T8: Very Long Baseline Interferometry

The dataset we are using for this tutorial is from the EVN experiment N14C3. This is a 6-cm network monitoring experiment.

The EVN data were obtained from the EVN Archive. As this is a "test" experiment, these data have no proprietary period. JIVE correlates all EVN observations and performs a preliminary reduction using the EVN pipeline. In order to decrease the size of the initial dataset, these data have been frequency averaged and some calibration tables from the pipeline have already been copied over *but not applied*.

We will be using AIPS to reduce these data as, to-date, other packages do not have the ability to fringe fit data, which is integral for non-connected element arrays.

1. Obtain data

Download the data from the tutorial webpage or here. The dataset is 1.1GB.

I recommend setting up the environment variable `MYDIR` to avoid having to type long paths. In the directory where your data are type

(for tcsh)
`setenv MYDIR `pwd``

(for bash)
`export MYDIR=`pwd``

2. Start AIPS

Start AIPS by simply running "aips" in the command line. If you wish, you can specify tv=local.

`aips tv=local`

AIPS will ask you to choose a printer and if you set-up correctly the default is usually ok. At this point 3 windows will pop-up (well 2 will pop up and 1 will be auto-minimised).

- AIPS_MSGSRV_1: The message server window lets you know what AIPS is doing. If you close this window, messages will show up in your terminal. I prefer to keep the message server on.
- AIPSTV: Plots and images are shown in the TV.
- AIPS_TEKSRV_1: Used for programs using the Tek graphics display, which is hardly used anymore.

3. AIPS User ID

Back in the terminal that you are running AIPS in, put in your AIPS user ID. For this tutorial I am choosing **userid 3**. You should of course choose whatever user id you want! Userid 1 is reserved for the local AIPS manager and is password protected, and userid 2 is likely where you tested your AIPS installation ☺.

You're now ready to load in the data!

4. Load in the data

In the AIPS terminal type the following:

```
free
```

Select a disk number with sufficient free space; this will be the `outdisk` and here number `1` will be used. Your case will look a bit different (you may have only one disk).

```
>free
AIPS 2: Disk Volume name              Total Full      Free Timd Access
AIPS 2:  #                           Mbytes    %    Mbytes days
AIPS 2:  1  <ome/euro2/AIPS/DA01    1877663   80    380514   14 Alluser
AIPS 2:  2  <ome/euro3/AIPS/DA01    1877663   88    236902   14 Alluser
```

The following typed in AIPS will load the data to "disk 1":

`default fitld`	Call the task `fitld` with all parameters set to default values.
`datain 'MYDIR:T8.fits`	Note the lack of a closing quote.
`digicor -1`	This should be off for the EVN correlator.
`outname 'data'`	What's in a name? ❀
`outdisk 1`	Outdisk 1 is default but change this if you are working on a different disk.
`inp`	Always check your inputs again before you 'go'!
`go`	

The message server will notify you that your data are being loaded. This may take a minute. The data will have finished loading when the message server says that FITLD "Appears to have ended successfully."

AIPS Tip: Make sure you do not have a quote at the end of the file name, or else the lower case characters get changed into upper case and the file won't be found!

AIPS Tip: AIPS supports both minimum match and tab-completion.

5. Inspect the data

We can see what data are loaded into AIPS by typing

```
pcat
```

This task prints all catalogued files in all available disks. Use `getn n` to get the nth catalogue file.

```
indi 1; getn 1; imhead
```

AIPS Tip: You can use a semicolon (;) to separate input parameters ...

```
>indi 1; getn 1; imhead
AIPS 1: Got(1)   disk= 1  user=   3   type=UV   DATA.SPLAT.1
AIPS 1: Image=MULTI     (UV)          Filename=DATA        .SPLAT .    1
AIPS 1: Telescope=EVN                 Receiver=VLBA
AIPS 1: Observer=N14C3                User #=     3
AIPS 1: Observ. date=22-OCT-2014      Map date=30-AUG-2015
AIPS 1: # visibilities     185887     Sort order  TB
AIPS 1: Rand axes: UU-L-SIN  VV-L-SIN  WW-L-SIN  TIME1  SUBARRAY
AIPS 1:            SOURCE  INTTIM  CORR-ID  ANTENNA1  ANTENNA2
AIPS 1: ----------------------------------------------------------------
AIPS 1: Type    Pixels   Coord value     at Pixel     Coord incr   Rotat
AIPS 1: COMPLEX      3   0.0000000E+00       1.00  1.0000000E+00    0.00
AIPS 1: STOKES       4  -1.0000000E+00       1.00 -1.0000000E+00    0.00
```

```
AIPS 1: FREQ        16   4.9272400E+09       1.00  1.0000000E+06    0.00
AIPS 1: IF           8   1.0000000E+00       1.00  1.0000000E+00    0.00
AIPS 1: RA           1    16 40 29.633       1.00       3600.000    0.00
AIPS 1: DEC          1    39 46 46.028       1.00       3600.000    0.00
AIPS 1: ----------------------------------------------------------------
AIPS 1: Coordinate equinox 2000.00
AIPS 1: Rest freq      0.000         Vel type: OPTICAL wrt YOU
AIPS 1: Alt ref. value  0.00000E+00  wrt pixel    0.00
AIPS 1: Maximum version number of extension files of type HI is   1
AIPS 1: Maximum version number of extension files of type NX is   1
AIPS 1: Maximum version number of extension files of type FQ is   1
AIPS 1: Maximum version number of extension files of type CT is   1
AIPS 1: Maximum version number of extension files of type AN is   1
AIPS 1: Maximum version number of extension files of type CL is   2
AIPS 1: Maximum version number of extension files of type FG is   1
AIPS 1: Maximum version number of extension files of type SU is   1
AIPS 1: Keyword = 'OLDRFQ  '  value =  4.92699000D+09
```

