

Hands-on Activities on DiFX

Alessandra Bertarini

IGG University of Bonn & MPIfR Bonn

Roger Cappallo

MIT Haystack Observatory



In this exercise we are using R1572.

R1572 is part of a weekly experiment dedicated to measuring UT1-UTC.

It is a dual band (8 GHz and 2 GHz) experiment

One polarization (RCP)

One bit sampling

Stations in this example Onsala and Wettzell.

Observed on 12th Feb 2013.

24 h long, but ~ 40 s of data for these exercises.



Correlator's VEX files need extra information:

Earth rotation parameters (x-wobble, y-wobble and UT1)

Clock information (gps-fmout from field system logs)

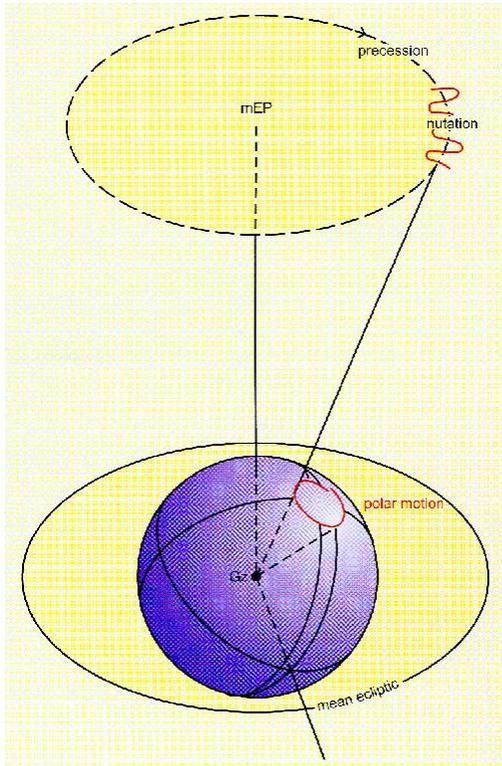
Data source (module, files on RAID)



Correlator's VEX files need (sometimes) to be changed:

Track assignment (only tape-like tracks are present in VEX)

Earth Rotation Parameters (x-wobble, y-wobble and UT1)



the predicted values are published from US Naval Observatory:

http://128.183.20.176/solve_save/usno_finals.erp

DiFX ancilliary program `geteop.pl` read the USNO file, reformat it and creates a file called `EOP.txt`

Log in to: `bicke.oso.chalmers.se -- user: difxnn`

(where nn = 01 to 10 e.g. difx01).

Change directory to `/common/difxnn/R1572`



Task 1:

Students will need to download VEX and the field system (FS) log files and other ancilliary files from:

`ftp://ftp.mpifr.bonn.mpg.de/outgoing/abertari/EVGA/r1572`

`mget *.log`

`get r1572.vex`

`get EOP.txt`

Task 2: Get familiar with the VEX file.

VEX is divided into sections. All sections have a \$SECTION (i.e. \$MODE, \$STATION \$SCHED, \$CLOCK....)

