUX CONSISTENCY

- Go to 13.hfa_fluxscale



For this data, a quasar J0519-454 was chosen as a flux calibrator. The pipeline obtained the flux mode of this calibrator from the measurement in ALMA database close in time. See 10.hifa_setjy for the flux model.

This page shows the flux densities of other calibrators derived from the flux scaling using the model of J0519-454.

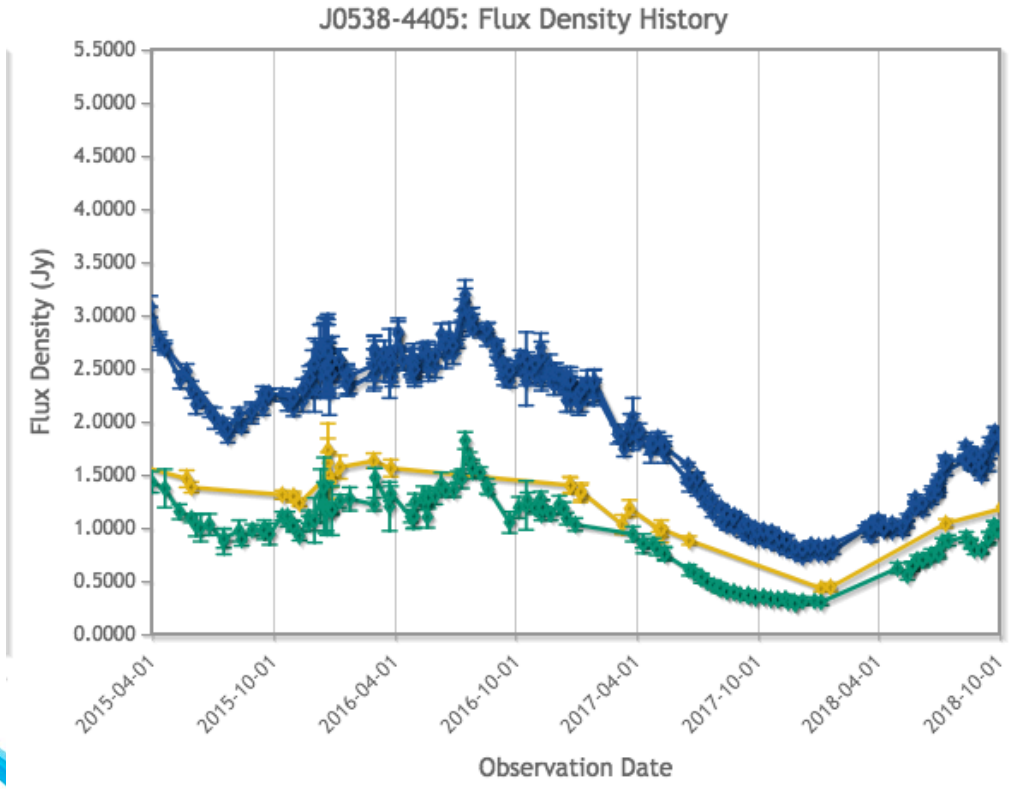
H. Nagai

FLUX CONSISTENCY

```
CASA <2>: aU.getALMAFlux(sourcename='J0538-4405',date='20150629',frequency='301.861GHz')

Using Band 3 measurement: 2.090 +- 0.050 (age=1 days) 103.5 GHz
Using Band 7 measurement: 1.030 +- 0.100 (age=2 days) 343.5 GHz
exact value: -0.589841,  1-sigma extrema: -0.492419, -0.694681,  mean unc=0.101131
Median Monte-Carlo result for 301.861000 = 1.116127 +- 0.226341 (scaled MAD = 0.222051)
Result using spectral index of -0.589841 for 301.861000 GHz from 103.490000 GHz = 1.111537 +- 0.226341 Jy
  Out[2]:
{'ageDifference': 1.0,
 'fluxDensity': 1.1115372967972896,
 'fluxDensityUncertainty': 0.22634072524386856,
 'meanAge': 1.5,
 'monteCarloFluxDensity': 1.1161274327138342,
 'spectralIndex': -0.58984117504330225,
 'spectralIndexUncertainty': 0.11320293251116684}
```

FLUX CONSISTENCY

almascience.eso.org/sc/

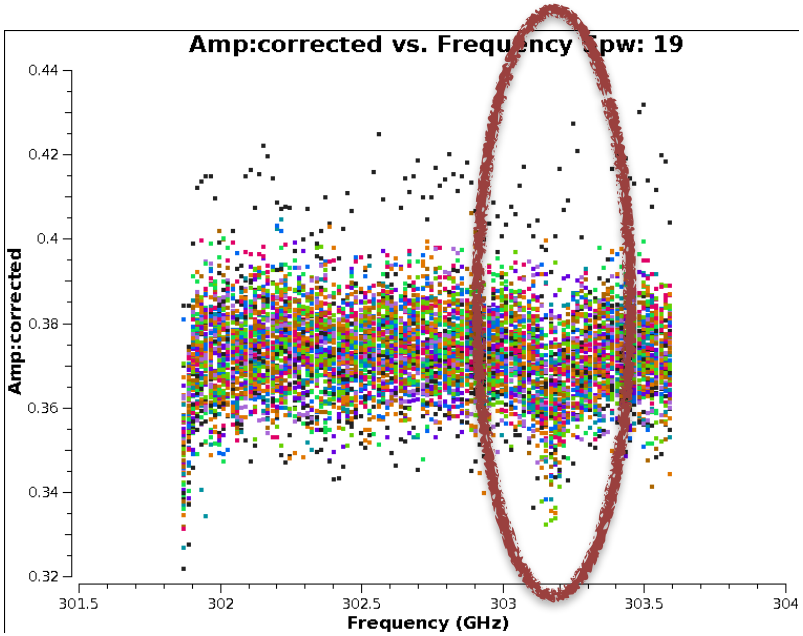


IF PROBLEMS FOUND…

Tweak the pipeline!

e.g, Extra flagging needed (atm line) or flux rescale:

- add extra flagging in calibration/ *flagtemplate.txt and rerun casa_pipescript.py.
- add flagging/setjy commands before imaging.



More details at:

[R. Miura's "Tweaking the Pipeline Script"]

[ALMA Science Pipeline Documentation]

REFERENCES

Allegro CASA Tutorial 2017:
http://www.alma-allegro.nl/alma-data-reduction-casa-training-day-march-3-2017/#presentations

EU ARC CASA Tutorial Nov 2014:
https://www.eso.org/projects/alma/arc/tw/bin/view/External/EUARCCASATutorialNov2014

Data inspection tutorial (H. Nagai):
https://alma-intweb.mtk.nao.ac.jp/~nagai/tutorial/tutorial.pdf

ERIS School 2017:
http://www.astron.nl/eris2017/lectures.php

Interferometric Data Processing Workshop for eMerlin & ALMA:
http://www.alma.ac.uk/index.php/meetings/uk-arc-node-meetings/256-alma-interferometric-data-processing-workshop-dublin-10-12-sept-20178

9th IRAM Interferometry School:
http://www.iram-institute.org/EN/content-page-342-7-67-331-342-0.html

REFERENCES

ALMA Documentation:

https://almascience.eso.org/documents-and-tools

ALMA Technical Handbook:

https://almascience.eso.org/documents-and-tools/cycle6/alma-technical-handbook

ALMA Archive and QA2 Data Products:

https://almascience.eso.org/documents-and-tools#section-3

ALMA Science Pipeline:

https://almascience.eso.org/processing/science-pipeline

EXTRA SLIDES

Step by step ALMA calibration
LT Maud

Calibration - with CASA

MANUAL

PIPELINE

Please navigate to your:

‘uid_XXXXX_scriptForCalibration.py’

OR

pipeline-weblog -> index.html

> cd /lustre/allegro/home/guestX/

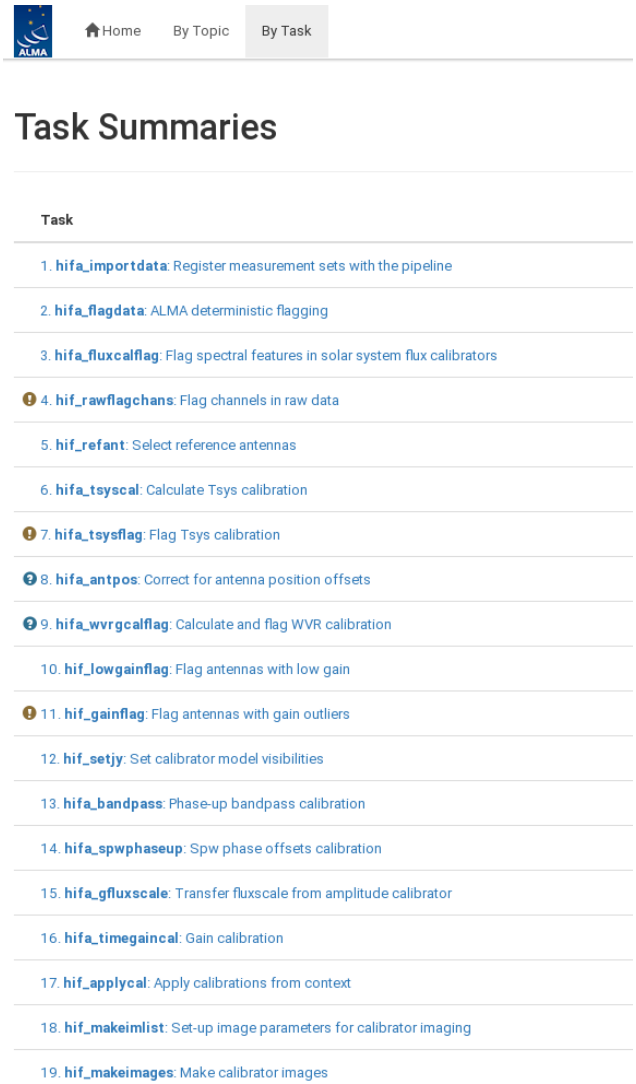
open_CASA_training_day/analysis/guestX

Calibration - with CASA

MANUAL

```
# ALMA Data Reduction Script

# Calibration

thesteps = []
step_title = {0: 'Import of the ASDM',
              1: 'Fix of SYSCAL table times',
              2: 'listobs',
              3: 'A priori flagging',
              4: 'Generation and time averaging of the WVR cal table',
              5: 'Generation of the Tsys cal table',
              6: 'Generation of the antenna position cal table',
              7: 'Application of the WVR, Tsys and antpos cal tables',
              8: 'Split out science SPWs and time average',
              9: 'Listobs, and save original flags',
              10: 'Initial flagging',
              11: 'Putting a model for the flux calibrator(s)',
              12: 'Save flags before bandpass cal',
              13: 'Bandpass calibration',
              14: 'Save flags before gain cal',
              15: 'Gain calibration',
              16: 'Save flags before applycal',
              17: 'Application of the bandpass and gain cal tables',
              18: 'Split out corrected column',
              19: 'Save flags after applycal'}

if 'applyonly' not in globals(): applyonly = False
try:
  print 'List of steps to be executed ...', mysteps
  thesteps = mysteps
except:
  print 'global variable mysteps not set.'
if (thesteps==[]):
  thesteps = range(0,len(step_title))
  print 'Executing all steps: ', thesteps
```

PIPELINE



Calibration - with CASA

MANUAL

```
# ALMA Data Reduction Script

# Calibration

thesteps = []
step_title = {0: 'Import of the ASDM',
              1: 'Fix of SYSCAL table times',
              2: 'listobs',
              3: 'A priori flagging',
              4: 'Generation and time averaging of the WVR cal table',
              5: 'Generation of the Tsys cal table',
              6: 'Generation of the antenna position cal table',
              7: 'Application of the WVR, Tsys and antpos cal tables',
              8: 'Split out science SPWs and time average',
              9: 'Listobs, and save original flags',
              10: 'Initial flagging',
              11: 'Putting a model for the flux calibrator(s)',
              12: 'Save flags before bandpass cal',
              13: 'Bandpass calibration',
              14: 'Save flags before gain cal',
              15: 'Gain calibration',
              16: 'Save flags before applycal',
              17: 'Application of the bandpass and gain cal tables',
              18: 'Split out corrected column',
              19: 'Save flags after applycal'}

if 'applyonly' not in globals(): applyonly = False
try:
  print 'List of steps to be executed ...', mysteps
  thesteps = mysteps
except:
  print 'global variable mysteps not set.'
if (thesteps==[]):
  thesteps = range(0,len(step_title))
  print 'Executing all steps: ', thesteps
```

PIPELINE

Home | By Topic | By Task

Task Summaries

Task
1. **hifa_importdata**: Register measurement sets with the pipeline
2. **hifa_flagdata**: ALMA deterministic flagging
3. **hifa_fluxcalflag**: Flag spectral features in solar system flux calibrators
4. **hif_rawflagchans**: Flag channels in raw data
5. **hif_refant**: Select reference antennas
6. **hifa_tsyscal**: Calculate Tsys calibration
7. **hifa_tsysflag**: Flag Tsys calibration
8. **hifa_antpos**: Correct for antenna position offsets
9. **hifa_wvrgcalflag**: Calculate and flag WVR calibration
10. **hif_lowgainflag**: Flag antennas with low gain
11. **hif_gainflag**: Flag antennas with gain outliers
12. **hif_setjy**: Set calibrator model visibilities
13. **hifa_bandpass**: Phase-up bandpass calibration
14. **hifa_spwphaseup**: Spw phase offsets calibration
15. **hifa_gfluxscale**: Transfer fluxscale from amplitude calibrator
16. **hifa_timegaincal**: Gain calibration
17. **hif_applycal**: Apply calibrations from context
18. **hif_makeimlist**: Set-up image parameters for calibrator imaging
19. **hif_makeimages**: Make calibrator images

Calibration - Tsys, AntPos, WVR

MANUAL

```
os.system('rm -rf uid___A002_Xa44acb_Xadb.ms.tsys')
gencal(vis = 'uid___A002_Xa44acb_Xadb.ms',
  caltable = 'uid___A002_Xa44acb_Xadb.ms.tsys',
  caltype = 'tsys')

# Flagging edge channels

flagdata(vis = 'uid___A002_Xa44acb_Xadb.ms.tsys',
  mode = 'manual',
  spw = '9:0~3;124~127,11:0~3;124~127,13:0~3;124~127,15:0~3;124~127',
  flagbackup = F)
```

- **gencal - this generates the Tsys correction table**
- **flagdata - used to flag the bad response TDM edge channels**

PIPELINE

6. hifa_tsyscal
7. hifa_tsysflag
8. hifa_antpos
9. hifa_wvrgcalflag

Weblog

WHY? : Correct for sky and receiver noise/variation
Plots : in 'QA' (or 'calibration') directory / weblog
'*tsys*plots'

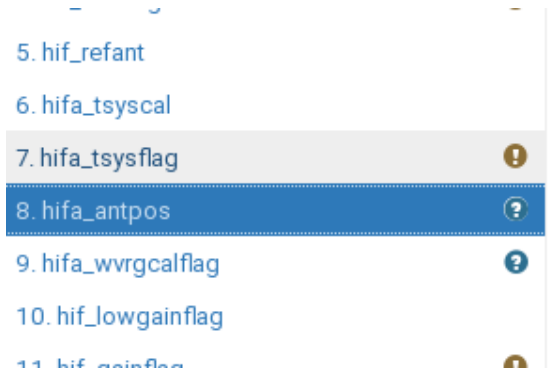
Calibration - Tsys, AntPos, WVR

MANUAL

```
os.system('rm -rf uid___A002_Xa44acb_Xadb.ms.antpos')
gencal(vis = 'uid___A002_Xa44acb_Xadb.ms',
  caltable = 'uid___A002_Xa44acb_Xadb.ms.antpos',
  caltype = 'antpos',
  antenna = 'DA45,DA51,DA53,DA55,DA58,DA60,DA63,DV01,DV03,DV04,DV08,DV
18,DV19,DV24,PM01,PM02,PM03,PM04',
  #parameter = [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]
  parameter = [-2.38629e-04,6.51773e-04,2.24732e-04,2.86116e-05,-5.424
-2.70048e-05,-1.74361e-04,-2.25566e-04,-7.66657e-05,-7.26511e-04,-2.7060
.25562e-05,-4.89141e-04,-1.33922e-04,-3.32959e-04,-2.69390e-04,7.35842e-
8316e-04,-1.34949e-05,-5.56033e-04,-1.81481e-04,4.73415e-05,-4.94519e-04
29e-05,-1.77334e-04,-2.49834e-04,5.08055e-05,2.10684e-05,-2.19155e-04,-9
04,-2.88958e-04,4.48046e-04,-1.07709e-03,-6.35845e-04,-4.46767e-04,-1.07
,1.40105e-04,6.45677e-04,-4.75390e-04,-1.65702e-04,4.36814e-04,-1.03487e
762e-04,-2.25950e-04,4.71491e-05,-4.81501e-04,-1.01380e-04,5.41472e-04,8
04,4.63888e-04,-1.72769e-03,-1.29760e-03,5.52132e-04,-1.73621e-03,-1.023
1.65628e-03,-1.30656e-03])
```

- **gencal - this generates the antpos correction table - comment out/in the parameters**

PIPELINE



Weblog

WHY? : Correct for antenna pad location

Plots : none - tabular list of offsets

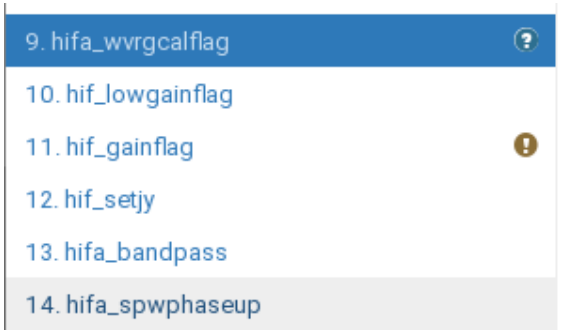
Calibration - Tsys, AntPos, WVR

MANUAL

```
wvrgcal(vis = 'uid___A002_Xa44acb_Xadb.ms',
  caltable = 'uid___A002_Xa44acb_Xadb.ms.wvr',
  spw = [17, 19, 21, 23],
  smooth = '6.048s',
  toffset = 0,
  tie = ['W33A,J1733-1304'],
  statsource = 'W33A')
```

- **wvrgcal - reads the water vapour radiometer signals per antenna and creates the antenna based phase solutions**