The header contains a lot of useful information such as the number of IFs (sub-bands), the number of frequency channels per IF, and the number of Stokes parameters. The header also shows what extension files are already associated with the data.

You will see that there are two CL (calibration) tables associated with the data. In this case, CL1 is an "pristine" calibration table created when loading in the data, and CL2 contains a priori amplitude calibration as well as parallactic angle correction. For these data, FG1 and CL2 were both generated from the EVN pipeline and have been copied over to these data.

Keeping track of your calibration tables is very important. I suggest writing down what is contained in each CL table, and any other tables as you create them.

Table	**Content**
CL 1	Pristine calibration table (always keep)
CL 2	A-Priori calibration: amplitude, parallactic angle (CL1+SNx+SNy+..)

There are many AIPS tasks that may be used to inspect your data and associated calibration tables. Some examples include `LISTR`, `PRTAN`, `PRTAB`, `SNPL`, `POSSM`, `UVPLT`, `VPLOT`.

AIPS Tip: You can learn more about a task/verb/adverb by typing `help abc`, where abc is the task/verb/adverb. For more details, you may also try `explain abc`.

Look at the observation summary using `LISTR`.

`default listr`	You can also use `task 'listr'`, but beware of previously set parameters!
`gent 1`	AIPS 1: SYMBOL? GENT
`opty 'scan'`	Produces a listing of every scan.
`inp`	
`go`	

```
 localhos  LISTR(31DEC15)      3     30-AUG-2015  00:33:00    Page    1
File = DATA        .SPLAT .   1 Vol = 1  Userid =    3
Freq =  4.927240000 GHz   Ncor =  4   No. vis =    185887
Scan summary listing

Scan      Source      Qual  Calcode Sub         Timerange          FrqID   START V
   1 3C345           : 0001           1  0/12:00:03 -   0/12:03:59     1        1
   2 3C345           : 0001           1  0/12:06:03 -   0/12:09:59     1     5322
   3 3C345           : 0001           1  0/12:12:03 -   0/12:12:59     1    11830
   4 3C345           : 0001           1  0/12:13:43 -   0/12:14:39     1    13372
   5 J1640+3946      : 0001           1  0/12:15:23 -   0/12:16:19     1    14938
   6 3C345           : 0001           1  0/12:17:03 -   0/12:17:59     1    16504
   7 J1640+3946      : 0001           1  0/12:18:43 -   0/12:19:39     1    17783
```

```
   8 3C345           : 0001           1  0/12:20:23 -   0/12:21:19     1   19062
   9 J1640+3946      : 0001           1  0/12:22:03 -   0/12:22:59     1   20628
  10 3C345           : 0001           1  0/12:23:43 -   0/12:24:39     1   22194
```

It is strongly recommended that you save a copy of the output somewhere as you will be referring to this regularly.

Questions for students: What sources are observed in these data? How long was the observation for? How many IFs are there?

We can use `prtan` to look at the antenna extension table. This is especially useful for figuring out the number AIPS assigns to each antenna. It is a good idea to store this information somewhere!

```
default prtan
getn 1
inp
go
```

```
Ant   1 = EF       BX=   425137.7629 BY= -1329919.0774 BZ=   753447.7862
Ant   2 = WB       BX=   222114.3845 BY= -1277412.1307 BZ=   917938.4967
Ant   3 = JB       BX=   -45741.1806 BY= -1813586.3689 BZ=   939502.9791
Ant   4 = ON       BX=   -72437.6057 BY=  -837626.2477 BZ=  1202680.9932
Ant   5 = NT       BX=  1600230.9267 BY=  -974712.0161 BZ=  -340498.4646
Ant   6 = TR       BX=   391796.8614 BY=  -496003.4529 BZ=   930053.6672
Ant   7 = SV       BX=  -275536.5331 BY=   208092.0135 BZ=  1382985.9353
Ant   8 = ZC       BX=  1028975.5745 BY=  1238596.8294 BZ=   244931.8515
Ant   9 = BD       BX= -2473769.3166 BY=  3842139.6437 BZ=   840687.6575
Ant  10 = SH       BX= -3910740.0471 BY=  5443755.5612 BZ=  -871655.7432
Ant  11 = HH       BX=  2326132.7752 BY=   170018.5180 BZ= -6915679.9457
Ant  12 = YS       BX=   829577.5785 BY= -2359756.9031 BZ=   -23898.1921
Ant  13 = JD       BX=   -45741.1806 BY= -1813586.3689 BZ=   939502.9791
```