PIPELINE

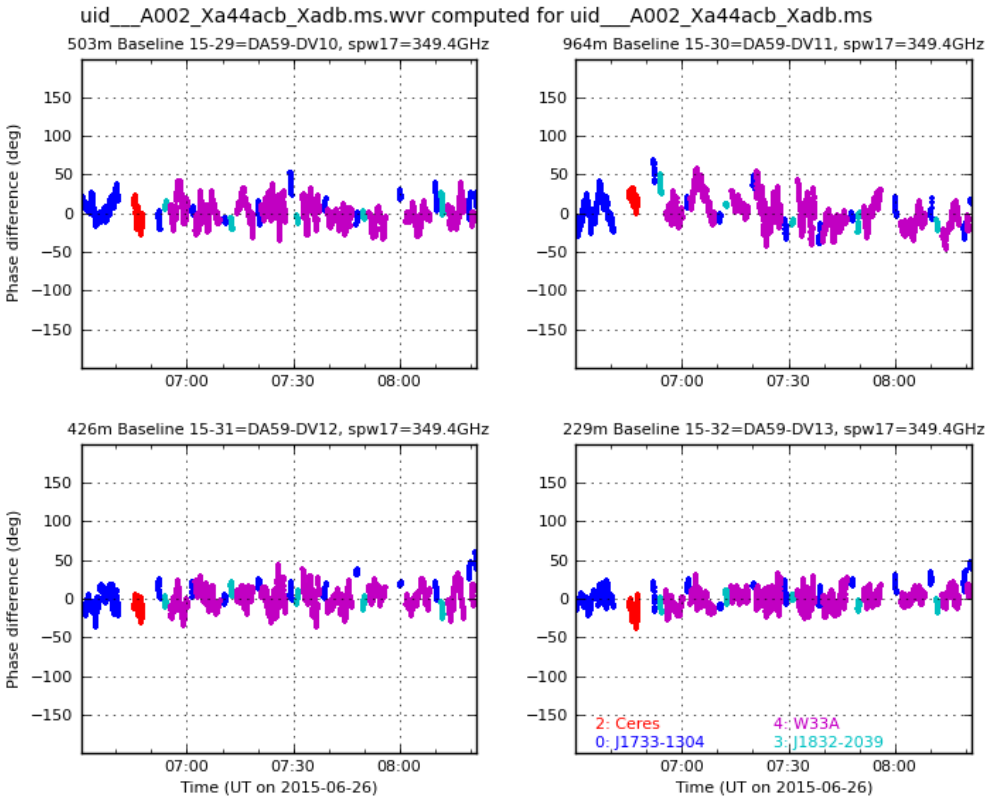


Weblog

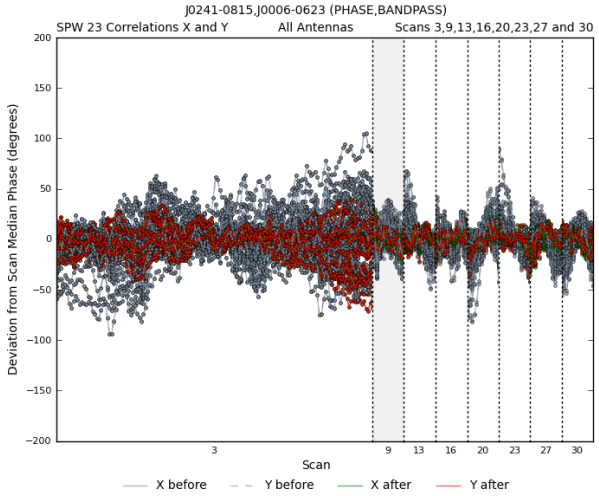
WHY? : Correct for atmospheric phase variations caused by the water vapour - short term
Plots : in 'QA' (or 'calibration') directory / weblog
'*wvr.smooth.plots'

Calibration - Tsys, AntPos, WVR

MANUAL



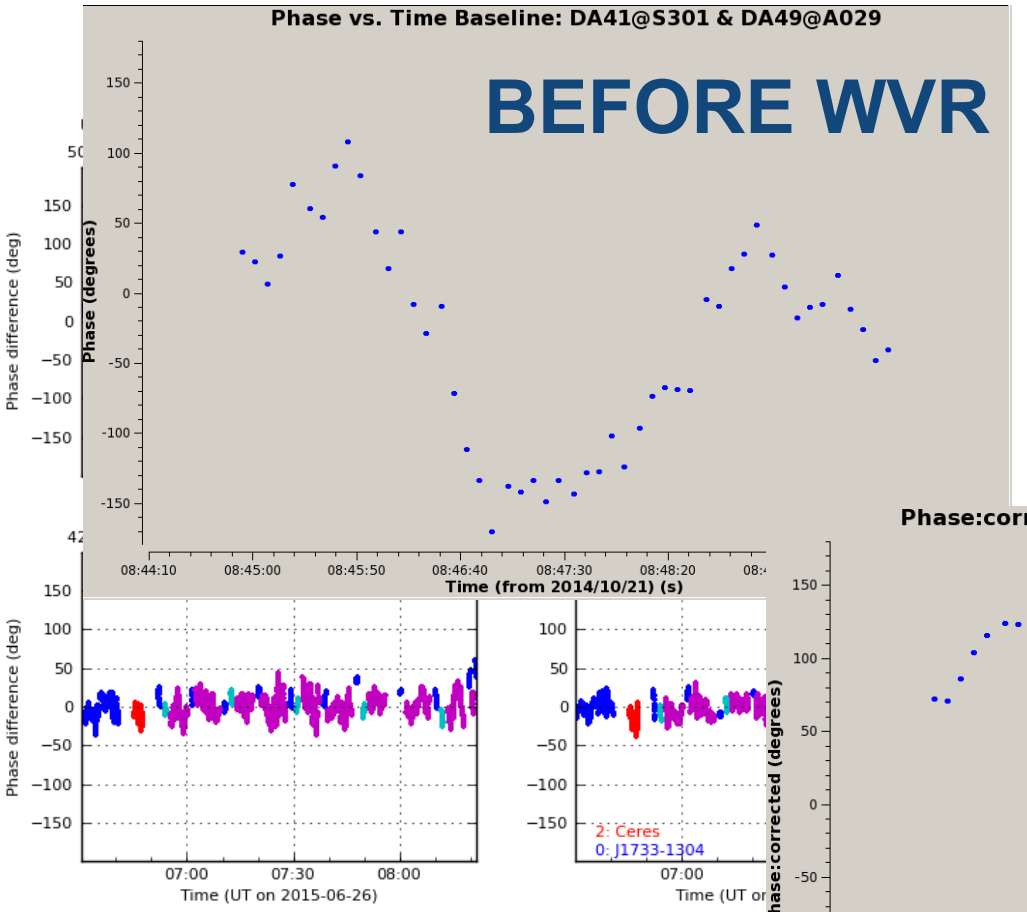
PIPELINE

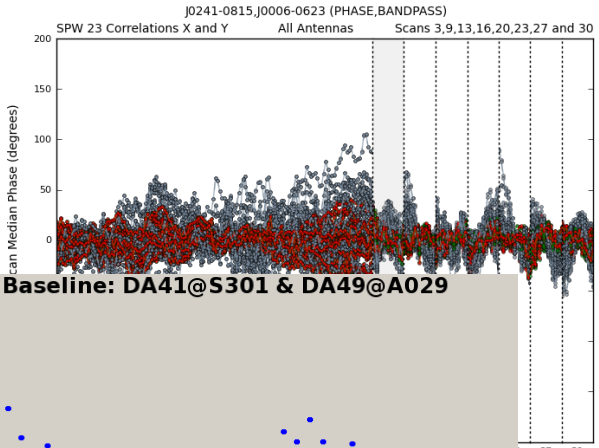


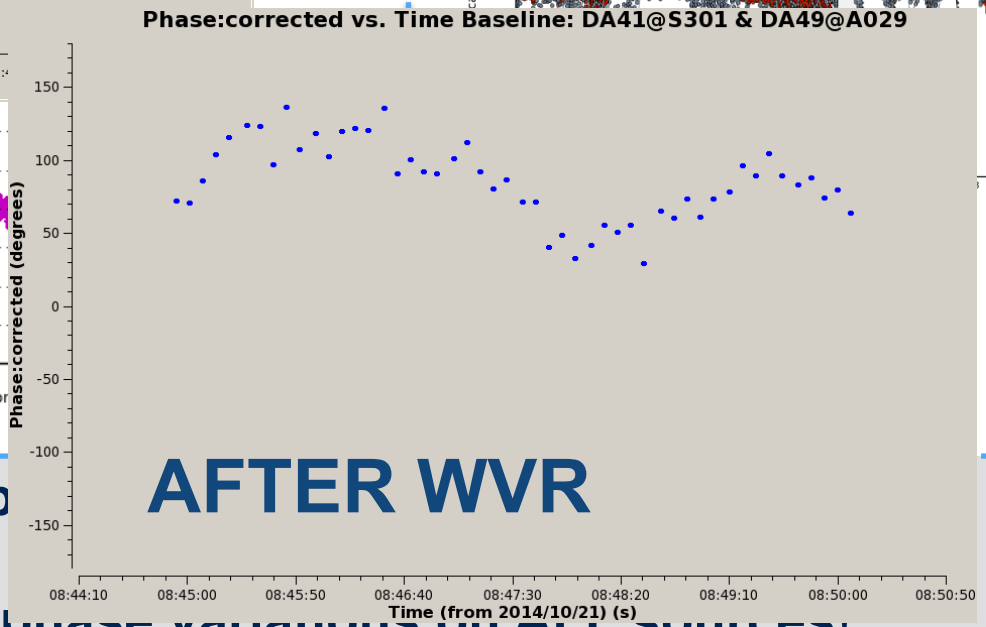
The atmosphere is continually varying - when on source and when on the calibrators - WVR acts to reduce the phase variations on ALL sources!

Calibration - Tsys, AntPos, WVR

PIPELINE







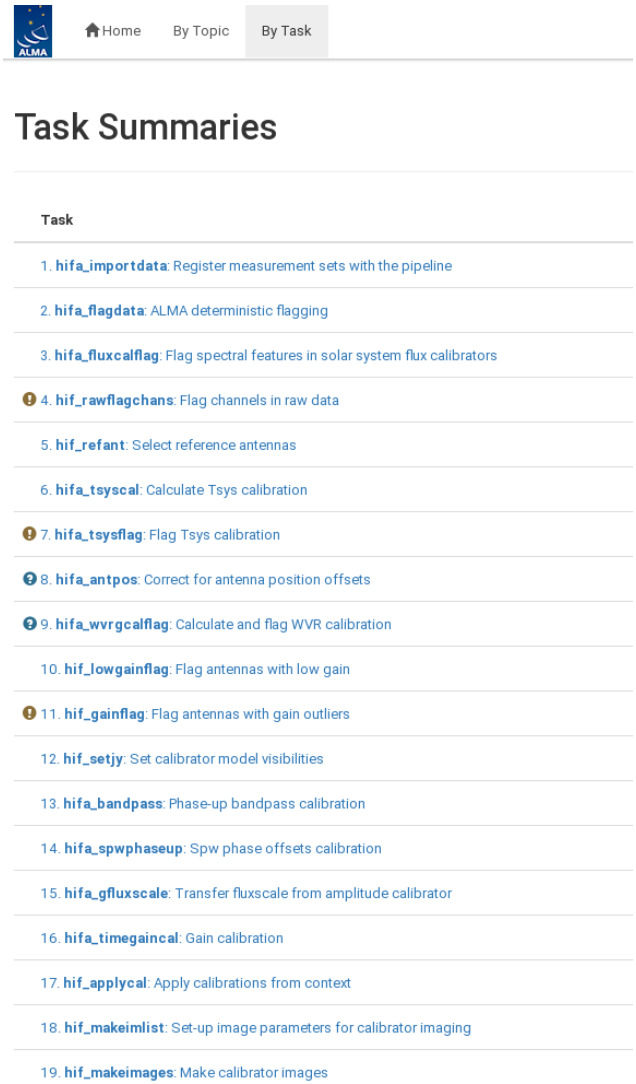
The atmosp
source and
reduce the phase variations on ALL sources!

Calibration - intermediate 'split'

MANUAL

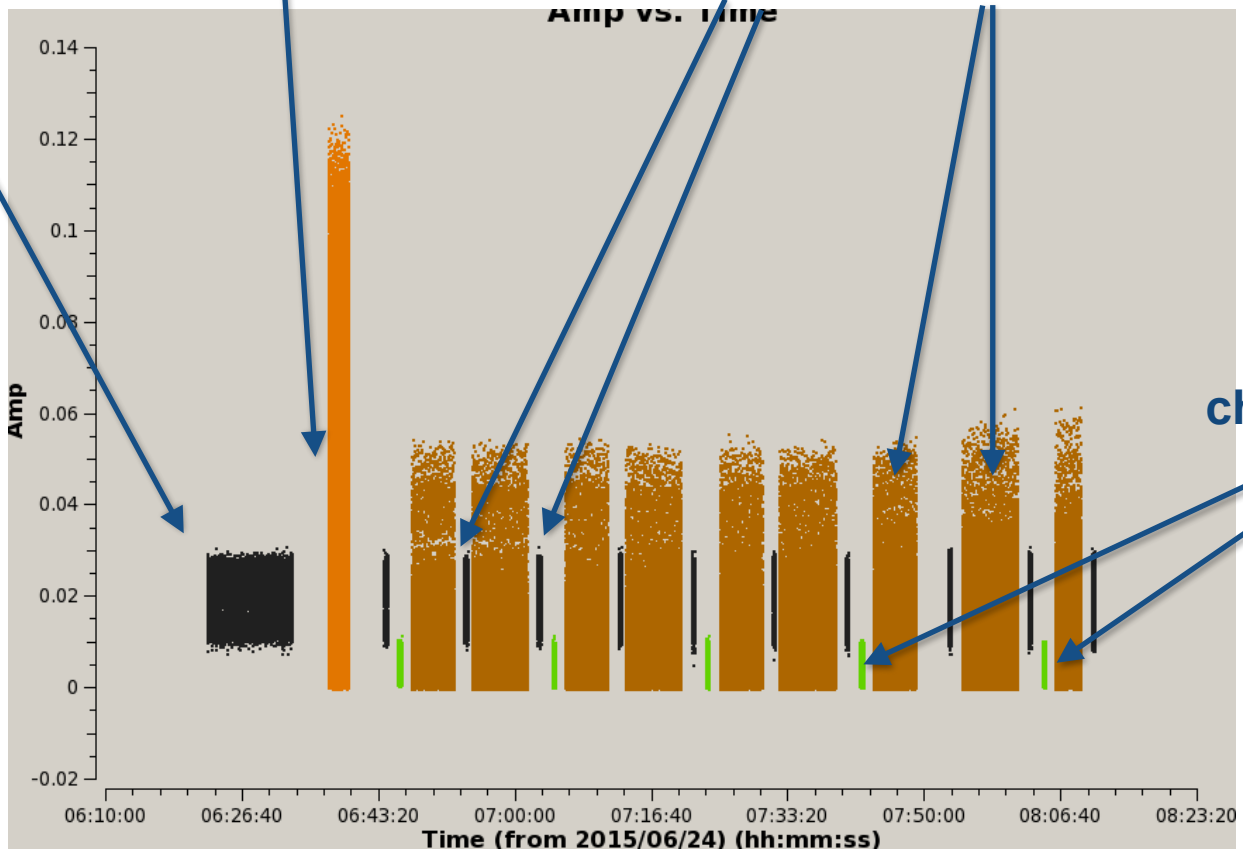
```
# ALMA Data Reduction Script

# Calibration

thesteps = []
step_title = {0: 'Import of the ASDM',
              1: 'Fix of SYSCAL table times',
              2: 'listobs',
              3: 'A priori flagging',
              4: 'Generation and time averaging of the WVR cal table',
              5: 'Generation of the Tsys cal table',
              6: 'Generation of the antenna position cal table',
              7: 'Application of the WVR, Tsys and antpos cal tables',
              8: 'Split out science SPWs and time average',
              9: 'Listobs, and save original flags',
              10: 'Initial flagging',
              11: 'Putting a model for the flux calibrator(s)',
              12: 'Save flags before bandpass cal',
              13: 'Bandpass calibration',
              14: 'Save flags before gain cal',
              15: 'Gain calibration',
              16: 'Save flags before applycal',
              17: 'Application of the bandpass and gain cal tables',
              18: 'Split out corrected column',
              19: 'Save flags after applycal'}

if 'applyonly' not in globals(): applyonly = False
try:
  print 'List of steps to be executed ...', mysteps
  thesteps = mysteps
except:
  print 'global variable mysteps not set.'
if (thesteps==[]):
  thesteps = range(0,len(step_title))
  print 'Executing all steps: ', thesteps
```

PIPELINE



Calibration - The data

Bandpass - Flux Cal - [Gain Cal - Source] x repeat

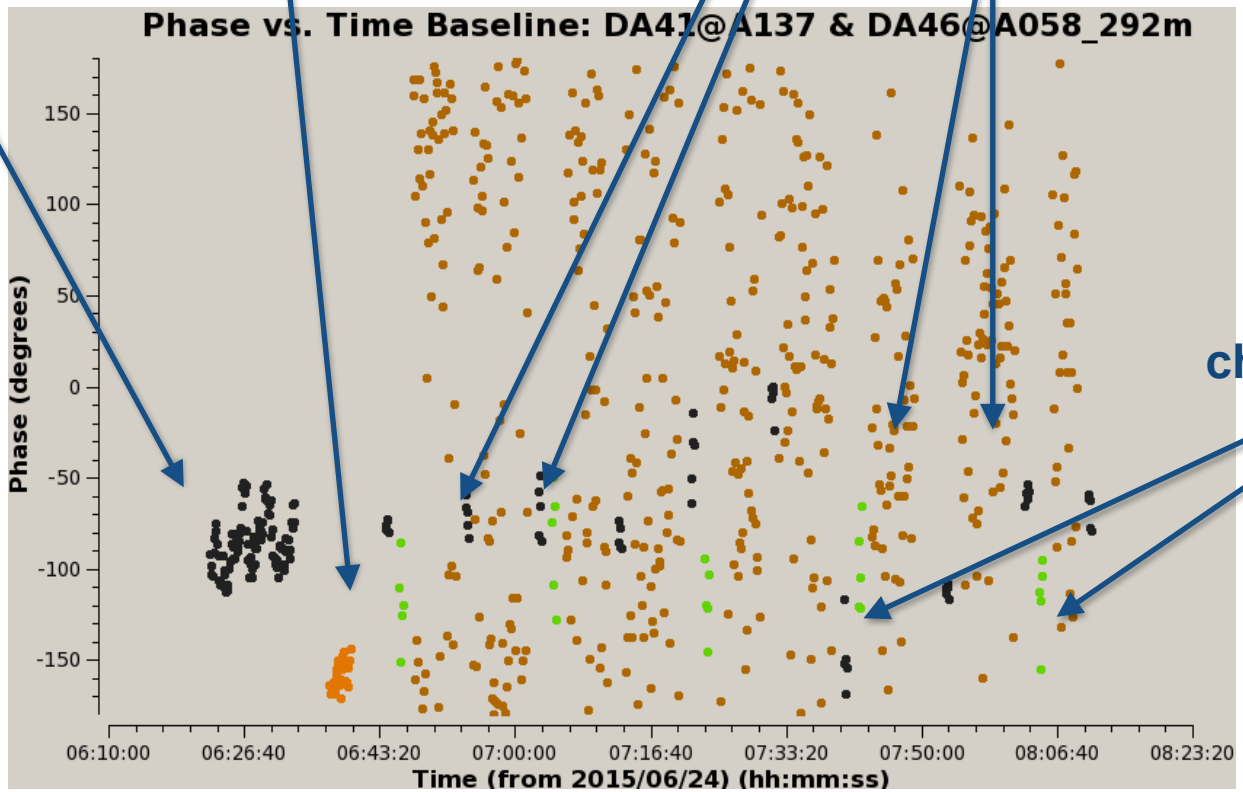


Plots : plotms(vis) - set colorize to 'field' & channel averaging to 999999 - yaxis = 'amp'

weblog - hif_applycal (after calibration only)

Calibration - The data

Bandpass - Flux Cal - [Gain Cal - Source] x repeat



check source

Plots : plotms(vis) - set colorize to 'field' & channel averaging to 999999 - yaxis = 'phase', iteraxis='baseline'
weblog - hif_applycal (after calibration only)

Calibration - Bandpass

- **The Source**
  - **strong source with enough S/N per channel** (caution - v.high S/N if your target is strong in continuum and you search for a weak line - check with your Contact Scientist)
  - **observed 5-10 minutes at start of observing run**
- **The Steps**
  - ***1*** **- must 'phase-up' - simple phase calibration on selected narrow range of channels to correct for decorrelation with time**
  - ***2*** **- apply phase solution 'on-the-fly' and correct for the frequency response**

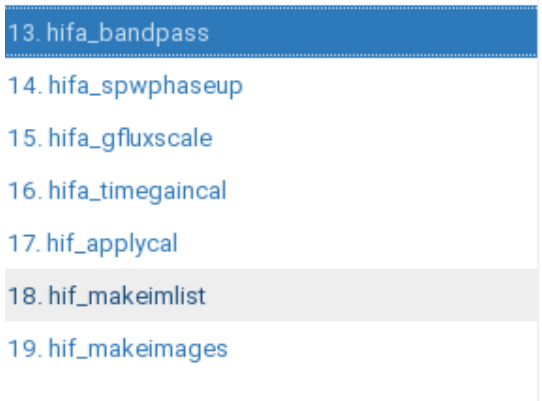
Calibration - Bandpass (1)

MANUAL

```
gaincal(vis = 'uid___A002_Xa44acb_Xadb.ms.split',
  caltable = 'uid___A002_Xa44acb_Xadb.ms.split.ap_pre_bandpass',
  field = '0', # J1733-1304
  spw = '0:1536~2304,1:1536~2304,2:1536~2304,3:1536~2304',
  scan = '1,2,',
  solint = 'int',
  refant = 'DA59',
  calmode = 'p')
```

- **gaincal - creates antenna based solutions to solve phases**
  - **caltable - table that will hold solutions**
  - **spw - select centre of bandwidth (dependent on width)**
  - **calmode - p for phase only**
  - **refant - reference antenna - centrally located**

PIPELINE



Weblog

WHY? :Phase up - solve phases with time !
Plots : in 'QA' (or 'calibration') directory / weblog '*ap_pre_bandpass.plots'

Calibration - Bandpass (1)

MANUAL

```
gaincal(vis = 'uid___A002_Xa44acb_Xadb.ms.split',
  caltable = 'uid___A002_Xa44acb_Xadb.ms.split.ap_pre_bandpass',
  field = '0', # J1733-130
  spw = '0:1536~2304,1:15
  scan = '1,2,',
  solint = 'int',
  refant = 'DA59',
  calmode = 'p')
```

solint = 'int'
INT is the integration time
Usually 6 sec

- **gaincal - creates antenna based solutions to solve phases**
  - **caltable - table that will hold solutions**
  - **spw - select centre of bandwidth (dependent on width)**
  - **calmode - p for phase only**
  - **refant - reference antenna - centrally located**

PIPELINE

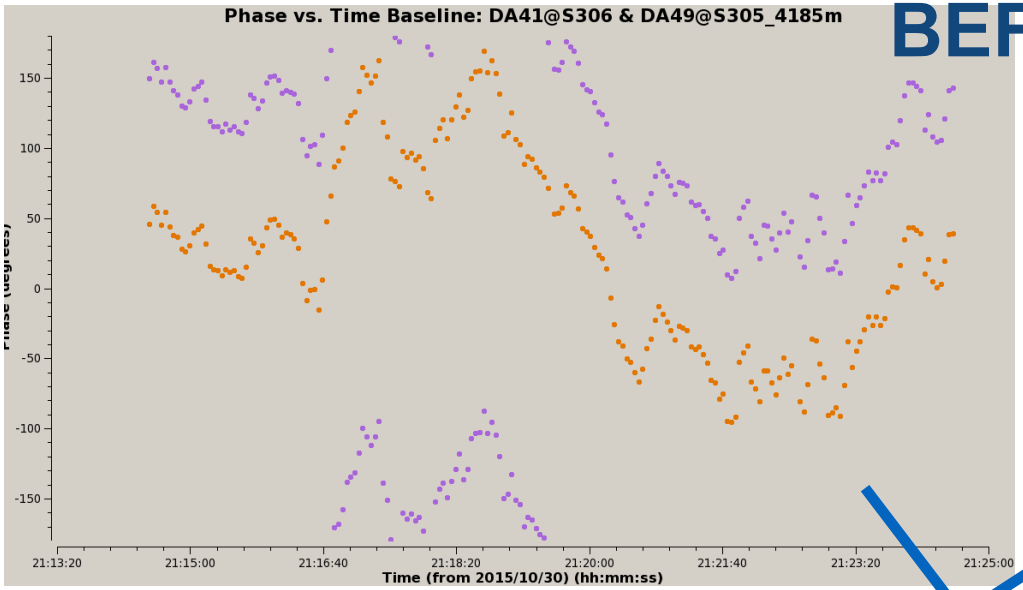
13. hifa_bandpass
14. hifa_spwphaseup
15. hifa_gfluxscale
16. hifa_timegaincal
17. hif_applycal
18. hif_makeimlist
19. hif_makeimages

Weblog

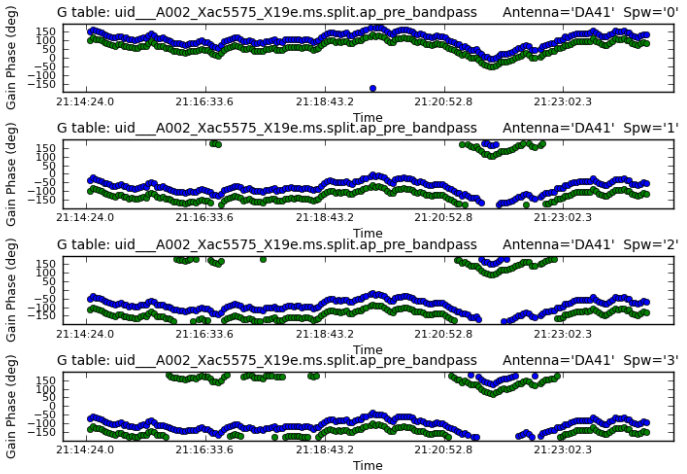
WHY? :Phase up - solve phases with time !
Plots : in 'QA' (or 'calibration') directory / weblog
'*ap_pre_bandpass.plots'

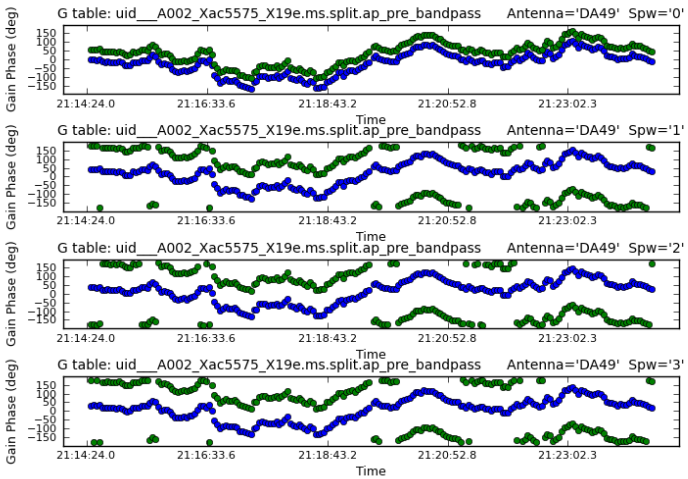
Calibration - Bandpass (1)

BEFORE

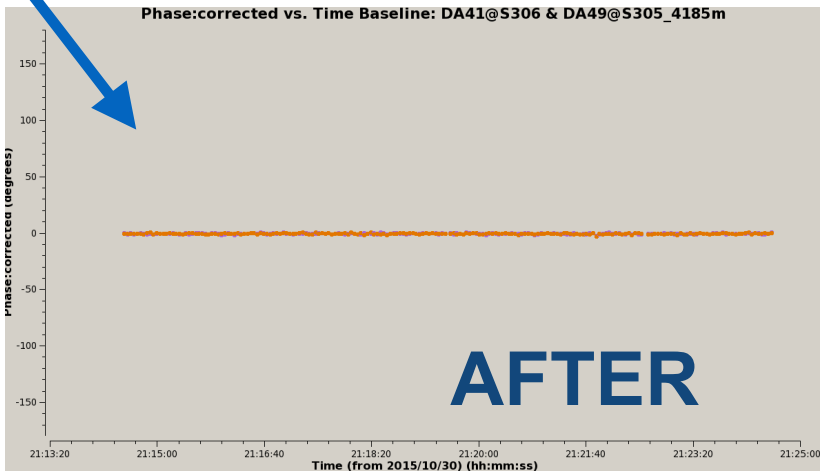


DA41 - Soln.





DA49 - Soln.



AFTER

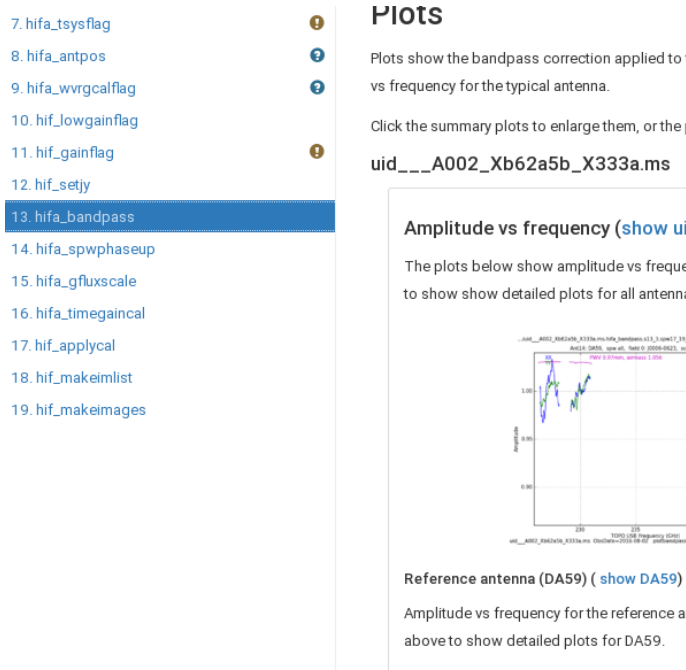
Calibration - Bandpass (2)

MANUAL

```
bandpass(vis = 'uid___A002_Xa44acb_Xadb.ms.split',
  caltable = 'uid___A002_Xa44acb_Xadb.ms.split.bandpass_smooth20ch',
  field = '0', # J1733-1304
  scan = '1,2,',
  solint = 'inf,4MHz',
  combine = 'scan',
  refant = 'DA59',
  solnorm = True,
  bandtype = 'B',
  gaintable = 'uid___A002_Xa44acb_Xadb.ms.split.ap_pre_bandpass')
```

- **bandpass - creates antenna based solutions for amp & phase with Freq.**
  - **caltable - table that will hold solutions**
  - **combine - scan - all data needs to be combined for each SPW**
  - **bandtype - B for bandpass. BP for B-poly if you bandpass is noisy - use with care**

PIPELINE



Weblog

WHY? :Solve for the frequency dependence
Plots : in 'QA' (or 'calibration') directory / weblog
'*bandpass.plots' & '*bandpass_smoothXXXX.plots'

Calibration - Bandpass (2)

MANUAL

```
bandpass(vis = 'uid___A002_Xa44acb_Xadb.ms.split',
  caltable = 'uid___A002_Xa44acb_Xadb.ms.split.bandpass_smooth20ch',
  field = '0', # J1733-1304
  scan = '1,2,',
  solint = 'inf,4MHz',
  combine = 'scan',
  refant = 'DA59',
  solnorm = True,
  bandtype = 'B',
  gaintable = 'uid___A002_Xa44acb_Xadb.ms.split.ap_pre_bandpass')
```

- **solint** - inf to average in time 4MHz or XXch for averaging in frequency

- **bandpass** - creates antenna based solutions for amp & phase with Freq.
  - **caltable** - table that will hold solutions
  - **combine** - scan - all data needs to be combined for each SPW
  - **bandtype** - B for bandpass. BP for B-poly if you bandpass is noisy - use with care

PIPELINE

7. hifa_tsysflag
8. hifa_antpos
9. hifa_wvrgcalflag
10. hif_lowgainflag
11. hif_gainflag
12. hif_setjy
13. hifa_bandpass
14. hifa_spwphaseup
15. hifa_gfluxscale
16. hifa_timegaincal
17. hif_applycal
18. hif_makeimlist
19. hif_makeimages

Plots

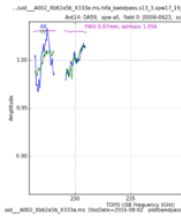
Plots show the bandpass correction applied to
vs frequency for the typical antenna.

Click the summary plots to enlarge them, or the

uid___A002_Xb62a5b_X333a.ms

Amplitude vs frequency (show ui

The plots below show amplitude vs freque
to show show detailed plots for all antenna



Reference antenna (DA59) (show DA59)

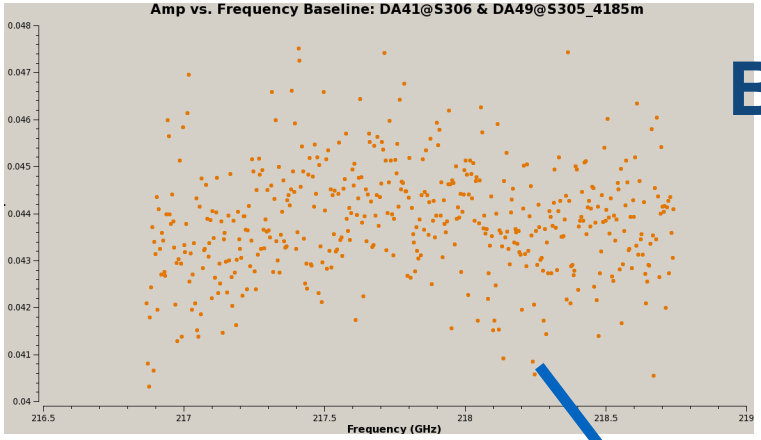
Amplitude vs frequency for the reference a
above to show detailed plots for DA59.

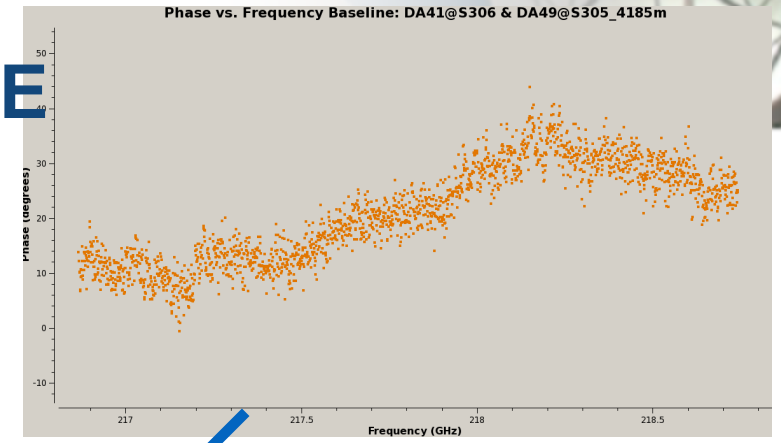
Weblog

WHY? :Solve for the frequency dependence
Plots : in 'QA' (or 'calibration') directory / weblog
'*bandpass.plots' & '*bandpass_smoothXXXX.plots'

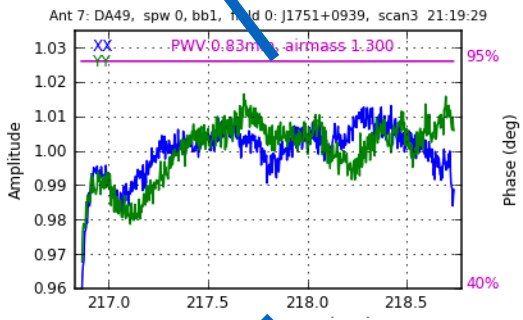
Calibration - Bandpass (2)

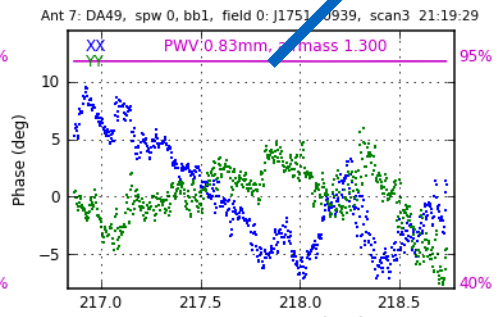
BEFORE





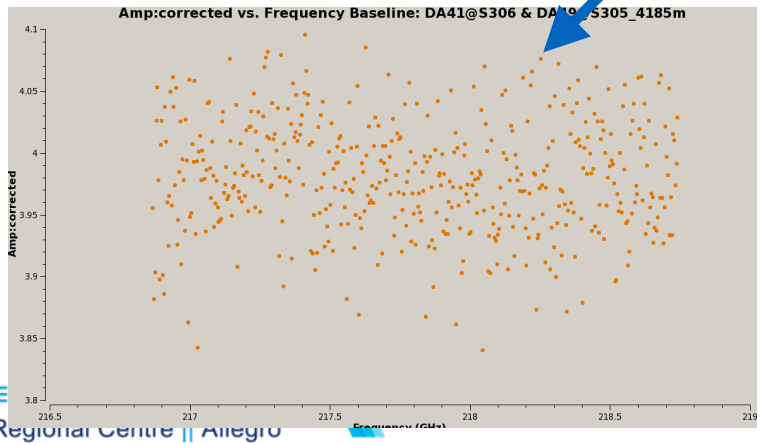
AMP.

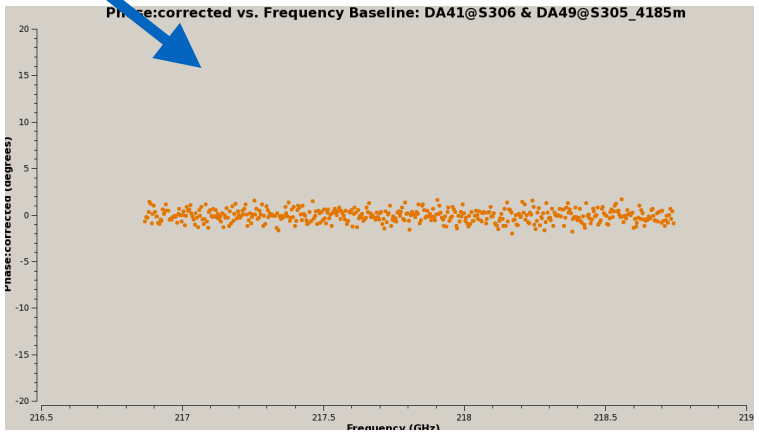




PHASE

AFTER





Calibration - Flux

- The Source
  - Solar System Object (SSO) - caution too resolved is an issue
  - QSO known/monitored source
- The Steps
  - *1* - use 'setjy' to set the flux scaling to refer to later
  - *GC1*- phase up all sources required for flux scaling to solve for decorrelation - flux cal, BP cal, gain cal (caution with resolved sources!!!)
  - *GC2*- apply the phase-up 'on-the-fly' and solve the amplitudes
  - 2 - scale the amplitude gains according to the output flux of the flux calibrator - bootstrapping !