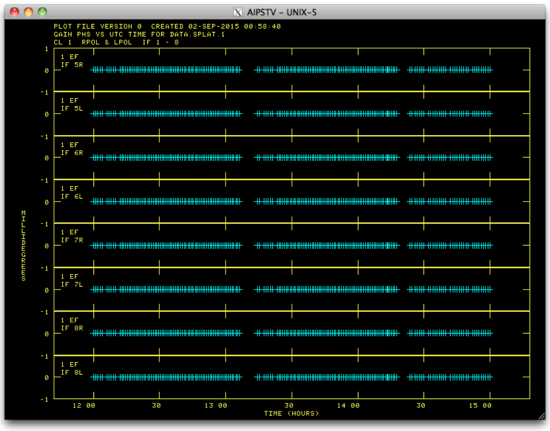
You can also use `prtab` to look at any of the extension tables. For example, if you take a look at the FG table you will see that the table contains flags for times when there are antennas off source.

Now let's take a look at the two available CL tables. Recall that CL1 is the "pristine" CL table and CL2 contains both a-priori amplitude corrections and parallactic angle corrections. `SNPL` plots phase against time by default.

`default snplt`	This task plots calibration *tables* against time
`getn 1`	
`inext 'cl'`	
`inver 1`	Plot CL1.
`dotv 1`	AIPS defaults to making PL files, but for now we will plot to tv
`nplots 8`	
`inp`	
`go`	

Make sure you look at the messages in your message server!

```
astaro> SNPLT1: Waiting 30 seconds:
astaro> SNPLT1:    Hit TV button A to pause indefinitely.  Hit button
astaro> SNPLT1:    B or C to continue sooner, button D to stop plotting
```

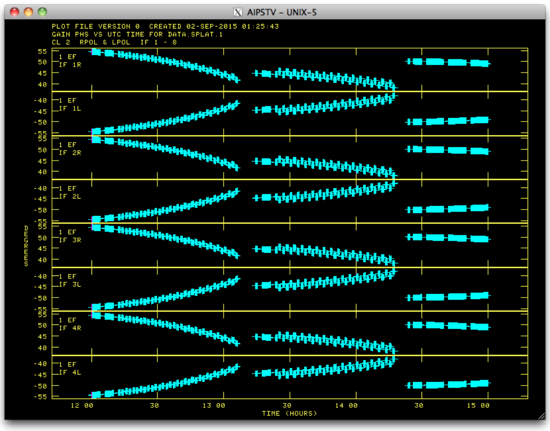


Now let's look at CL2.

AIPS Tip: "tget" is a useful verb to remember as it allows you to get all the parameters for a task from the last time the task was run, or the parameters were specifically saved using "tput".

AIPS Tip: Note that we are setting `inver 0` because 0 => highest in this task (see `help snpl`). This is *not* true for all tasks!

`tget snplt`	Get `SNPL` with the same parameters that were used last time.
`inver 0`	Plot CL2. You can also set this explicitly as `inver 2`.
`inp`	This is especially important if you are using `tget`!
`go`	



You will notice that there is a marked difference in phase between CL1 and CL2 for EF, but perhaps not much of a difference for WB. Any ideas why? (Hint: `tget prtan`)

We can use `POSSM` to plot the data against frequency. For this first plot we will not apply any calibration, which is the default.

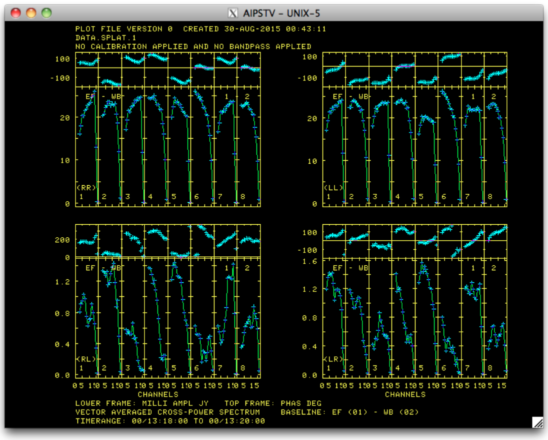
`default possm`	This tasks plots the data against frequency.
`getn 1`	
`timer 0 12 07 0 0 12 09 0`	A 2-minute chunk on 3C345.
`dotv 1`	
`aparm(9) 1`	Plots the IFs in the same frame. Feel free to try `aparm(9) 0`
`nplots 4`	Number of plots per page.
`anten 1 0`	Only plot antennas to 1 (which is EF)
`inp`	

```
go
```

You might notice that there is a baseline to EF missing. Which antenna is missing? Let's try a different source.

```
tget possm
timer XXXXXXXX
inp
go
```

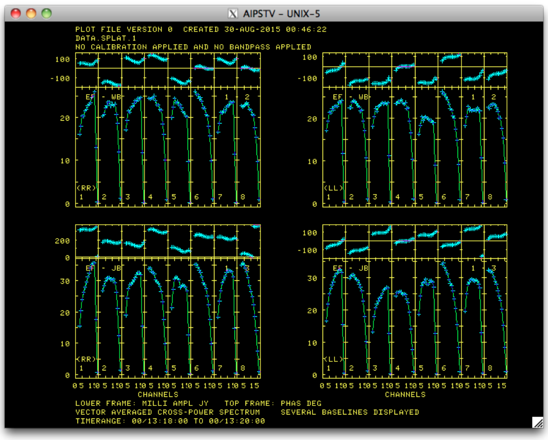
Choose a suitable source and timerange



We aren't going to worry about polarisation for this tutorial so for now we can forget about the cross-pols.

```
tget possm
stokes 'half'
inp
go
```

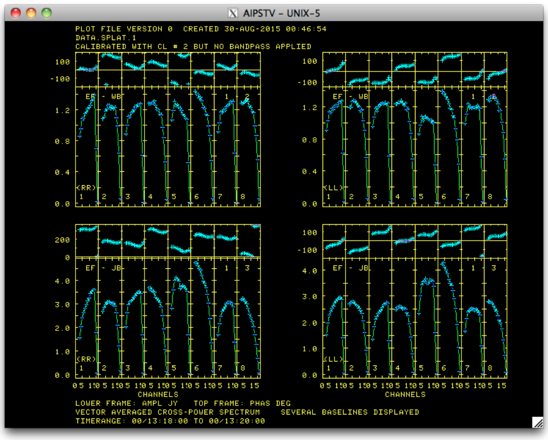
'half' = 'RR,LL'



Note the amplitude range... Recall that CL2 contains a-priori amplitude corrections so let's look at what happens when we apply them.

```
tget possm
docal 1
inp
go
```

Turn on calibration. In this case leaving `gainu 0` is ok.



You might notice that there is something wrong with SV's R-pol. It might be a good idea to flag this.

Bonus activity: You might find it fun to play with `UVPLT` to see how the amplitude scale changes.

6. Flagging

One of the nice things about VLBI is that most terrestrial RFI won't correlate! Nonetheless, there are times that flagging is necessary. AIPS has a number of flagging tasks such as `UVFLG`, `TVFLG`, `SPFLG` and `RFLAG`. We will only be using `UVFLG` today

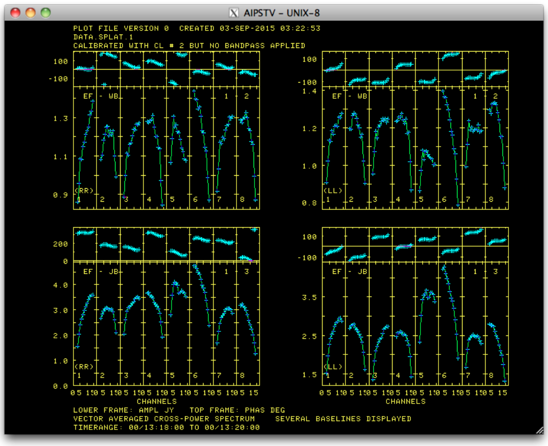
`default uvflg`	
`getn 1`	
`anten x 0`	Only want to flag SV
`stokes 'RR'`	Flag only the R-pol
`opco 'flag'`	Flag the data
`reason 'badpol'`	Give a reason... this is a bad polarisation.
`inp`	Super important when flagging!
`go`	

While we are here perhaps we should also get rid of the edge channels, which we could see in the previous `POSSM` plots had some zero values.

`tget uvflg`	
`anten 0`	Reset to flag all antennas
`stokes ''`	Reset to flag all pols
`bch 1; ech 2`	Flag channels 1 and 2
`reason 'edgech'`	
`inp`	
`go`	

`tget uvflg`	
`bch 15; ech 16`	Flag channels 15 and 16
`inp`	
`go`	

This will write the flags into FG1. Let's `tget POSSM` to see if it worked.



7. Remove instrumental delay

As you saw in the above POSSM plot, not only is there a gradient in phase within each IF, the mean phases for each IF are also quite different. This is due to the independent signal paths for each IF, and must be corrected for.

In this step we will be correcting for instrumental delays, the phase slope as a function of frequency, and as such we only want data over a short time period because we do not want changes as a function of time to come into play. This is why it is important to have bright calibrators for VLBI observations! The effect of removing instrumental delays is to bring all the phases to zero, at least for the chosen timerange.

`default fring`	
`getn 1`	
`timer 0 13 18 0 0 13 20 0`	Use the timerange that you used for `POSSM`
`docal 1; gainu 2`	`gainu 0` would work here too
`weightit 1`	This is usually done for EVN data.
`refant X`	What makes a good reference ant?
`solint 5`	This is longer than the timerange to ensure we get one data point only.
`dparm(9) 1`	Do not fit rate.
`inp`	
`go`	

Look at the header (`imh`). FRING should have generated a solution (SN) table. We will use `SNPL` to look at the delay solutions.