Gain cal steps

Calibration - Flux (1)

MANUAL

SSO

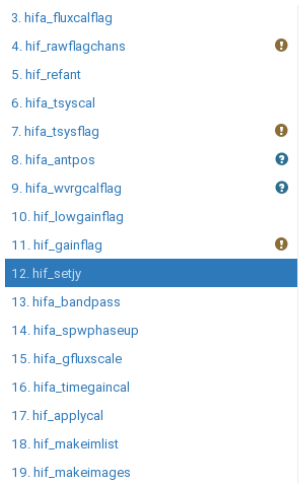
```
setjy(vis = 'uid___A002_Xa44acb_Xadb.ms.split',
  field = '2', # Ceres
  spw = '0,1,2,3',
  standard = 'Butler-JPL-Horizons 2012')
```

QSO

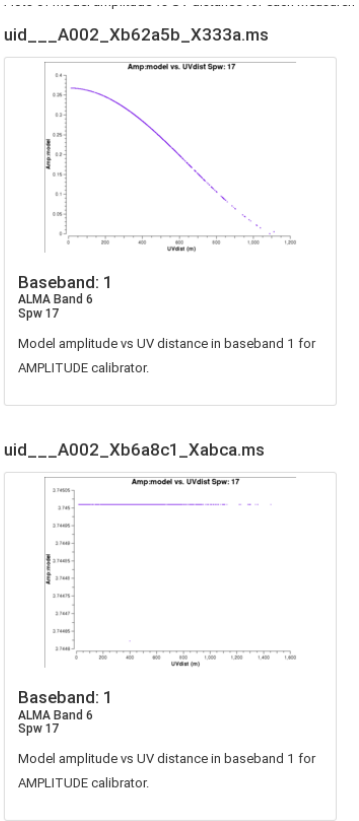
```
setjy(vis = 'uid___A002_Xac5575_X19e.ms.split',
  standard = 'manual',
  field = 'J1751+0939',
  fluxdensity = [4.04889627574, 0, 0, 0],
  spix = -0.476483659328,
  reffreq = '226.451094779GHz')
```

- **setjy - sets the flux scaling for the flux calibrator source**
- **standard - the model OR manual**

PIPELINE



Weblog



WHY? :Setting the correct flux scale to use later
Plots : in 'QA' directory / weblog

Calibration - Flux (1)

MANUAL PIPELINE

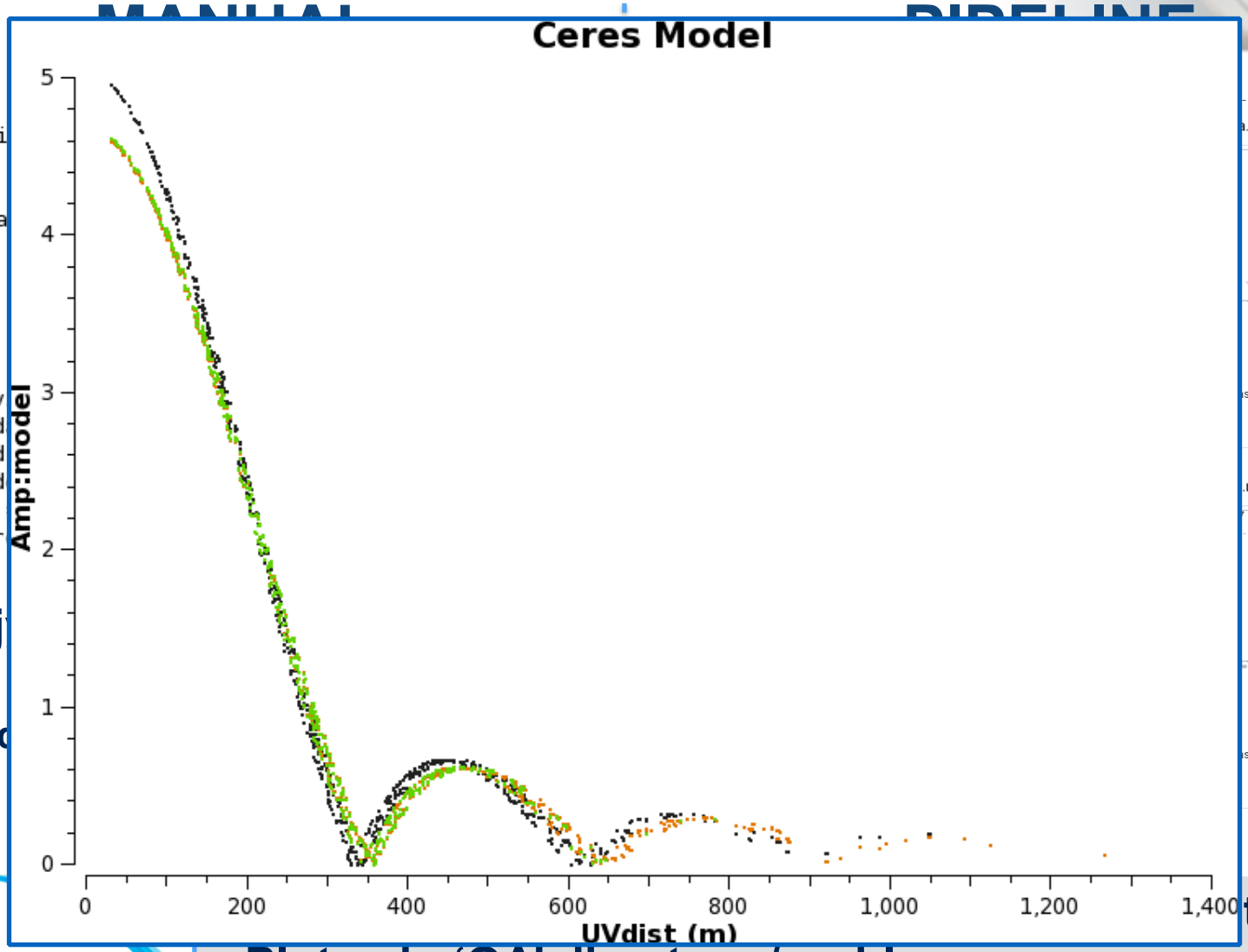
SSO

```
setjy(vi
  field
  spw =
  standa
```

QSO

```
setjy(v
  stand
  field
  fluxd
  spix
  reffr
```

- setj
  the
- stand



ter

Plots : in 'QA' directory / weblog

Calibration - Gains

- The Source
  - QSO point source - known amp and phase visibilities
- What?
  - Short term phases
  - Long term amplitudes
  - ~Long term phases - per visit to phase calibrator
- The steps
  - *1* - 'phase-up' to solve for decorrelation due to rapid phase changes cause be atmosphere - *required to get the correct amplitudes
  - *2* - Slow variation of amplitude with time - solved with 'phase-up' solutions applied 'on-the-fly'
  - *3* - Phase variations due to atmospheric changes are monitored by the gain calibrator and interpolated to the source

Calibration - Gains (1) - phase 'int'

MANUAL

```
gaincal(vis = 'uid___A002_Xac5575_X19e.ms.split',
  caltable = 'uid___A002_Xac5575_X19e.ms.split.phase_int',
  field = '0,1,2', # J1751+0939,J1924+1540  - J1922+1530
  solint = 'int',
  refant = 'DV10',
  gaintype = 'G',
  calmode = 'p',
  gaintable = 'uid___A002_Xac5575_X19e.ms.split.bandpass')
```

- **gaincal - creates the antenna based phase solutions**
  - **field - ALL calibrator sources**
  - **solint - 'int' should be selected for best phase solutions**
  - **gaintable - apply bandpass OTF**

PIPELINE

14. hifa_spwphaseup
15. hifa_gfluxscale
16. hifa_timegaincal
17. hif_applycal
18. hif_makeimlist
19. hif_makeimages

Weblog

WHY? :Correct for all phase variations in time
Plots : in 'QA' directory / weblog
'*split.phase_int.plots'

Calibration - Gains (1) - phase 'int'

MANUAL

```
gaincal(vis = 'uid___A002_Xac5575_X19e.ms.split',
  caltable = 'uid___A002_Xac5575_X19e.ms.split.phase_int',
  field = '0,1,2', # J1751+0939,J1924+1540  - J1922+1530
  solint = 'int',
  refant = 'DV10',
  gaintype = 'G',
  calmode = 'p',
  gaintable = 'uid___A002_Xac5575_X19e.ms.split.bandpass')
```

- **gaincal - creates the antenna based phase solutions**
  - **field - ALL calibrator sources**
  - **solint - 'int' should be selected for best phase solutions**
  - **gaintable - apply bandpass OTF**

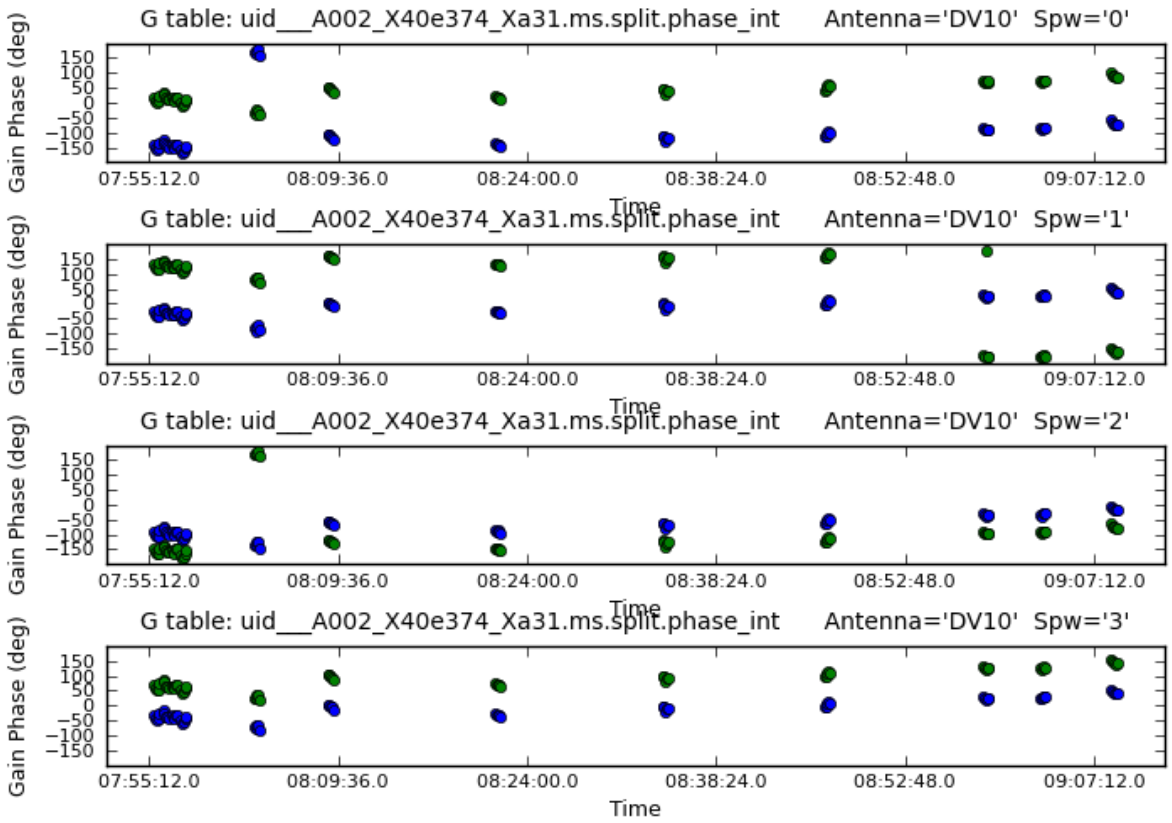
PIPELINE

14. hifa_spwphaseup
15. hifa_gfluxscale
16. hifa_timegaincal
17. hif_applycal
18. hif_makeimlist
19. hif_makeimages

Weblog

If one (or more) of your calibrators is weak (HF data) this can be increased or multiple gain tables are required, be aware this could cause fluxes to have a larger uncertainty unless atmosphere is stable

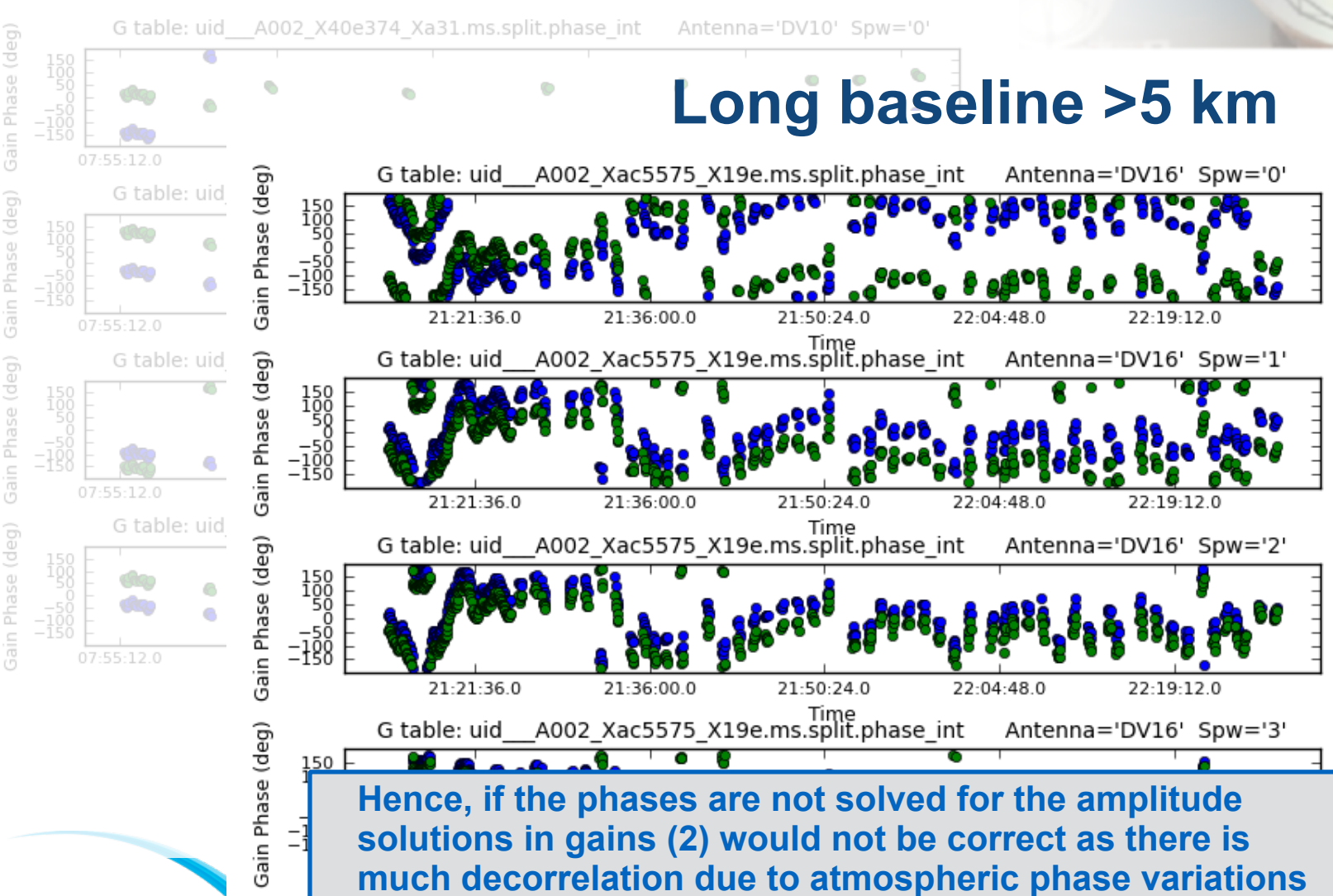
Calibration - Gains (1) - phase 'int'



~short baseline <1 km

Calibration - Gains (1) - phase 'int'

Long baseline >5 km



Hence, if the phases are not solved for the amplitude solutions in gains (2) would not be correct as there is much decorrelation due to atmospheric phase variations (worse for longer baselines and higher frequencies)

Calibration - Gains (2) - amp 'inf'

MANUAL

```
gaincal(vis = 'uid___A002_Xac5575_X19e.ms.split',
  caltable = 'uid___A002_Xac5575_X19e.ms.split.ampli_inf',
  field = '0,1,2', # J1751+0939,J1924+1540  J1922+1530
  solint = 'inf',
  refant = 'DV10',
  gaintype = 'T',
  calmode = 'a',
  gaintable = ['uid___A002_Xac5575_X19e.ms.split.bandpass',
5_X19e.ms.split.phase_int'])
```

- **gaincal - creates the antenna based amp solutions**
  - **field - ALL calibrator sources**
  - **solint - 'inf' i.e. per scan/visit to the gain cal**
  - **gaintable - apply bandpass AND the 'int' phase-up solutions OTF**

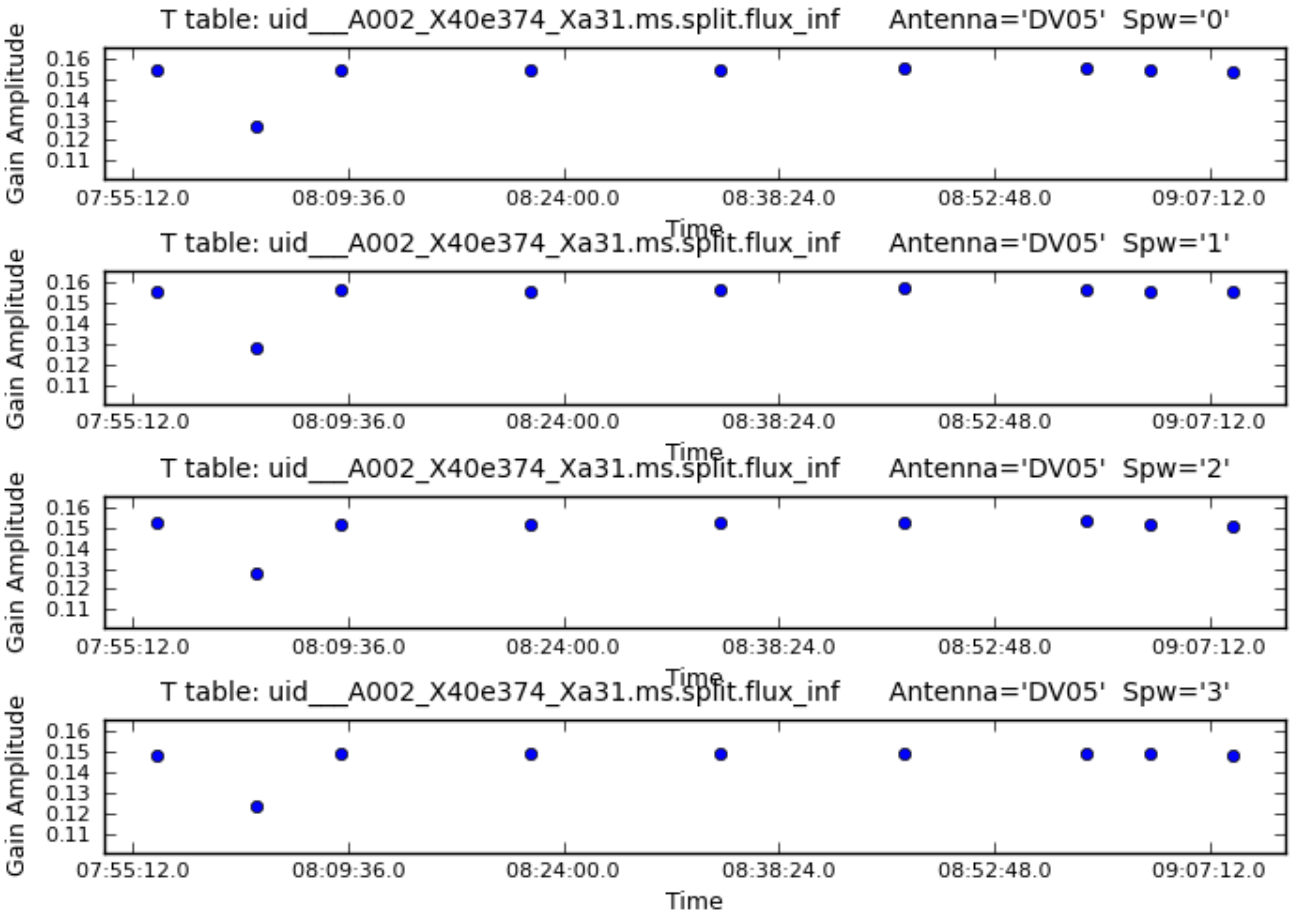
PIPELINE

15. hifa_gfluxscale
16. hifa_timegaincal
17. hif_applycal
18. hif_makeimlist
19. hif_makeimages

Weblog

WHY? :Correct for slow amplitude variations in time
Plots : in 'QA' directory / weblog
'*split.ampli_int.plots' OR '*split.flux_inf.plots'

Calibration - Gains (2) - amp 'inf'



Solutions ONLY every visit to the gain calibrator

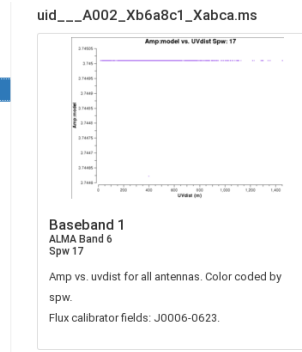
Calibration - Flux (2)

MANUAL

```
fluxscaleDict = fluxscale(vis = 'uid___A002_Xac5575_X19e.ms.split',
  caltable = 'uid___A002_Xac5575_X19e.ms.split.ampli_inf',
  fluxtable = 'uid___A002_Xac5575_X19e.ms.split.flux_inf',
  reference = '0') # J1751+0939
```

- **fluxscale** - compares the input amplitude gain table and the model in setjy
  - **caltable** - the long term amplitude gains previously solved
  - **fluxtable** - the new output gain table with correct gains to scale fluxes
  - **reference** - the source used with the setjy model earlier

PIPELINE



WHY? :Setting the correct gains for flux scaling
Plots : in 'QA' directory / weblog
'*split.flux_inf.plots'
File : '*split.fluxscale'

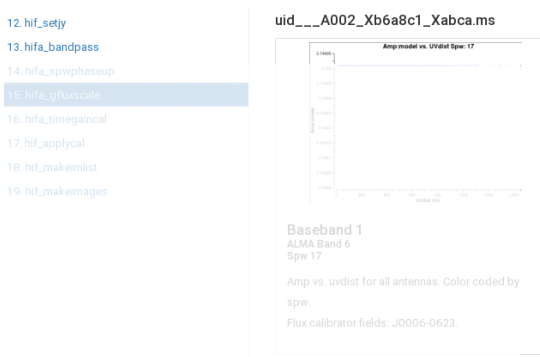
Calibration - Flux (2)