Table	Content
CL 1	Pristine calibration table (always keep)
CL 2	A-Priori calibration: amplitude, parallactic angle (CL1+SNx+SNy+..)
SN 1	Fringe Finder: instrumental delay/clocks

`default snplt`	
`getn 1`	
`inext 'sn'`	Look at the SN table. There is only 1 right now so the default for `inver` is ok.
`opty 'dela'`	

`dotv 1`

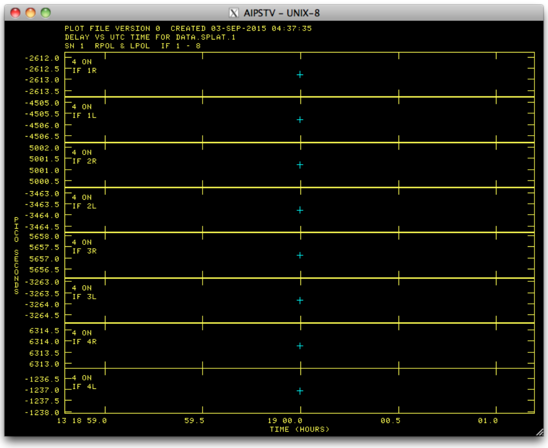
`nplots 8` — 8 is nice because you see all IFs in two pages

`inp`

`go`

Look puzzled at the spectrum that is still overlaid on your data then clean up the TV and go again ☺

`tvini; go`



The above plot is for ON IFs 1 - 4. Why does EF displays a zero delay correction?

When you are happy with the SN table, you can apply it to your CL table (CL2+SN1 = CL3). It is best to be explicit with `gainver` and `gainuse` when using CLCAL.

`default clcal`

`getn 1`

`gainv 2` — Using CL2

`gainu 3` — Create CL3

`snver 1` — Using SN1

`refant 1`

`inp`

`go`

Table	Content
CL 1	Pristine calibration table (always keep)
CL 2	A-Priori calibration: amplitude, parallactic angle (CL1+SNx+SNy+..)
SN 1	Fringe Finder: instrumental delay/clocks
CL 3	CL2+SN1: Cumulative calibration

We will use `SNPL` to look at CL3.

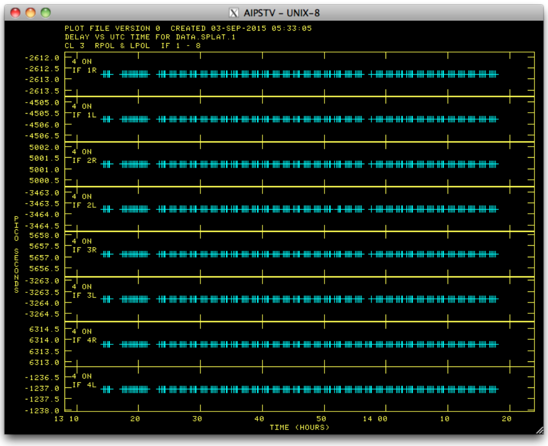
`tget snplt`

`inext 'cl'` — Look at the CL table.

`inver 3` — `inver 0 is ok here too`

`inp`

`go`



We can use `POSSM` to see how the phases have been corrected.

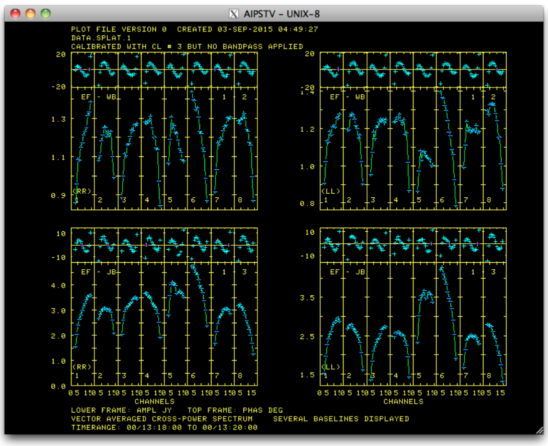
Plot data prior to global delay correction

```
default possm
getn 1
timer 0 13 18 0 0 13 20 0
dotv 1
docal 1; gainu 2
aparm(9) 1
nplots 4
anten 1 0
stokes 'half'
inp
go
```

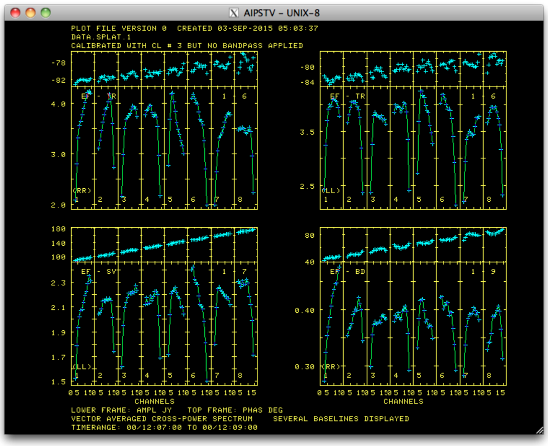
Apply CL2

```
tget possm
gainu 3
inp
go
```

Current CL table. `gainu 0` works here too.



Bonus activity: Choose a source in a different part of the sky and look at `POSSM`. Is the phase still flat?



8. Frequency and time-dependent phase calibration

We have corrected for the instrumental delay. However, we now need to correct for delay and rate as a function of *time*. Consequently we will be using a smaller solution interval.

`default fring`	
`getn 1`	
`calsour '1848+283'''`	
`docal 1; gainu 3`	`gainu 0` would work here too
`weightit 1`	
`refant 1`	
`solint 1`	Solution interval of 1 minute
`aparm(5) 1`	Combine all IFs for improved SNR
`aparm(9) 1`	Turns on `search` (below)
`dparm 1 200 50 1`	The delay and rate windows within which to find solutions.
`search 1 3 2`	If solutions fail using 1 (EF), then try 3 (JB) ...
`inp`	
`go`	

astaro> FRING1: Found 4040 good solutions
astaro> FRING1: Failed on 184 solutions

Table	Content
CL 1	Pristine calibration table (always keep)
CL 2	A-Priori calibration: amplitude, parallactic angle (CL1+SNx+SNy+..)
SN 1	Fringe Finder: instrumental delay/clocks
CL 3	CL2+SN1: Cumulative calibration
SN 2	Fringe Fit/Rate (atmospheric phase/delay)

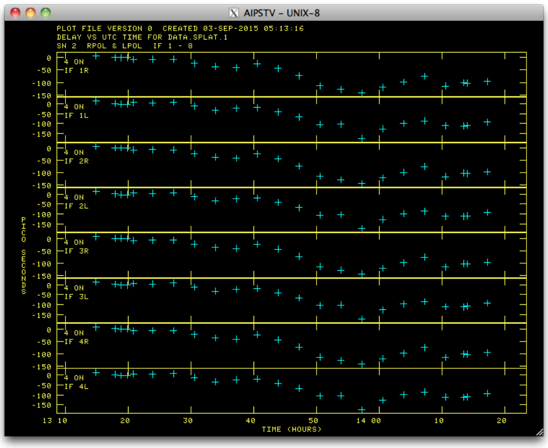
`default snplt`	
`getn 1`	
`inext 'sn'`	
`inver 2`	`inver 0` would work here as well
`opty 'dela'`	Also look at phase and rate.

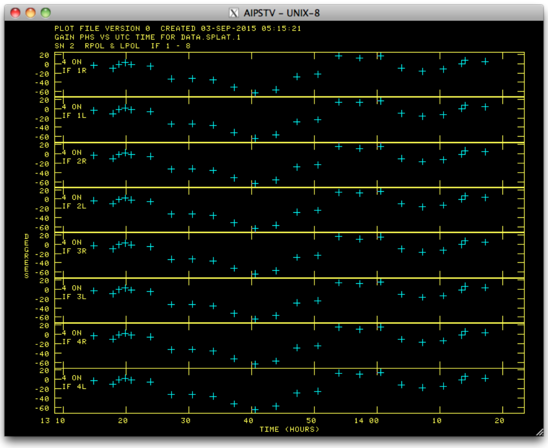
```
dotv 1
```

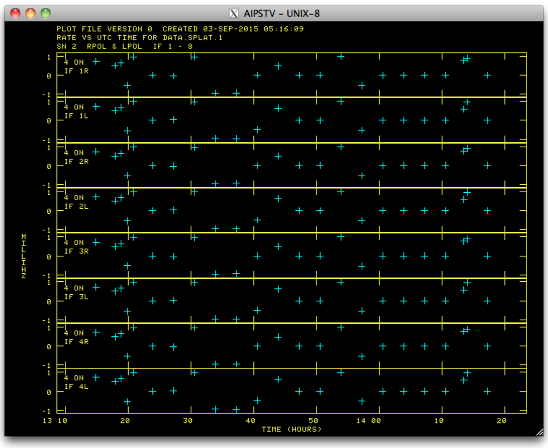
```
nplots 8
```

```
inp
```

```
go
```







All the above plots show the solutions for ON.

When you are happy with the SN table, you can apply it to your CL table (CL3+SN2 = CL4).

As this is a phase referencing experiment, we are applying the phase calibration we have made for 1848+283 to our "target", J1849+3024. What is the physical distance between these two sources? Can we use these solutions to phase calibrate the other sources in these data?

```
default clcal
```

```
getn 1
```

`calsour '1848+283'''`

`sour '1848+283''J1849+3024'`

`gainv 3` The default would also work here.

`gainu 4` The default would also work here.

`snver x` This must be set explicitly, otherwise both SN1 and SN2 would be applied

`refant 1`

`inp`

`go`

Table	**Content**
CL 1	Pristine calibration table (always keep)
CL 2	A-Priori calibration: amplitude, parallactic angle (CL1+SNx+SNy+..)
SN 1	Fringe Finder: instrumental delay/clocks
CL 3	CL2+SN1: Cumulative calibration
SN 2	Fringe Fit/Rate (atmospheric phase/delay)
CL 4	CL3+SN2: Total calibration

`tget snplt`

`inext 'cl'`

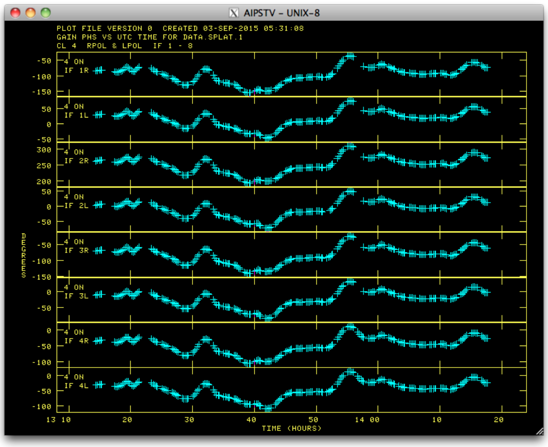
`inver 4` `inver 0` would work here as well

`sour '1848+283''J1849+3024'`

`opty 'phas'` Also look at delay and rate.