MANUAL

```
fluxscaleDict = fluxscale(vis = 'uid___A002_Xac5575_X19e.ms.split',
  caltable = 'uid___A002_Xac5575_X19e.ms.split.ampli_inf',
  fluxtable = 'uid___A002_Xac5575_X19e.ms.split.flux_inf',
  reference = '0') # J1751+0939
```

- fluxscale - compares the input amplitude gain table and the model in setjy
  - caltable - the long term amplitude gains previously solved
  - fluxtable - the new output gain table with correct gains to scale fluxes
  - reference -

PIPELINE



If one (or more) of your calibrators is weak (HF data) the flux reported could 'appear' higher some SPWs in particular narrow ones - as amplitude errors can ONLY be positive - can also map other SPW solutions

Plots : in 'QA' directory / weblog

'*split.flux_inf.plots'

File : '*split.fluxscale'

Calibration - Gains (3) - phase 'inf'

MANUAL

```
os.system('rm -rf uid___A002_Xa44acb_Xadb.ms.split.phase_inf')
gaincal(vis = 'uid___A002_Xa44acb_Xadb.ms.split',
  caltable = 'uid___A002_Xa44acb_Xadb.ms.split.phase_inf',
  field = '0,2,3', # J1733-1304,Ceres,J1832-2039
  solint = 'inf',
  refant = 'DA59',
  gaintype = 'G',
  calmode = 'p',
  gaintable = 'uid___A002_Xa44acb_Xadb.ms.split.bandpass_smooth20ch')
```

- **gaincal - creates the antenna based phase solutions**
  - **field - ALL calibrator sources**
  - **solint - 'inf' i.e. per scan/visit to the gain cal**
  - **gaintable - apply bandpass**

PIPELINE

16. hifa_timegaincal
17. hif_applycal
18. hif_makeimlist
19. hif_makeimages

16. Gain Calibration

This task creates gain solutions for each measurement

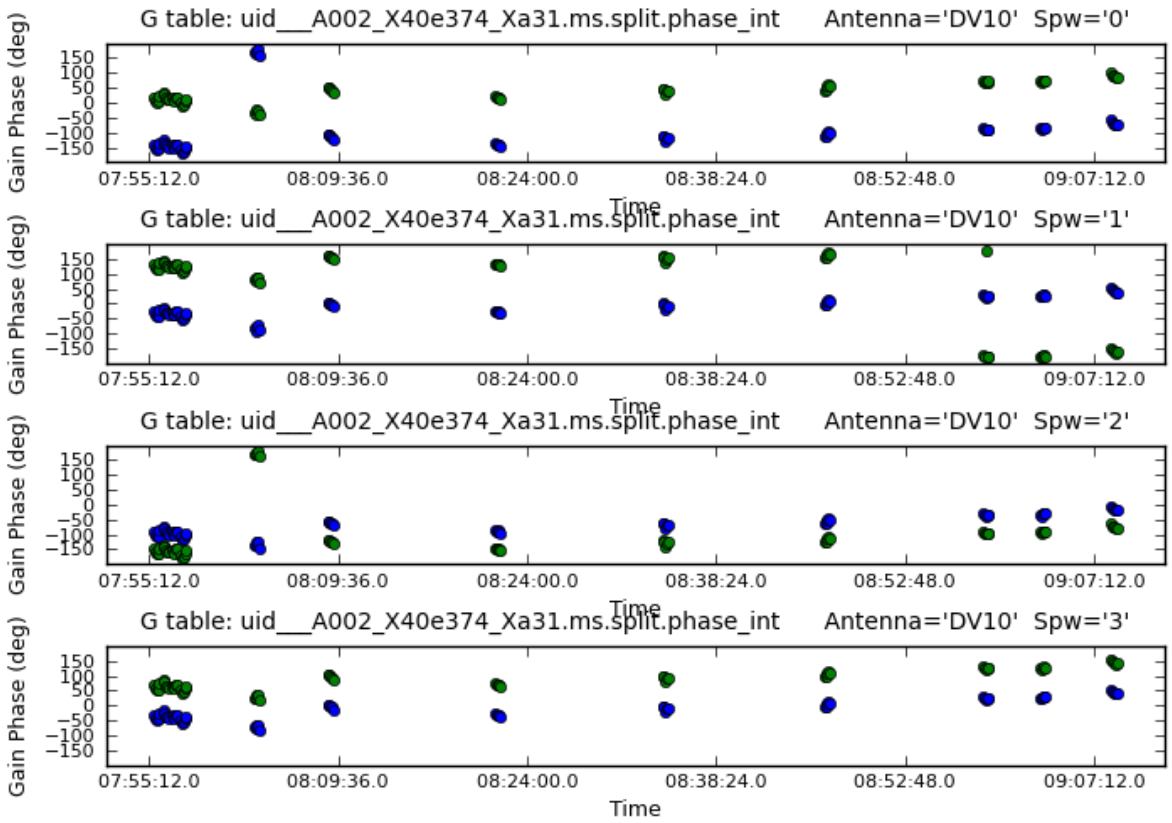
- Plots
  - Phase vs time
  - Phase structure
  - Amplitude vs time
- Diagnostic plots
  - Phase vs time
  - Amplitude vs time

Weblog

WHY? :Correct for phase variations in time (per scan) to apply to the science target
Plots : in 'QA' directory / weblog '*split.phase_inf.plots'

Calibration - Gains (3) - phase ‘inf’

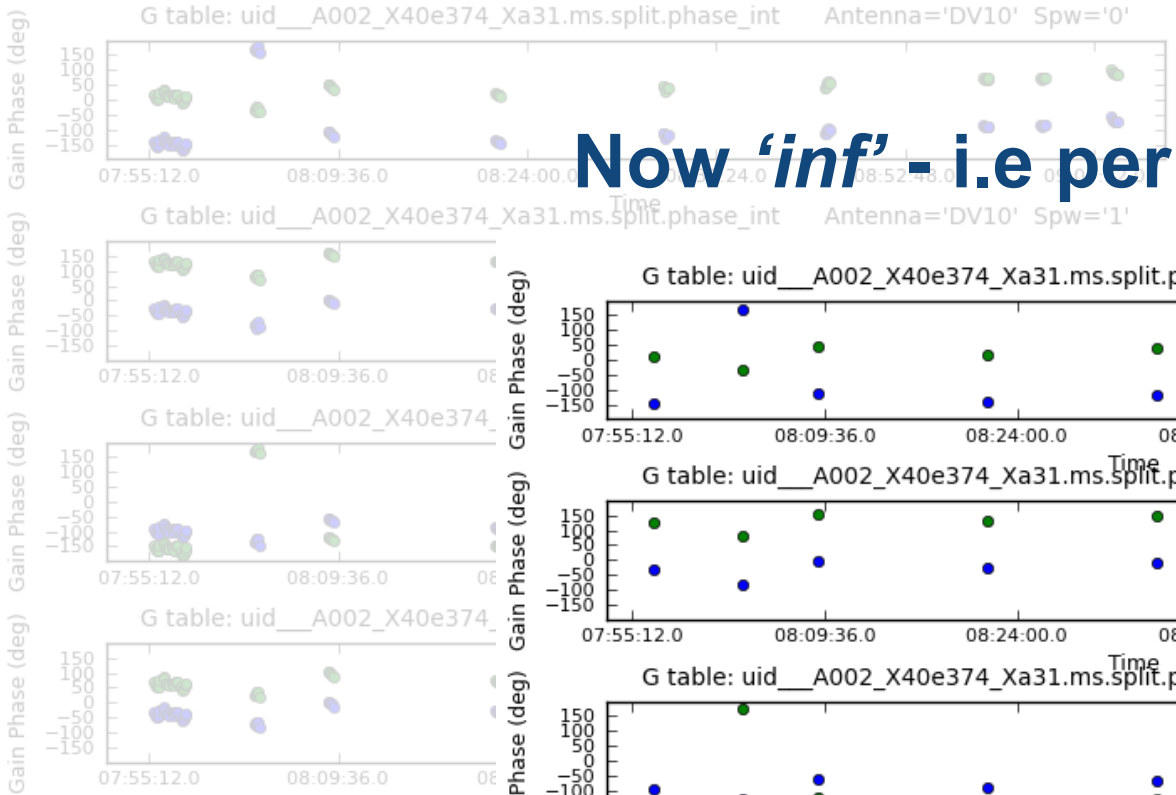
Recall - ‘int’ - integration time



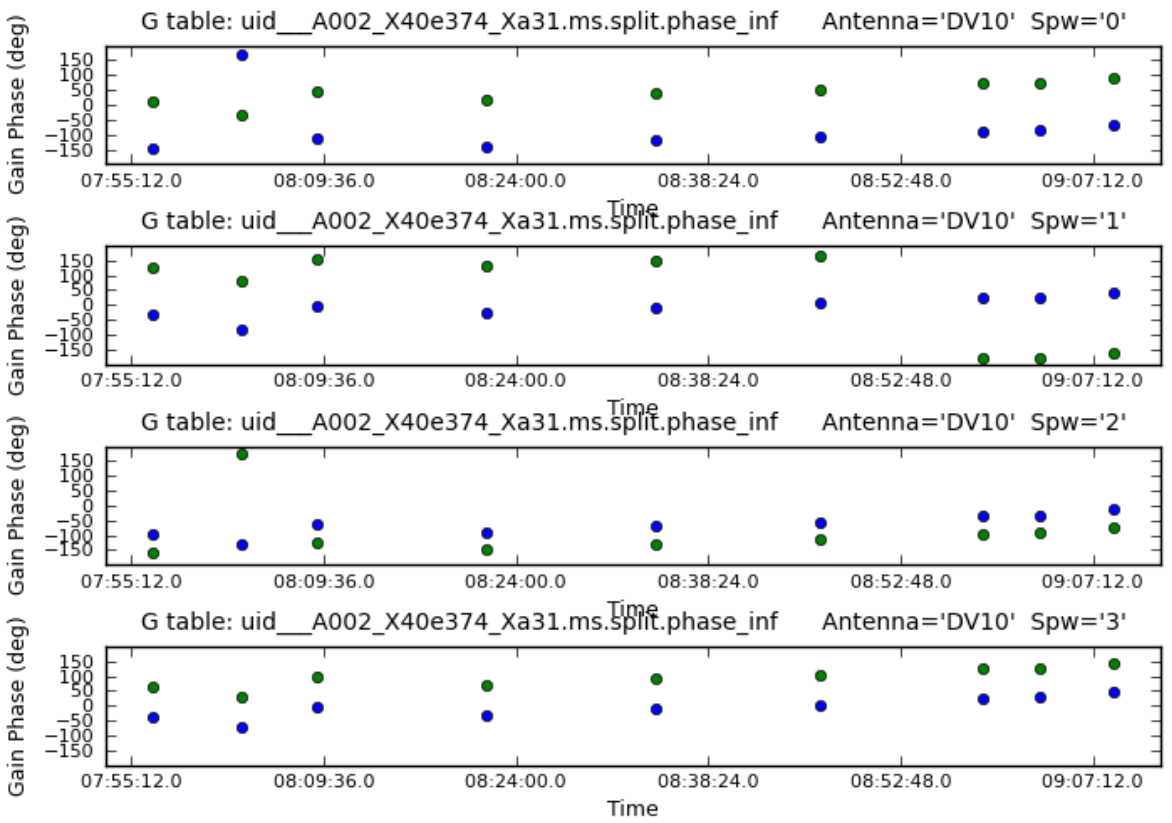
~short baseline <1 km

Calibration - Gains (3) - phase ‘inf’

Recall - ‘int’ - integration time



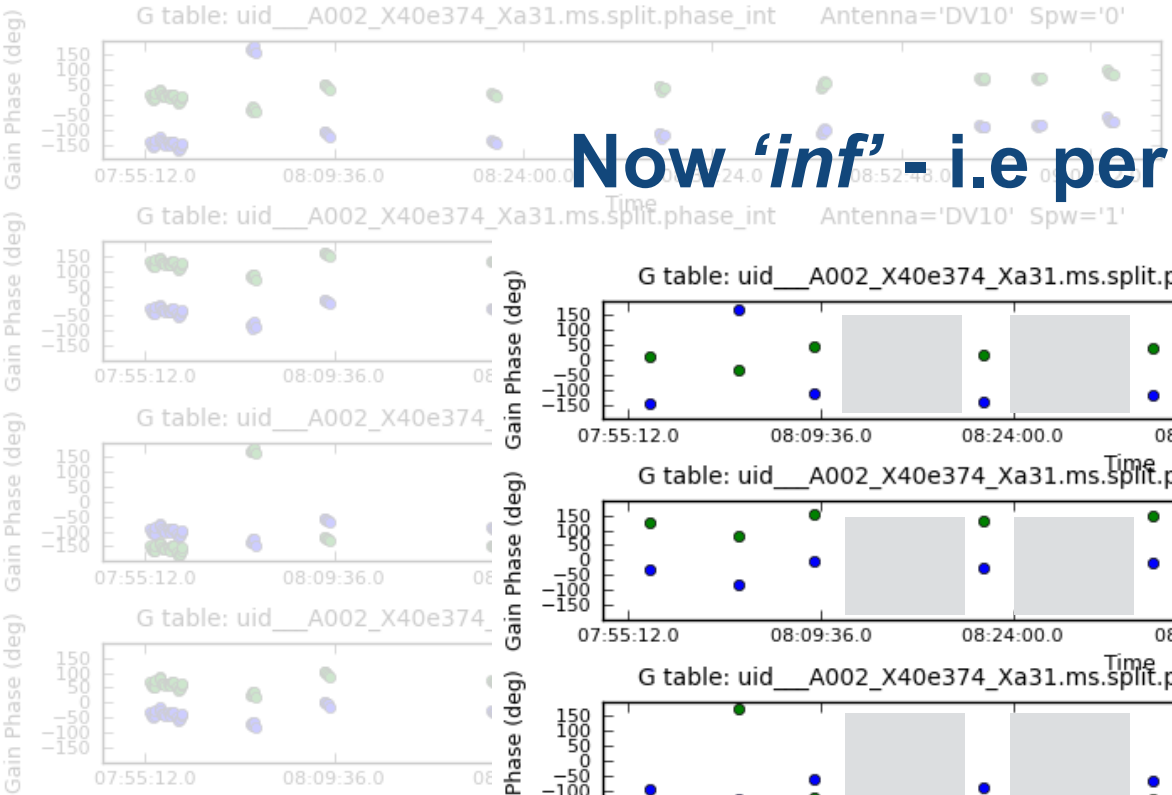
Now *‘inf’* - i.e per gain can scan



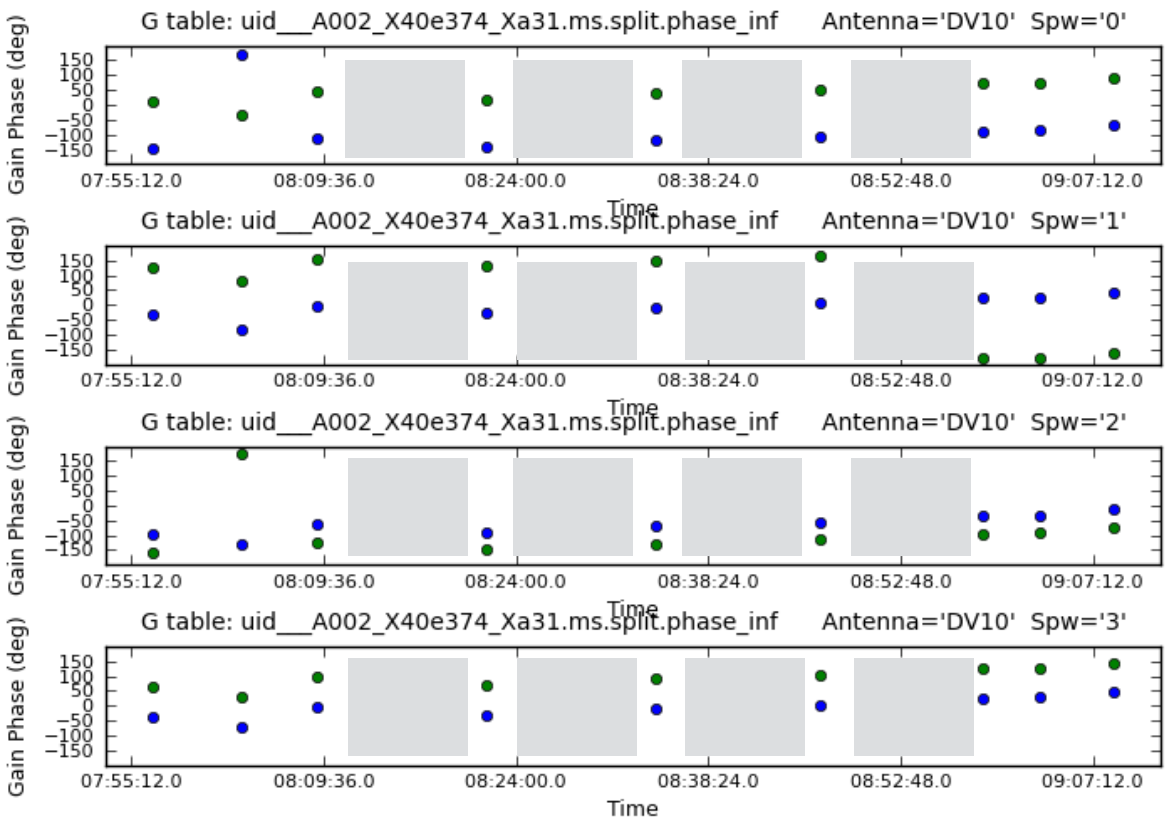
~short ba

Calibration - Gains (3) - phase 'inf'

Recall - 'int'