`inp`

`go`



9. Bandpass calibration

Bandpass calibration corrects for the response of the receiver as a function of frequency.

`default bpass`

`getn 1`

`calsour '1848+283'''`

`docal 1` Apply calibration (in this case CL4)

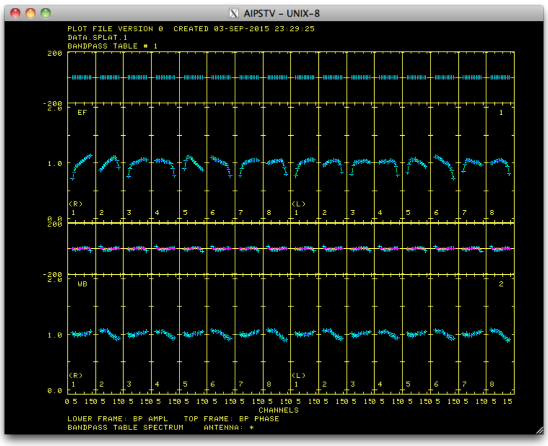
`refant 1`	
`solint -1`	Use whole time range.
`weightit 1`	
`inp`	
`go`	

Table	Content
CL 1	Pristine calibration table (always keep)
CL 2	A-Priori calibration: amplitude, parallactic angle (CL1+SNx+SNy+..)
SN 1	Fringe Finder: instrumental delay/clocks
CL 3	CL2+SN1: Cumulative calibration
SN 2	Fringe Fit/Rate (atmospheric phase/delay)
CL 4	CL3+SN2: Total calibration
BP 1	Bandpass calibration

Look at the bandpass *table*.

`default possm`	
`getn 1`	
`dotv 1`	
`aparm 0, 1, 0, 2, -180, 180, 0, 2, 3`	`aparm(8) 2` plots the bandpass table.
`nplots 2`	
`stokes 'half'`	
`inp`	
`go`	

AIPS Tip: Commas are necessary for `aparm` here, otherwise AIPS/POPS treats "1.5 -180" as an expression.

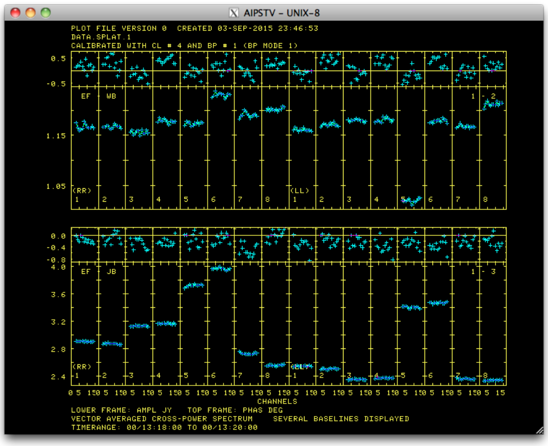


Now look at the the BP table applied to the *data*.

`tget possm`	
`docal 1; gainu 4`	Default gainu will work too.
`doband 1; bpver 1`	Turn on bandpass calibration
`aparm 0`	
`aparm(9) 3`	

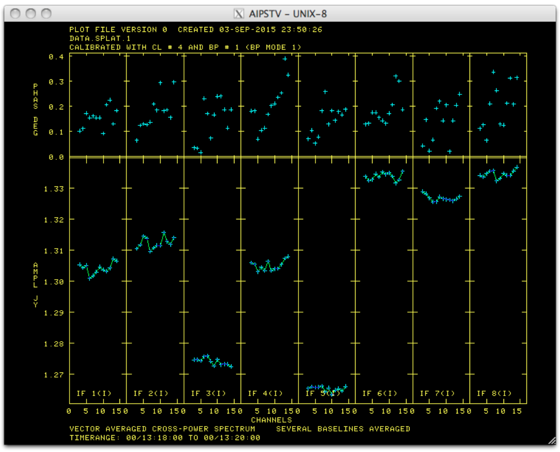
`timer 0 13 18 0 0 13 20 0`	You can also select on source
`anten 1 0`	
`inp`	
`go`	

AIPS Tip: AIPS recognises many Emacs shortcuts, such as ctrl-r for reverse searching.



We are working towards a continuum (Stokes I) image. This requires the data to be combined for all baselines averaged over all frequencies and time. We can get a feeling if this is sensible by averaging across all baselines and time in `POSSM`.

`tget possm`	
`anten 0`	All antennas
`nplots 0`	Average all data to produce one plot.
`stokes 'i'`	Because this is what we will be imaging.
`inp`	
`go`	



10. Apply the calibration (Split the data)

We are now ready to apply our calibration tables to the data. We will split the data to make imaging easier, and in the process of splitting the data we will apply CL4, BP1 and FG1 to these data.