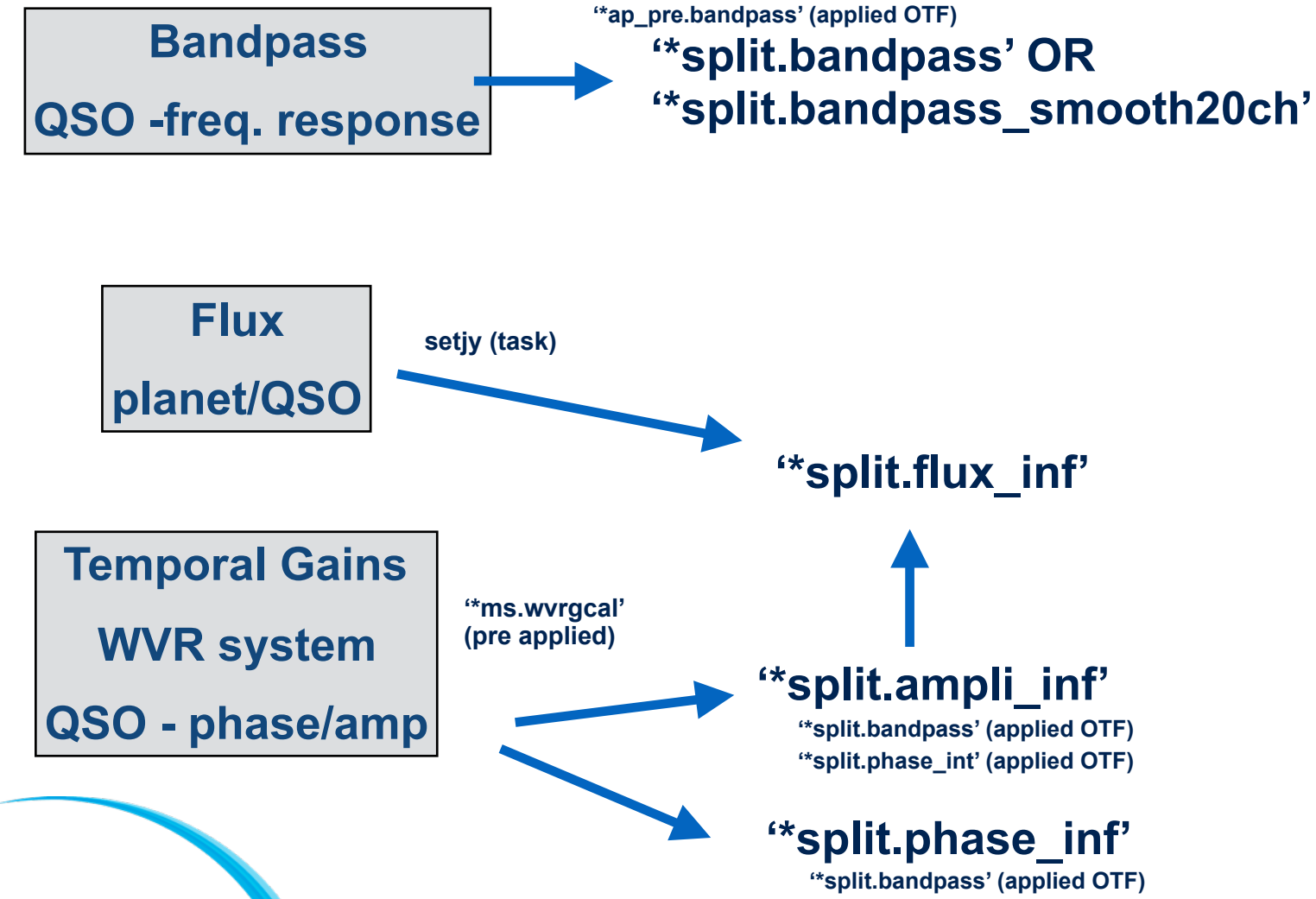


Now *'inf'* - i.e per gain can scan

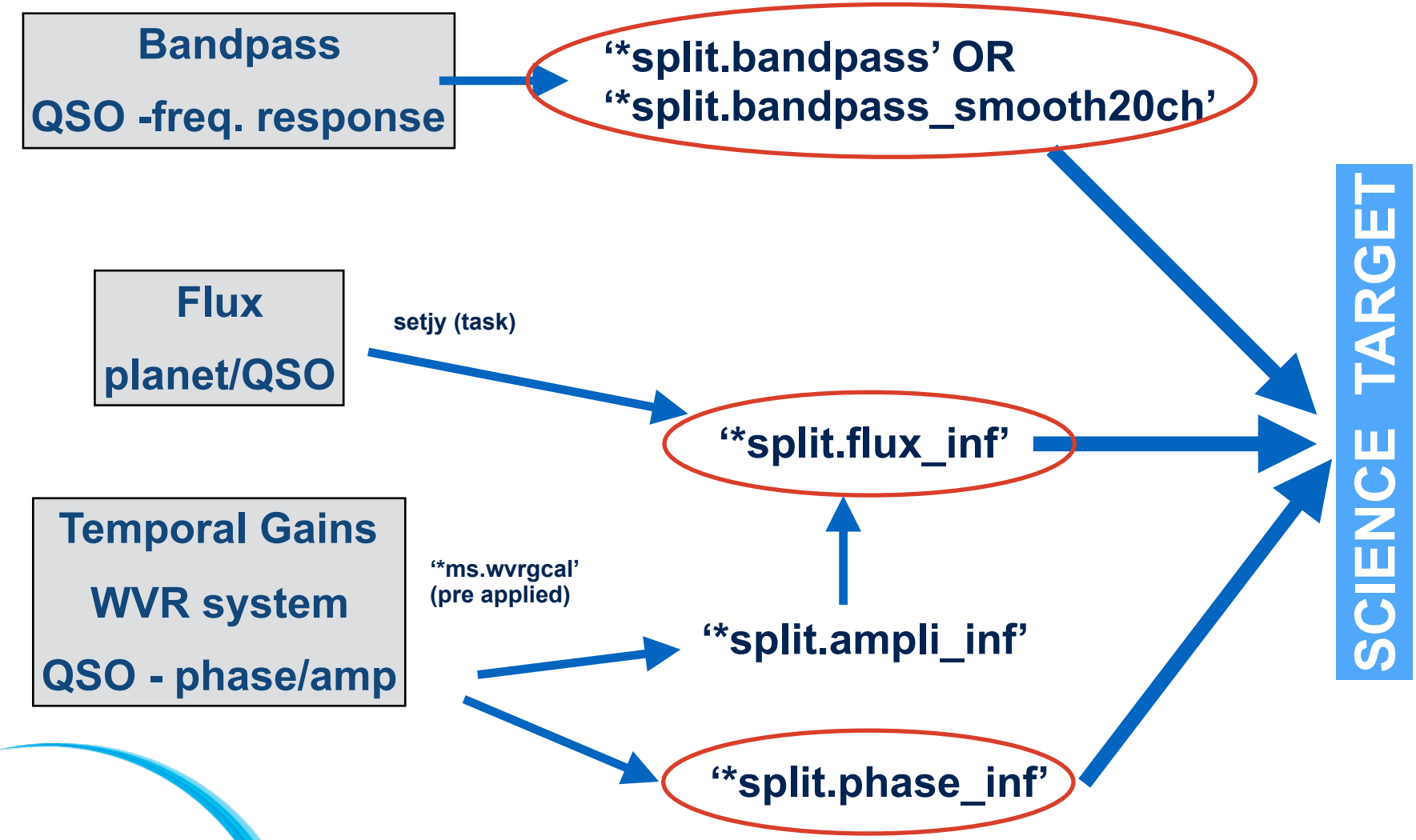


~short ba

Calibration - tables



Calibration - tables



Calibration - Application

MANUAL

```
for i in ['0', '2', '3']: # Bandpass, Ceres,J1832-2039
  applycal(vis = 'uid___A002_Xa42f75_X83e.ms.split',
    field = str(i),
    gaintable = ['uid___A002_Xa42f75_X83e.ms.split.bandpass_smooth20ch',
      'uid___A002_Xa42f75_X83e.ms.split.phase_int',
        'uid___A002_Xa42f75_X83e.ms.split.flux_inf'],
    gainfield = ['', i, i],
    interp = 'linear,linear',
    calwt = T,
    flagbackup = F)

applycal(vis = 'uid___A002_Xa42f75_X83e.ms.split',
  field = '1,4',
  gaintable = ['uid___A002_Xa42f75_X83e.ms.split.bandpass_smooth20ch',
   'uid___A002_Xa42f75_X83e.ms.split.phase_inf',
 'uid___A002_Xa42f75_X83e.ms.split.flux_inf'],
  gainfield = ['', '0', '0'], # J1733-1304
  interp = 'linear,linear',
  calwt = T,
  flagbackup = F)
```

- **applycal - applies the calibration tables**

PIPELINE

17. hif_applycal
18. hif_makeimlist
19. hif_makeimages

Contents

- Applied calibrations
- Flagged data after calibration application
- Plots
  - Calibrated amplitude vs frequency
  - Calibrated phase vs frequency
  - Calibrated amplitude vs UV distance
  - Calibrated amplitude vs time
  - Calibrated phase vs time
  - (Corrected amplitude / model) vs antenn
  - (Corrected amplitude / model) vs UV dis
  - Science target: calibrated amplitude vs
  - Science target: calibrated phase vs freq
  - Science target: calibrated amplitude vs

Weblog

WHY? :Apply all required tables
Plots : use plotms(), QA directory / weblog

Calibration - Application (1)

MANUAL

```
for i in ['0', '2', '3']: # Bandpass, Ceres,J1832-2039
  applycal(vis = 'uid___A002_Xa42f75_X83e.ms.split',
    field = str(i),
    gaintable = ['uid___A002_Xa42f75_X83e.ms.split.bandpass_smooth20ch',
      'uid___A002_Xa42f75_X83e.ms.split.phase_int',
        'uid___A002_Xa42f75_X83e.ms.split.flux_inf'],
    gainfield = ['', i, i],
    interp = 'linear,linear',
    calwt = T,
    flagbackup = F)
```

- **applycal - applies the calibration tables**
  - **gaintable - order to apply, BP, phase, flux**
  - **gainfield - which field to use in the gaintable**
  - **interp - the interpolation mode to use**

Calibration - Application (1)

MANUAL

```
for i in ['0', '2', '3']: # Bandpass, Ceres,J1832-2039
  applycal(vis = 'uid___A002_Xa42f75_X83e.ms.split',
    field = str(i),
    gaintable = ['uid___A002_Xa42f75_X83e.ms.split.bandpass_smooth20ch',
      'uid___A002_Xa42f75_X83e.ms.split.phase_int',
        'uid___A002_Xa42f75_X83e.ms.split.flux_inf'],
    gainfield = ['', i, i],
    interp = 'linear,linear',
    calwt = T,
    flagbackup = F)
```

field - these are the bandpass, flux cal, check source

ORDER !! - which field is used from which table

- **calwt** - true for correct weight for CASA >4.3.1, for correct ACA + 12m merging

- **applycal** - applies the calibration tables
  - **gaintable** - order to apply, BP, phase, flux
  - **gainfield** - which field to use in the gaintable
  - **interp** - the interpolation mode to use

For all calibrators EXCEPT phase cal, the 'int' phase solution is applied from itself

Calibration - Application (2)

MANUAL

- **applycal - apply the calibration tables - *to the science target***

```
applycal(vis = 'uid___A002_Xa42f75_X83e.ms.split',
  field = '1,4',
  gaintable = ['uid___A002_Xa42f75_X83e.ms.split.bandpass_smooth20ch',
   'uid___A002_Xa42f75_X83e.ms.split.phase_inf',
 'uid___A002_Xa42f75_X83e.ms.split.flux_inf'],
  gainfield = ['', '1', '1'], # J1733-1304
  interp = 'linear,linear',
  calwt = T,
  flagbackup = F)
```

Calibration - Application (2)

MANUAL

- **applycal - apply the calibration tables - *to the science target***

```
applycal(vis = 'uid___A002_Xa42f75_X83e.ms.split',
  field = '1,4',
  gaintable = ['uid___A002_Xa42f75_X83e.ms.split.bandpass_smooth20ch',
   'uid___A002_Xa42f75_X83e.ms.split.phase_inf',
 'uid___A002_Xa42f75_X83e.ms.split.flux_inf'],
  gainfield = ['', '1', '1'], # J1733-1304
  interp = 'linear,linear',
  calwt = T,
  flagbackup = F)
```

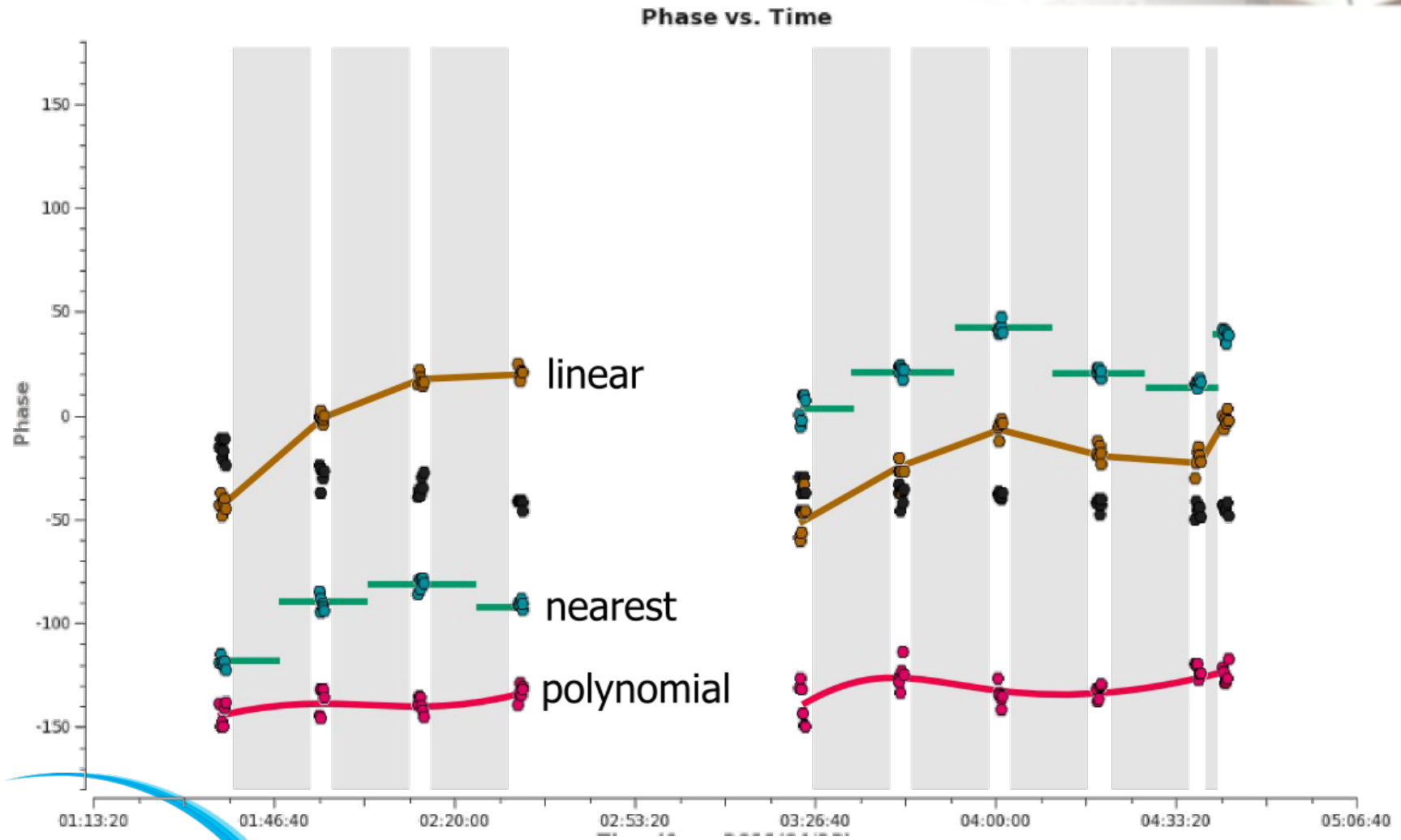
field - gain cal & science target

inf phase solution - can only interpolate phases

gainfield - solutions for field '1' are applied to BOTH '1' and '4'

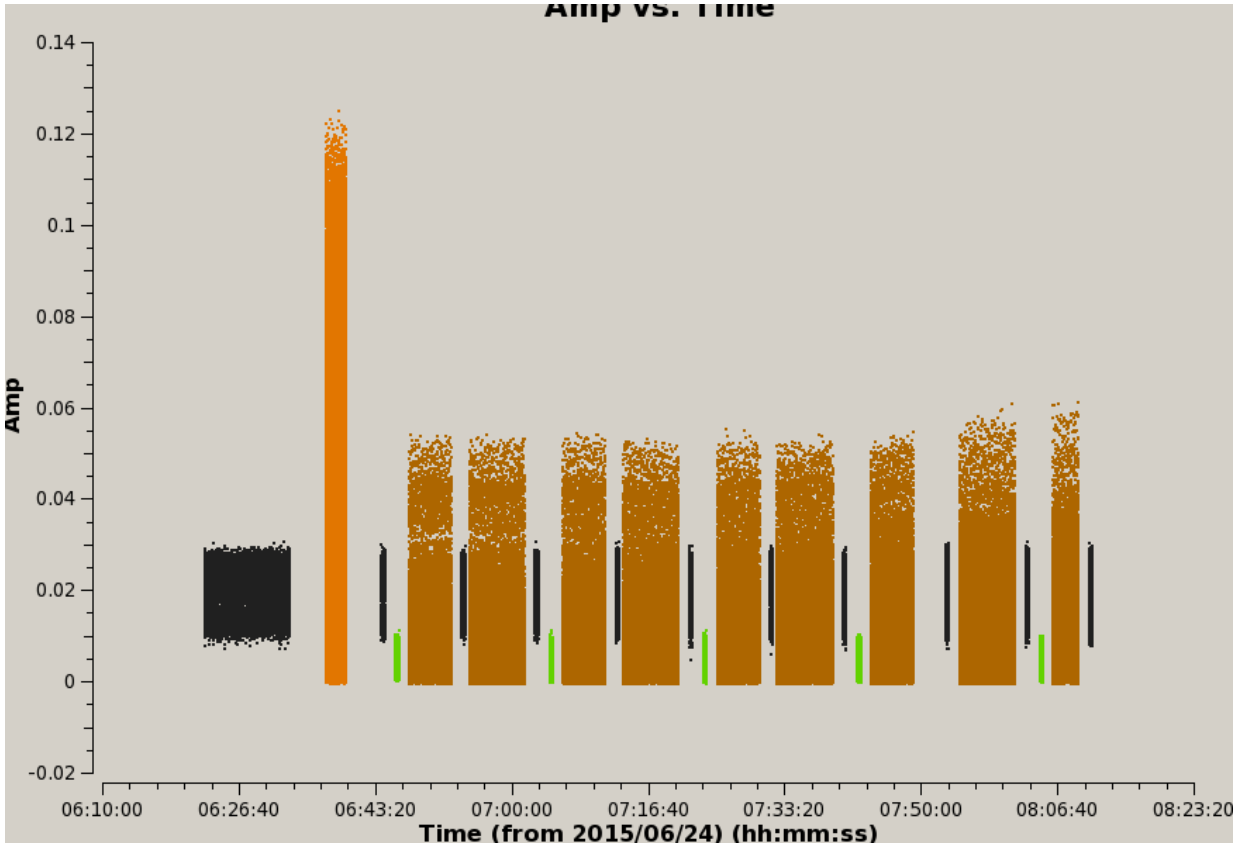
The gaincal and the science target have the solutions from the gaincal applied to them

Calibration - Application (2)



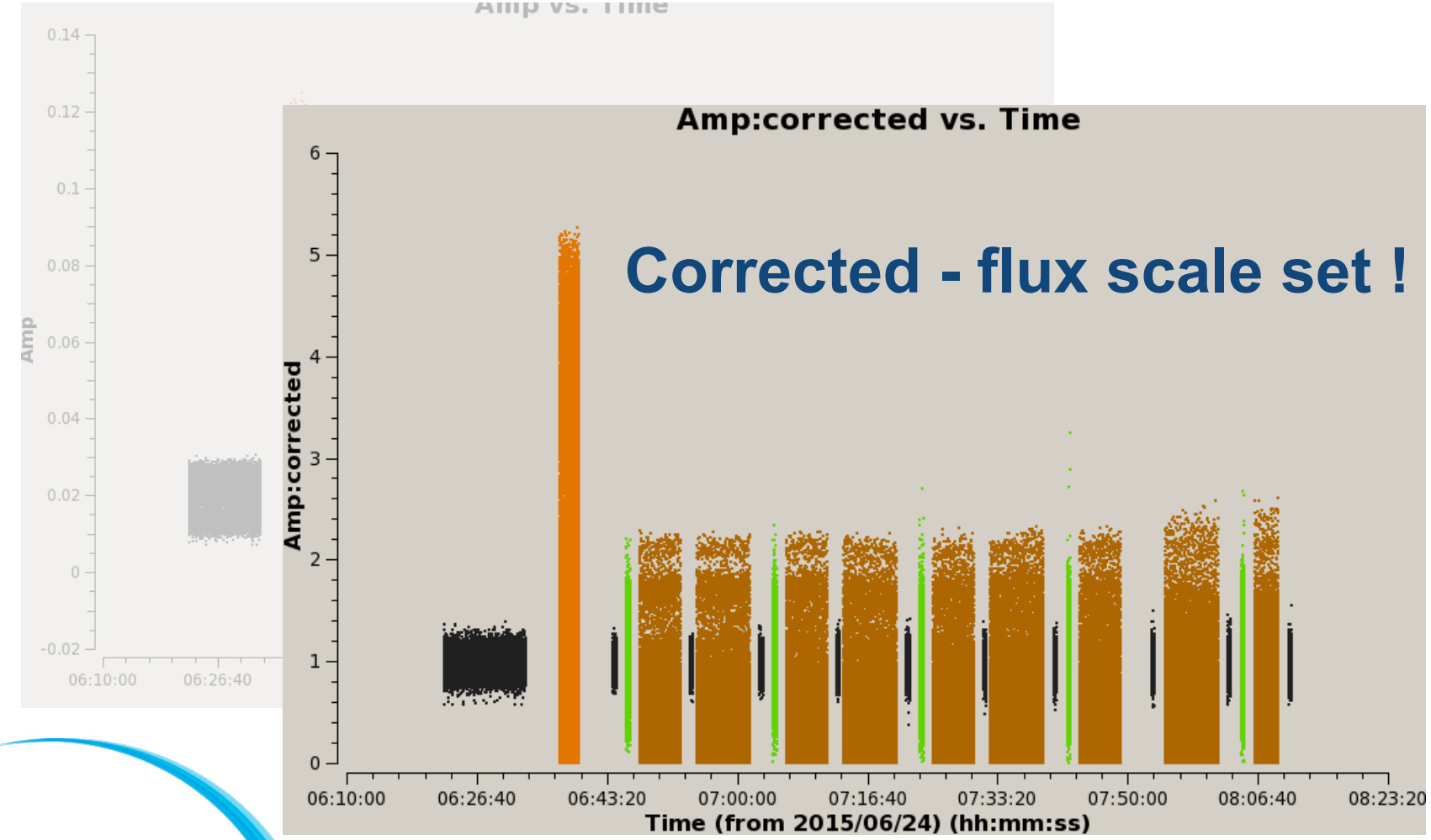
Calibration - The data (amp)

Recall - ‘raw’ data



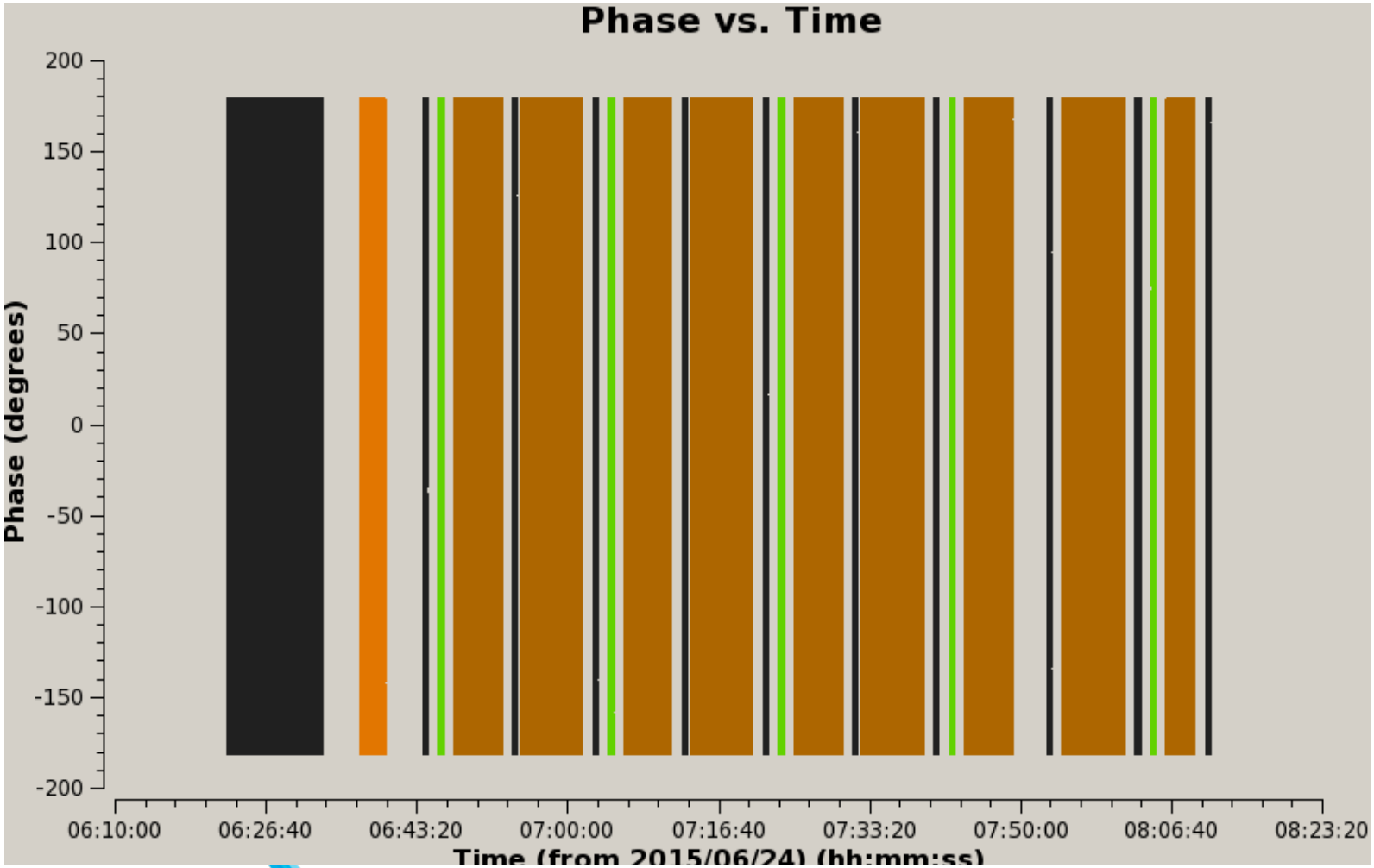
Calibration - The data (amp)

Recall - ‘raw’ data



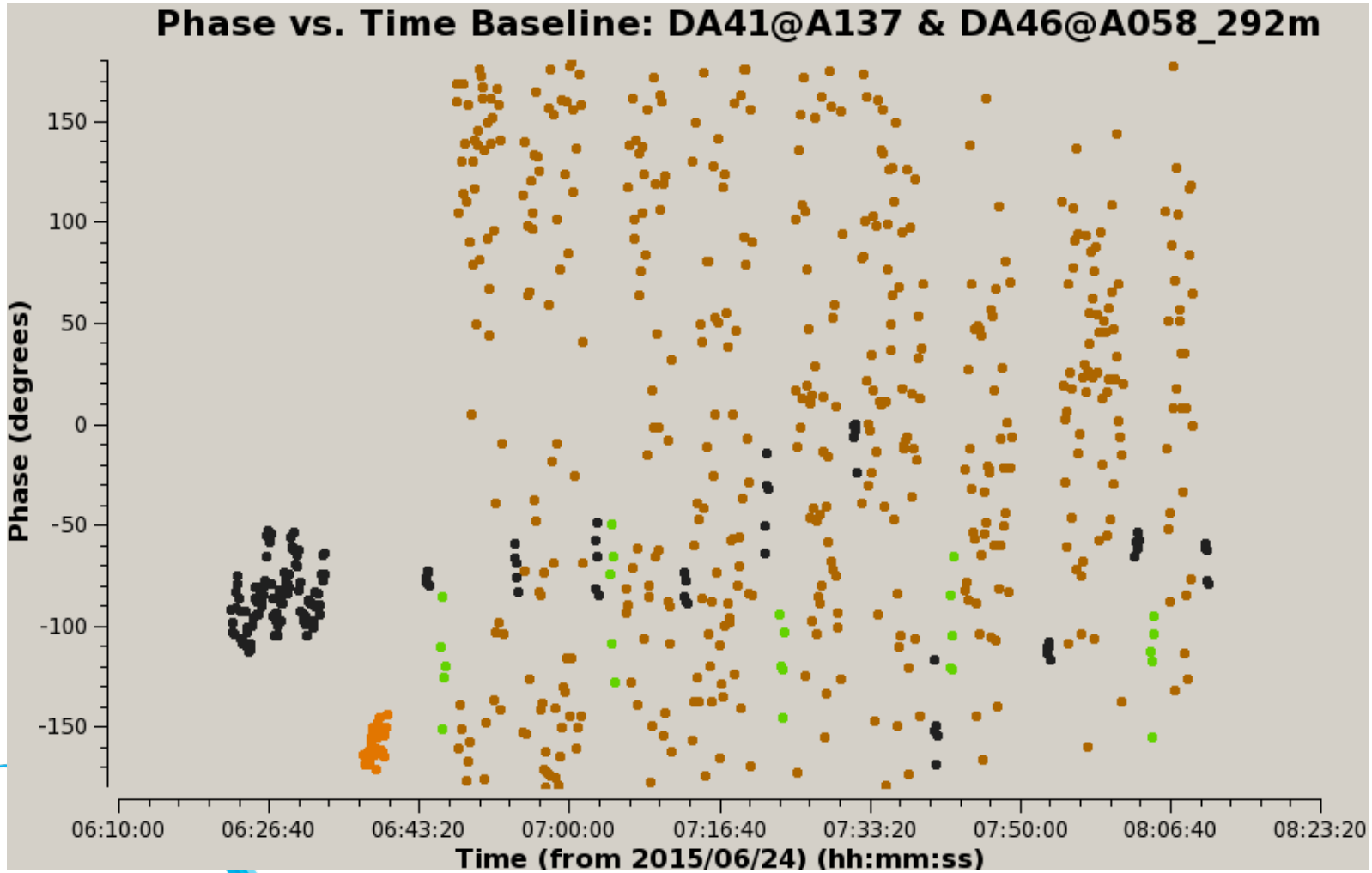
Calibration - The data (phase)

Recall - ‘raw’ data - all baselines - phases ‘everywhere’



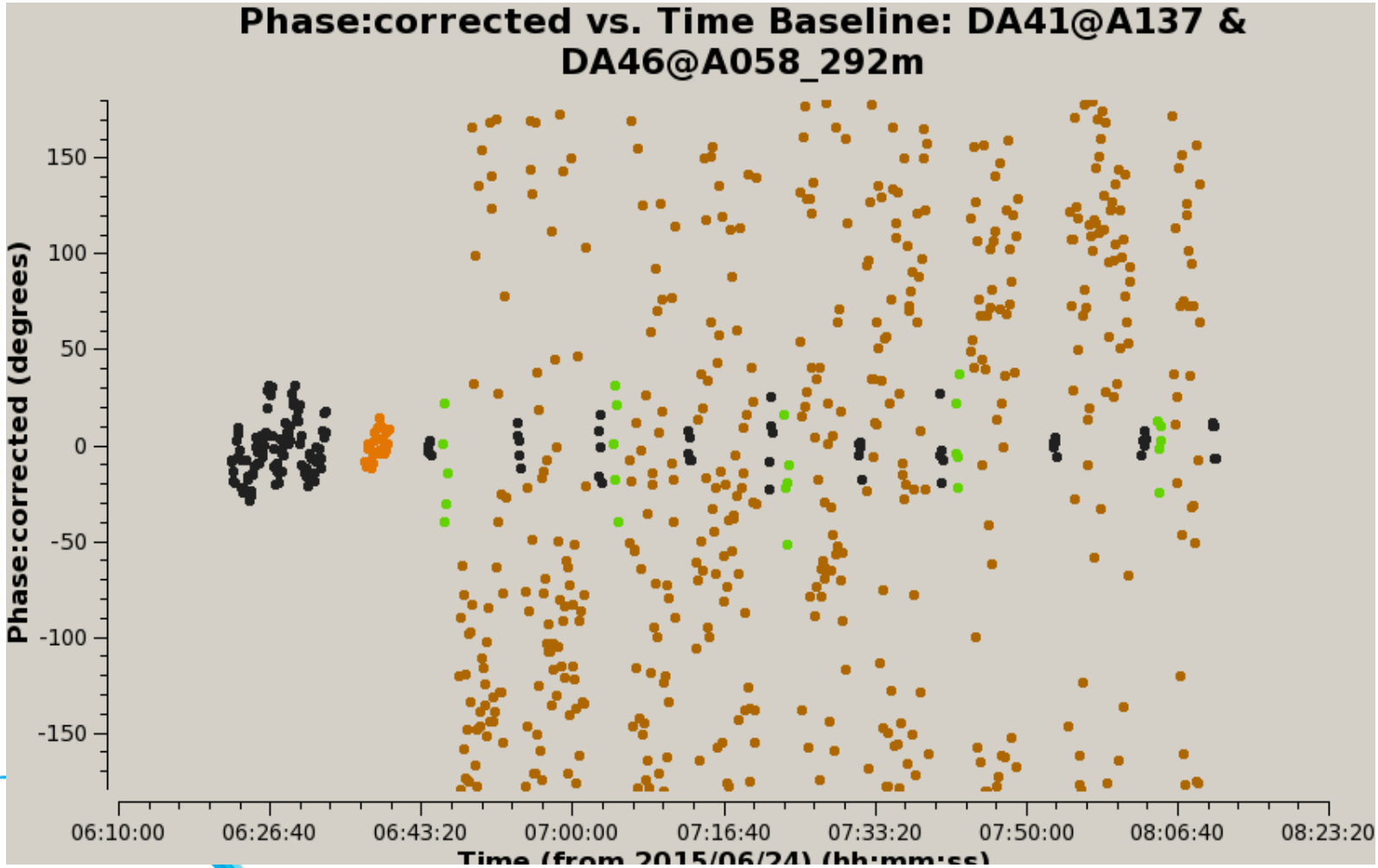
Calibration - The data (phase)

Recall - ‘raw’ data - single baseline - non-zero calibrator phases



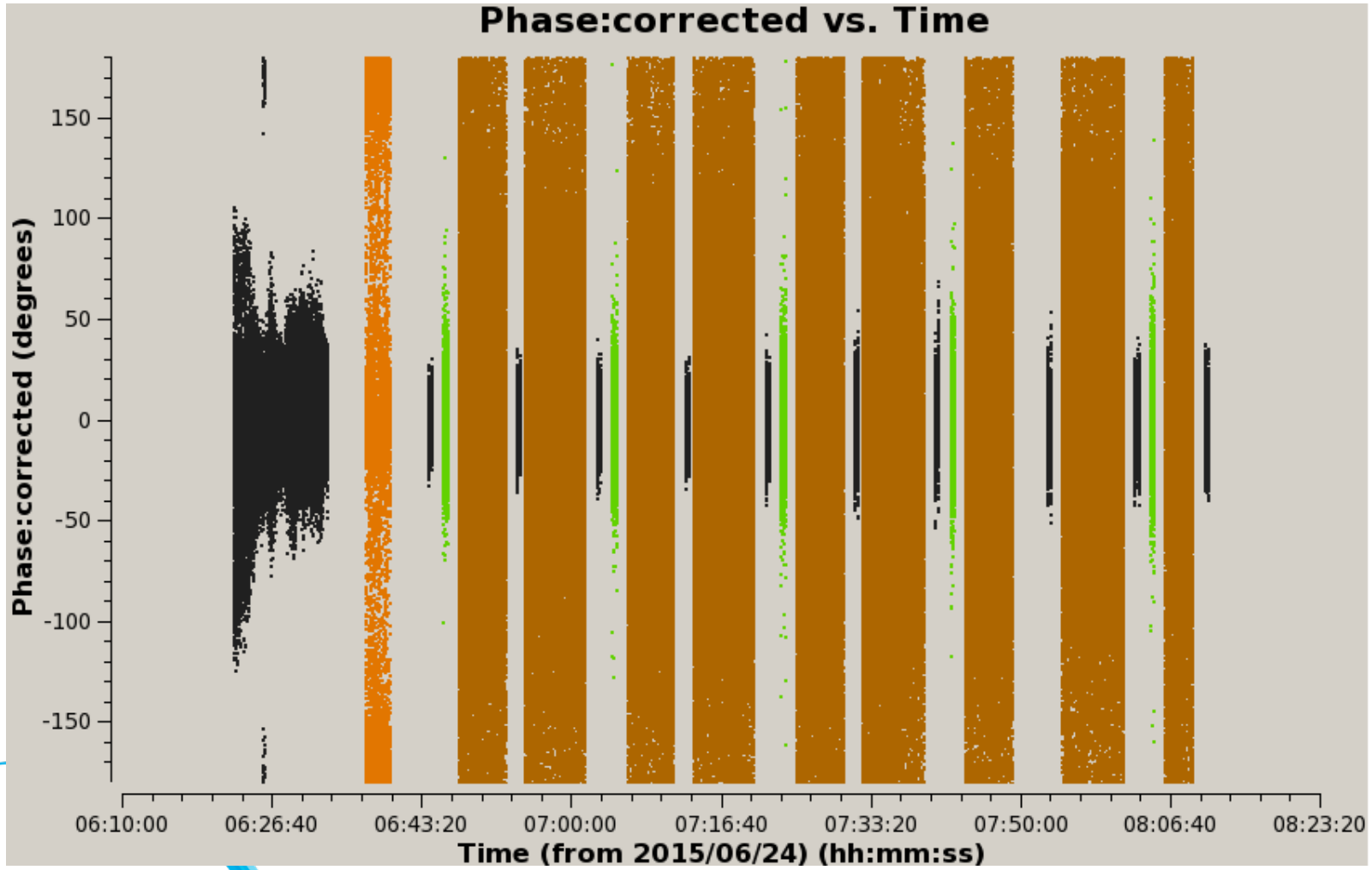
Calibration - The data (phase)

Recall - corrected data - single baseline - calibrators ~ zero phase



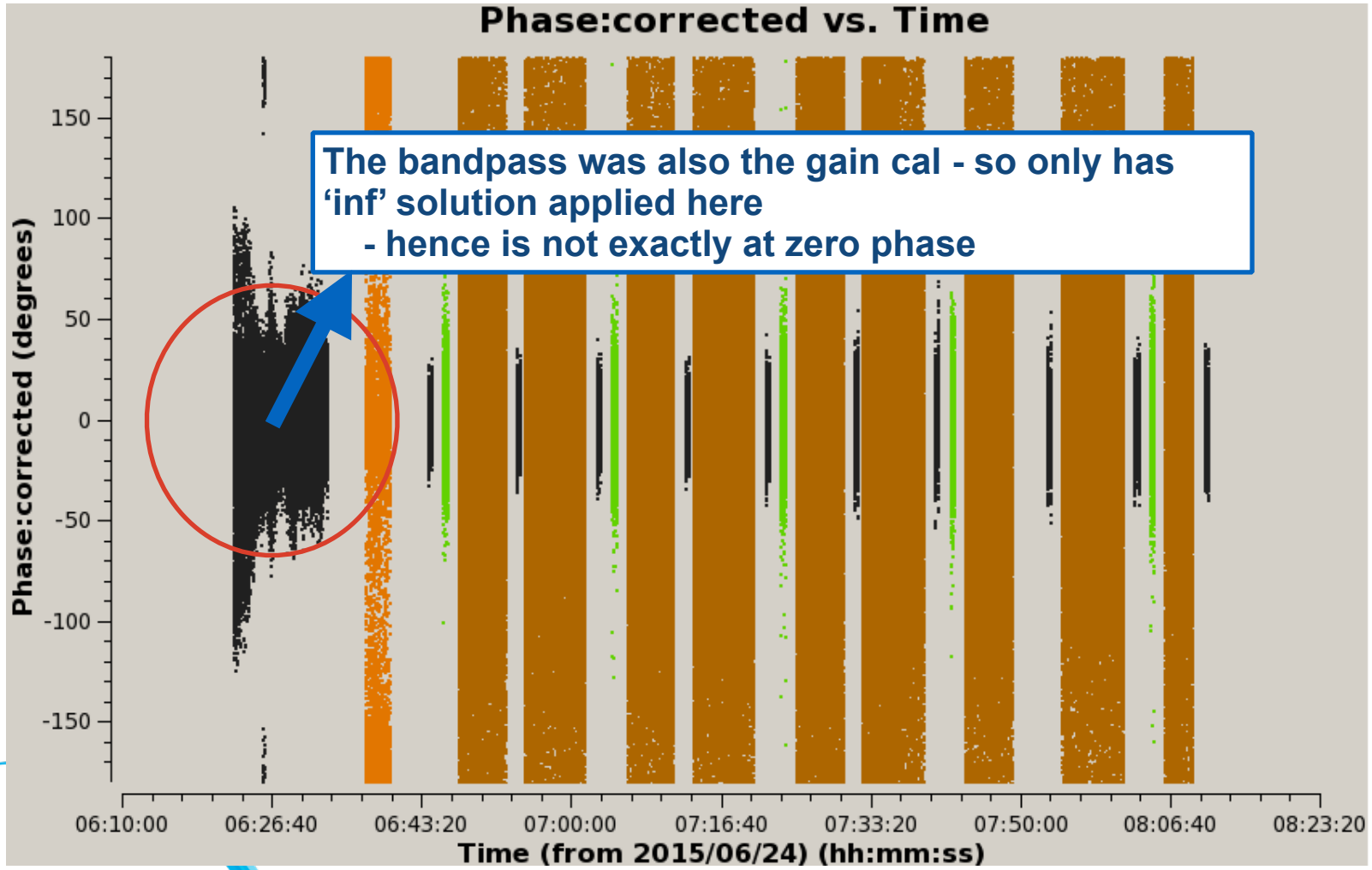
Calibration - The data (phase)