`default split`	
`getn 1`	
`sour '1848+283''J1849+3024'`	We only want to split off the sources we've calibrated for

	now.
`doband 1; bpver 1`	Apply the bandpass table.
`docal 1; gainu x`	Apply the calibration table.
`flagv 1`	Apply the flags.
`aparm 2 1 0`	Average all channels per IF.
`inp`	
`go`	

Take a look at the available catalogue entries now (pcat).

```
>pca
AIPS 1: Catalog on disk  1
AIPS 1:  Cat Usid Mapname      Class   Seq  Pt     Last access      Stat
AIPS 1:    1    3 DATA        .SPLAT .    1 UV 30-AUG-2015 21:04:34
AIPS 1:    2    3 J1849+3024  .SPLIT .    1 UV 30-AUG-2015 21:04:52
AIPS 1:    3    3 1848+283    .SPLIT .    1 UV 30-AUG-2015 21:04:34
```

When you look at the header you will find that there are markedly less extension files that are associated with the data...

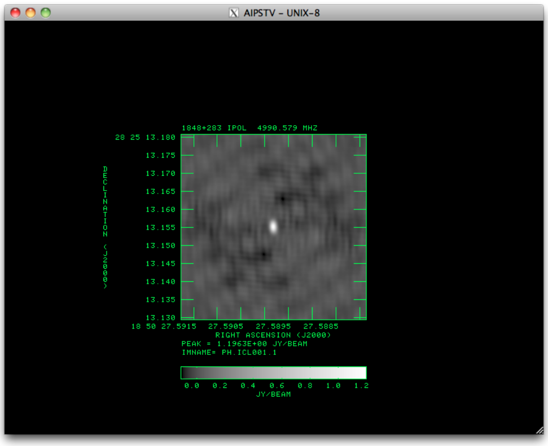
11. First Pass Image...

This is the first pass image of the phase calibrator that will be used as a model for future self-calibration iterations. Of course you care about your science target, but I like to always take a quick look at my calibrator to make sure there aren't obvious issues. As we will be cleaning interactively, the following steps will be explained in much more detail during the tutorial!

`default imagr`	
`getn 3`	
`imsiz 256`	No. of pixels across the image
`cell XXXXX`	Size per pixel in arcsec.
`outna 'PH'`	
`niter 1000`	
`robust 0`	Image weighting. 0 is somewhere between uniform and natural.
`dotv 1`	Clean interactively.
`inp`	
`go`	

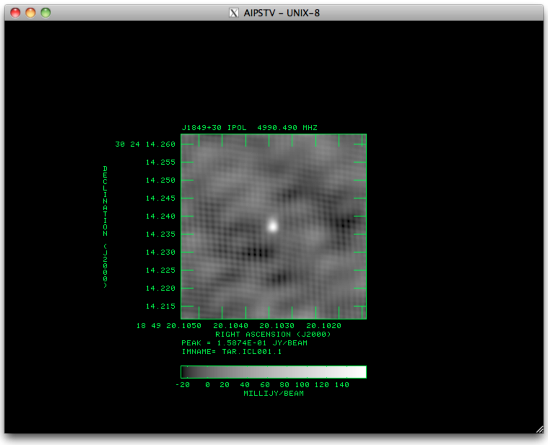
Let's take a look at our image! Below are some tasks to play with.

`reca`	Renumber the catalogue files. Not super important.
`getn n`	n is the catalogue number of your clean map (ICL001)
`tvall`	Show the image on the tv
`tvlab`	Overplot labels
`tvbox`	Draw a box around the source.
`imstat`	Image statistics. Turn inver on and off to measure the peak/noise.



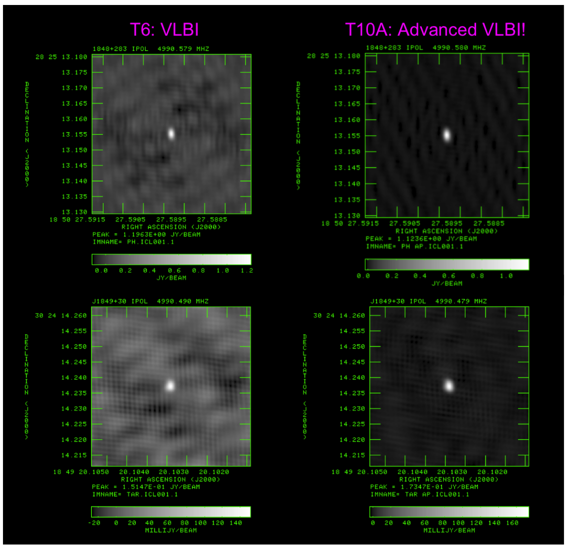
This is the first pass image of the target that will be used as a model for future self-calibration iterations

```
tget imagr
getn 2
outna 'TAR'
inp
go
```



12. Final Image...

Some eye-candy to tempt you to coming to Tutorial 10A ☺



13. Closing AIPS

To close AIPS cleanly...

```
kleenex
```