Recall - corrected data - ALL baseline - calibrators ~ zero phase



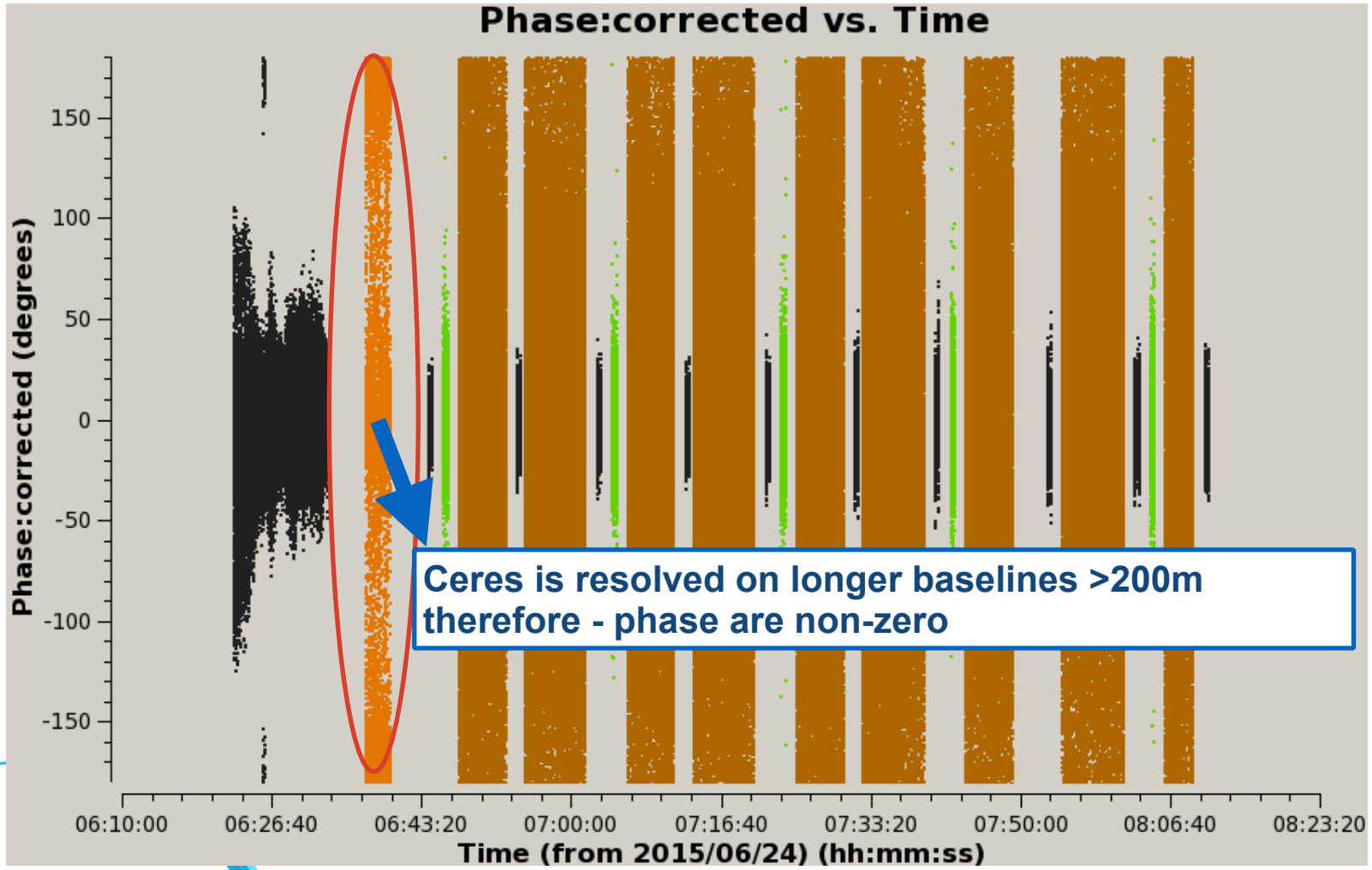
Calibration - The data (phase)

Recall - corrected data - ALL baseline - calibrators ~ zero phase



Calibration - The data (phase)

Recall - corrected data - ALL baseline - calibrators ~ zero phase



Calibration - caveats

- Solar system object as flux cal - more steps
- Narrow and wideband mixed (e.g. 2GHz and <250MHz) - SPW mapping
- Low SNR issues on gain cals - SPW combination
- Lots of narrow SPW - Bandwidth Switching

Calibration - COMPLETE!!

EXTRA SLIDES ON SSO

Calibration - Gains (1) SSO

MANUAL

```
gaincal(vis = 'uid___A002_Xa44acb_Xadb.ms.split',
  caltable = 'uid___A002_Xa44acb_Xadb.ms.split.phase_short_int',
  field = '2', # Ceres
  selectdata = T,
  antenna = 'DA46,DA49,DA53,DA55,DA59,DA63,DV04,DV08,DV18,DV19,PM02,PM04&',
  solint = 'int',
  refant = 'DA59',
  gaintype = 'G',
  calmode = 'p',
  gaintable = 'uid___A002_Xa44acb_Xadb.ms.split.bandpass')

gaincal(vis = 'uid___A002_Xa44acb_Xadb.ms.split',
  caltable = 'uid___A002_Xa44acb_Xadb.ms.split.phase_short_int',
  field = '0,3', # J1733-1304,J1832-2039
  selectdata = T,
  solint = 'int',
  refant = 'DA59',
  gaintype = 'G',
  calmode = 'p',
  append = T,
  gaintable = 'uid___A002_Xa44acb_Xadb.ms.split.bandpass')
```

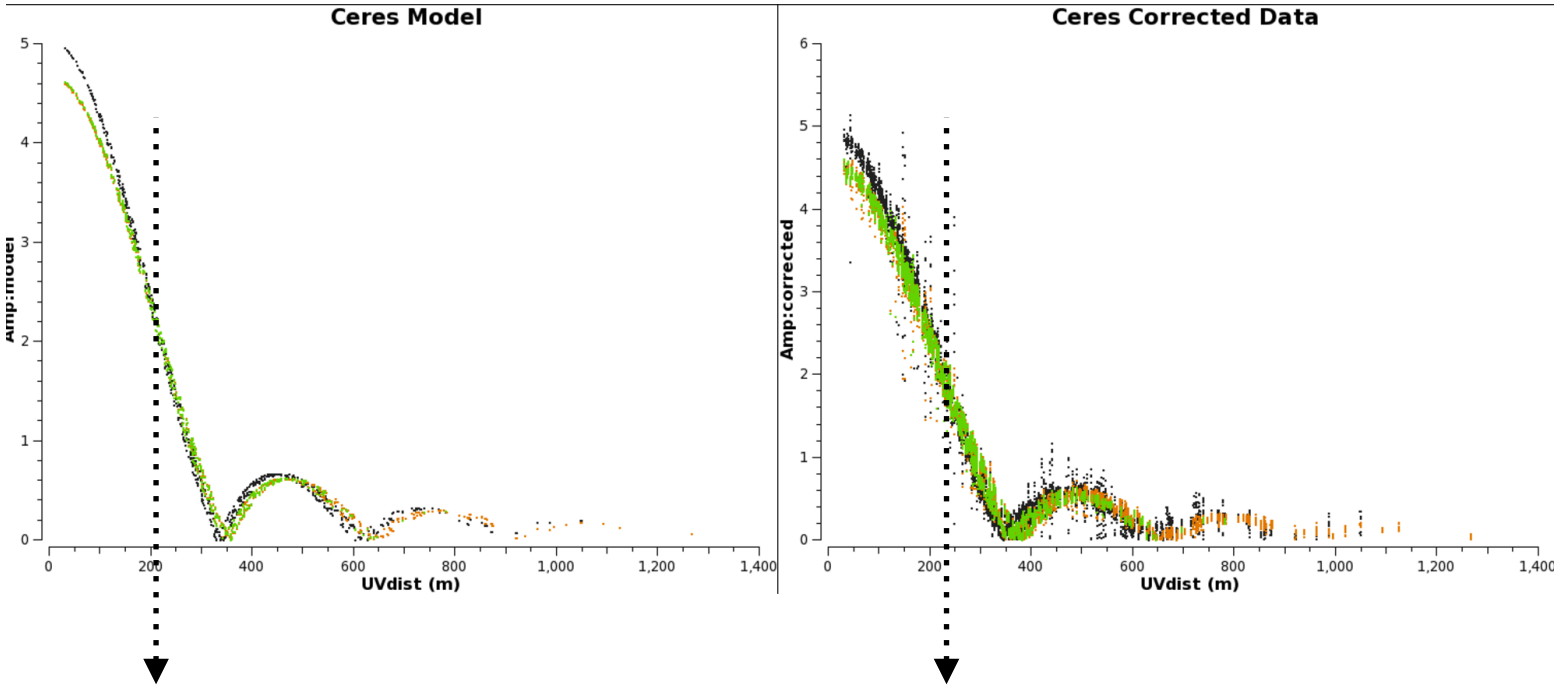
- **antenna - select the antennas close OR**
- **uvrange - select a short uvrange (e.g. <200m)**

PIPELINE

14. hifa_spwphaseup
15. hifa_gfluxscale
16. hifa_timegaincal
17. hif_applycal
18. hif_makeimlist
19. hif_makeimages

WHY? :Correct for all phase variations in time - short baselines - SSO is unresolved - 2 STEPS!!
Plots : in 'QA' directory / weblog '*split.phase_short_int.plots'

Calibration - Gains (1) SSO



ONLY baselines <200m used to bootstrap flux value to other calibrators - i.e. assume a point source such that a phase-up can be accomplished (i.e. phase_short)
Plots: QA directory

Calibration - Gains (2) SSO

MANUAL

```
os.system('rm -rf uid___A002_Xa44acb_Xadb.ms.split.ampli_short_inf')
gaincal(vis = 'uid___A002_Xa44acb_Xadb.ms.split',
  caltable = 'uid___A002_Xa44acb_Xadb.ms.split.ampli_short_inf',
  field = '0,2,3', # J1733-1304,Ceres,J1832-2039
  selectdata = T,
  solint = 'inf',
  refant = 'DA59',
  gaintype = 'T',
  calmode = 'a',
  gaintable = ['uid___A002_Xa44acb_Xadb.ms.split.bandpass_smooth20ch',
'uid___A002_Xa44acb_Xadb.ms.split.phase_short_int'])
```

- **gaincal - creates the antenna based amp solutions**
  - **field - ALL calibrator sources**
  - **solint - 'inf' i.e. per scan/visit to the gain cal**
  - **gaintable - apply bandpass AND the 'int' SHORT phase-up solutions OTF**

PIPELINE

15. hifa_gfluxscale
16. hifa_timegaincal
17. hif_applycal
18. hif_makeimlist
19. hif_makeimages

WHY? :Correct for slow amplitude variations in time
Plots : In some cases in 'QA' directory / weblog '*split.ampli_short_int.plots'

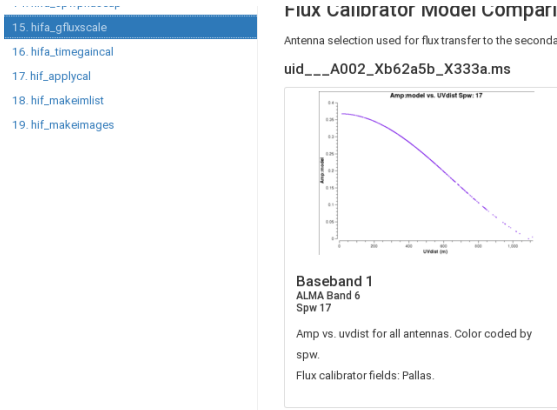
Calibration - Flux (2) SSO

MANUAL

```
fluxscaleDict = fluxscale(vis = 'uid___A002_Xa44acb_Xadb.ms.split',
  caltable = 'uid___A002_Xa44acb_Xadb.ms.split.ampli_short_inf',
  fluxtable = 'uid___A002_Xa44acb_Xadb.ms.split.flux_short_inf',
  reference = '2') # Ceres
```

```
f = open('uid___A002_Xa44acb_Xadb.ms.split.fluxscale')
fc = f.readlines()
f.close()

for phaseCalName in ['J1733-1304']:
  for i in range(len(fc)):
    if fc[i].find('Flux density for '+phaseCalName) != -1 and re.search('in Sp
*?\))? is: [0-9]+\.[0-9]+', fc[i], re.DOTALL|re.IGNORECASE) != None:
      line = (re.search('in SpW=[0-9]+(?: \(.*?\))? is: [0-9]+\.[0-9]+', fc[i],
IGNORECASE)).group(0)
      spwId = (line.split('='))[1].split()[0]
      flux = float((line.split(': '))[1].split()[0])
      setjy(vis = 'uid___A002_Xa44acb_Xadb.ms.split',
        field = phaseCalName.replace(';','*;').split(';')[0],
        spw = spwId,
        standard = 'manual',
        fluxdensity = [flux,0,0,0])
```

PIPELINE



WHY? :Setting the correct gains for flux scaling - then recalibrate all baselines with bootstrapped flux
Plots : in 'QA' directory / weblog - '*split.flux_inf.plots'

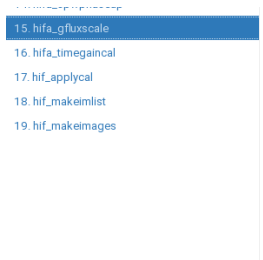
Calibration - Flux (2) SSO

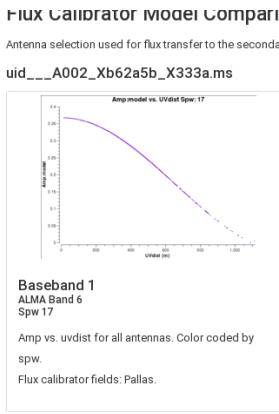
MANUAL

```
fluxscaleDict = fluxscale(vis = 'uid___A002_Xa44acb_Xadb.ms.split',
  caltable = 'uid___A002_Xa44acb_Xadb.ms.split.ampli_short_inf',
  fluxtable = 'uid___A002_Xa44acb_Xadb.ms.split.flux_short_inf',
  reference = '2') # Ceres
```

```
f = open('uid___A002_Xa44acb_Xadb.ms.split.fluxscale')
fc = f.readlines()
f.close()

for phaseCalName in ['J1733-1304']:
  for i in range(len(fc)):
    if fc[i].find('Flux density for '+phaseCalName) != -1 and re.search('in Sp\
*?\))? is: [0-9]+\.[0-9]+', fc[i], re.DOTALL|re.IGNORECASE) != None:
      line = (re.search('in SpW=[0-9]+(?: \(.*?\))? is: [0-9]+\.[0-9]+', fc[i],
IGNORECASE)).group(0)
      spwId = (line.split('='))[1].split()[0]
      flux = float((line.split(':'))[1].split()[0])
      setjy(vis = 'uid___A002_Xa44acb_Xadb.ms.split',
        field = phaseCalName.replace(';','*;').split(';')[0],
        spw = spwId,
        standard = 'manual',
        fluxdensity = [flux,0,0,0])
```

PIPELINE





- **loop - searches the *fluxscale file for the flux of another calibrator - gain cal**

WHY? :Setting the correct gains for flux scaling - then recalibrate all baselines with bootstrapped flux
Plots : in 'QA' directory / weblog - '*split.flux_inf.plots'

Calibration - Flux (2) SSO

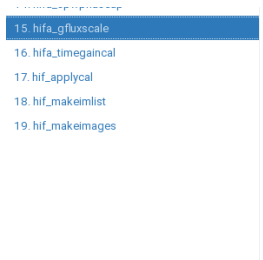
MANUAL

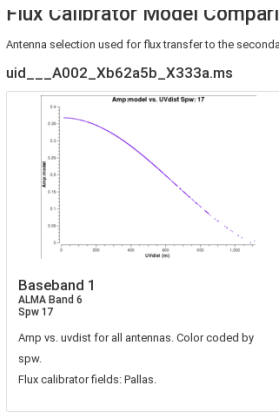
```
fluxscaleDict = fluxscale(vis = 'uid___A002_Xa44acb_Xadb.ms.split',
  caltable = 'uid___A002_Xa44acb_Xadb.ms.split.ampli_short_inf',
  fluxtable = 'uid___A002_Xa44acb_Xadb.ms.split.flux_short_inf',
  reference = '2') # Ceres
```

```
f = open('uid___A002_Xa44acb_Xadb.ms.split.fluxscale')
fc = f.readlines()
f.close()

for phaseCalName in ['J1733-1304']:
  for i in range(len(fc)):
    if fc[i].find('Flux density for '+phaseCalName) != -1 and re.search('in Sp\
*?\))? is: [0-9]+\.[0-9]+', fc[i], re.DOTALL|re.IGNORECASE) != None:
      line = (re.search('in SpW=[0-9]+(?: \(.*?\))? is: [0-9]+\.[0-9]+', fc[i],
IGNORECASE)).group(0)
      spwId = (line.split('='))[1].split()[0]
      flux = float((line.split(':'))[1].split()[0])
      setjy(vis = 'uid___A002_Xa44acb_Xadb.ms.split',
        field = phaseCalName.replace(';','*;').split(';')[0],
        spw = spwId,
        standard = 'manual',
        fluxdensity = [flux,0,0,0])
```

- **setjy** - called again to set gain cal bootstrapped flux

PIPELINE





- **loop** - searches the *fluxscale file for the flux of another calibrator - gain cal

WHY? :Setting the correct gains for flux scaling - then recalibrate all baselines with bootstrapped flux
Plots : in 'QA' directory / weblog - '*split.flux_inf.plots'

Calibration - Redo-Gains/Flux - SSO

MANUAL

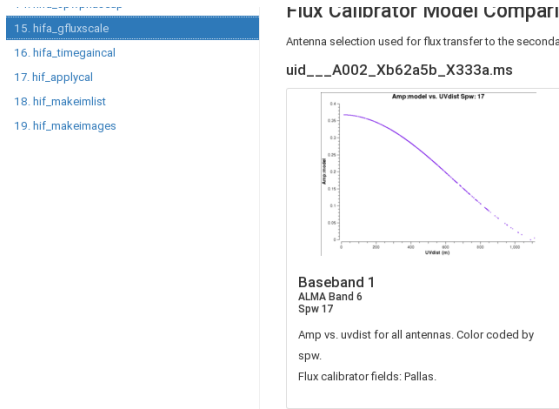
```
gaincal(vis = 'uid___A002_Xa44acb_Xadb.ms.split',
  caltable = 'uid___A002_Xa44acb_Xadb.ms.split.phase_int',
  field = '0,2,3', # J1733-1304,Ceres,J1832-2039
  solint = 'int',
  refant = 'DA59',
  gaintype = 'G',
  calmode = 'p',
  gaintable = 'uid___A002_Xa44acb_Xadb.ms.split.bandpass_smooth20ch')
```

- **gaincal - creates the antenna based phase solutions - ALL baselines**

```
gaincal(vis = 'uid___A002_Xa44acb_Xadb.ms.split',
  caltable = 'uid___A002_Xa44acb_Xadb.ms.split.flux_inf',
  field = '0,2,3', # J1733-1304,Ceres,J1832-2039
  solint = 'inf',
  refant = 'DA59',
  gaintype = 'T',
  calmode = 'a',
  gaintable = ['uid___A002_Xa44acb_Xadb.ms.split.bandpass_smooth20ch',
'uid___A002_Xa44acb_Xadb.ms.split.phase_int'])
```

- **gaincal - creates the antenna based amp solutions - ALL baselines**

PIPELINE



WHY? : Resolve phase-up, then re-solve amplitudes for ALL baselines after flux bootstrap from resolved SSO