Within a section there are definitions that point to other sections:

```
$STATION
```

```
def Ft;
```

```
    ref $ANTENNA = FORTLEZA;
```

```
enddef;
```

```
$MODE;  
  
def GEOSX-SX;  
  
ref $FREQ = GEOSX-SX01:Ft:Hh:Ke:Ny:Ts:Wz:Yg;  
ref $BBC = GEOSX-SX01:Ft:Hh:Ke:Ny:Ts:Wz:Yg;  
ref $IF = GEOSX-SX01:Ft:Hh:Ke:Ny:Wz:Yg;  
ref $TRACKS = Mk34121-SX01:Ft:Wz;  
ref $TRACKS = Mark5B:Hh:Ke:Kk:Ny:Tc:Yg;  
$PHASE_CAL_DETECT = Standard:Ft:Hh:Ke:Kk:Ny:Wz;  
  
[...]  
  
enddef;
```



```
$STATION
```

```
def Ny;
```

```
ref $SITE = NYALES20;
```

```
ref $ANTENNA = NYALES20;
```

```
enddef;
```

```
$ANTENNA
```

```
def NYALES20;
```

```
antenna_diam = 20.00 m;
```

```
axis_type = az : el;
```

```
axis_offset = 0.51980 m;
```

```
antenna_motion = az : 120.0 deg/min : 9 sec;
```

```
antenna_motion = el : 120.0 deg/min : 9 sec;
```

```
enddef;
```



```
$FREQ
```

```
def GEOSX-SX01;
```

```
chan_def = &X : 8212.99 MHz : U : 8.000 MHz : &CH01 : &BBC01 :&U_cal;
```

```
chan_def = &X : 8212.99 MHz : L : 8.000 MHz : &CH02 : &BBC01 :&U_cal;
```

```
chan_def = &X : 8252.99 MHz : U : 8.000 MHz : &CH03 : &BBC02 :&U_cal;
```

```
chan_def = &X : 8352.99 MHz : U : 8.000 MHz : &CH04 : &BBC03 :&U_cal;
```

```
chan_def = &X : 8512.99 MHz : U : 8.000 MHz : &CH05 : &BBC04 :&U_cal;
```

```
chan_def = &X : 8732.99 MHz : U : 8.000 MHz : &CH06 : &BBC05 :&U_cal;
```

```
chan_def = &X : 8852.99 MHz : U : 8.000 MHz : &CH07 : &BBC06 :&U_cal;
```

```
chan_def = &X : 8912.99 MHz : U : 8.000 MHz : &CH08 : &BBC07 :&U_cal;
```

```
chan_def = &X : 8932.99 MHz : U : 8.000 MHz : &CH09 : &BBC08 :&U_cal;
```

```
chan_def = &X : 8932.99 MHz : L : 8.000 MHz : &CH10 : &BBC08 :&U_cal;
```

```
chan_def = &S : 2225.99 MHz : U : 8.000 MHz : &CH11 : &BBC09 :&U_cal;
```

```
chan_def = &S : 2245.99 MHz : U : 8.000 MHz : &CH12 : &BBC10 :&U_cal;
```

```
chan_def = &S : 2265.99 MHz : U : 8.000 MHz : &CH13 : &BBC11 :&U_cal;
```

```
chan_def = &S : 2295.99 MHz : U : 8.000 MHz : &CH14 : &BBC12 :&U_cal;
```

```
chan_def = &S : 2345.99 MHz : U : 8.000 MHz : &CH15 : &BBC13 :&U_cal;
```

```
chan_def = &S : 2365.99 MHz : U : 8.000 MHz : &CH16 : &BBC14 :&U_cal;
```

```
sample_rate = 16.0 Ms/sec;
```

```
enddef;
```

Example of v2d file:

vex = vex file name

antennas = two letter code of the participating stations (e.g. antennas = NY, WZ...)

singleScan = True/False

SETUP r1572 → like a VEX section

{

tInt = integration time in second

doPolar = True/False

nChan = no. spectral channels

}

Task 3: Calculate the recording speed knowing that:

Nyquist sample rate = 16 Msample/s

Number of BBC channels = 16

Polarization = only RCP

Bit sampling = 1 bit



Task 4: Insert the EOP in the VEX file.

Insert the file EOP.txt (generated by geteop) in the VEX (no special position as long as it does not "break a \$SECTION or is within a def / enddef)

DiFX needs to have 5 values of EOP and two of them must be prior to the observation start.

E.g. if observation starts at Day of the Year (DOY) 100, then geteop needs as start DOY 98.

DOY of r1572 is found in \$SCHED section in the scan name (e.g. 100-1700 => DOY-HHMM).

Task 5: Clock offset and rate correction in the vex

now is:

```
$CLOCK;
```

```
def Ny;
```

```
clock_early=2012y142d17h00m : 0 usec :2012y142d17h00m0s : 0 ;
```

```
enddef;
```

```
def Wz;
```

```
clock_early=2012y142d17h00m : 0 usec :2012y142d17h00m0s : 0;
```

```
enddef;
```

The 0 us are the gps-fmout values that can be found in the FS log files.

The 0 s/s are the clock drifts and need to be calculated

fgrep gps-fmout from the r1572on.log and r1535wz.log

e.g.:

2011.100.19.23.44.00/gps-fmout/+4.345E-006

2011.100.19.31.13.00/gps-fmout/+4.312E-006

[..]

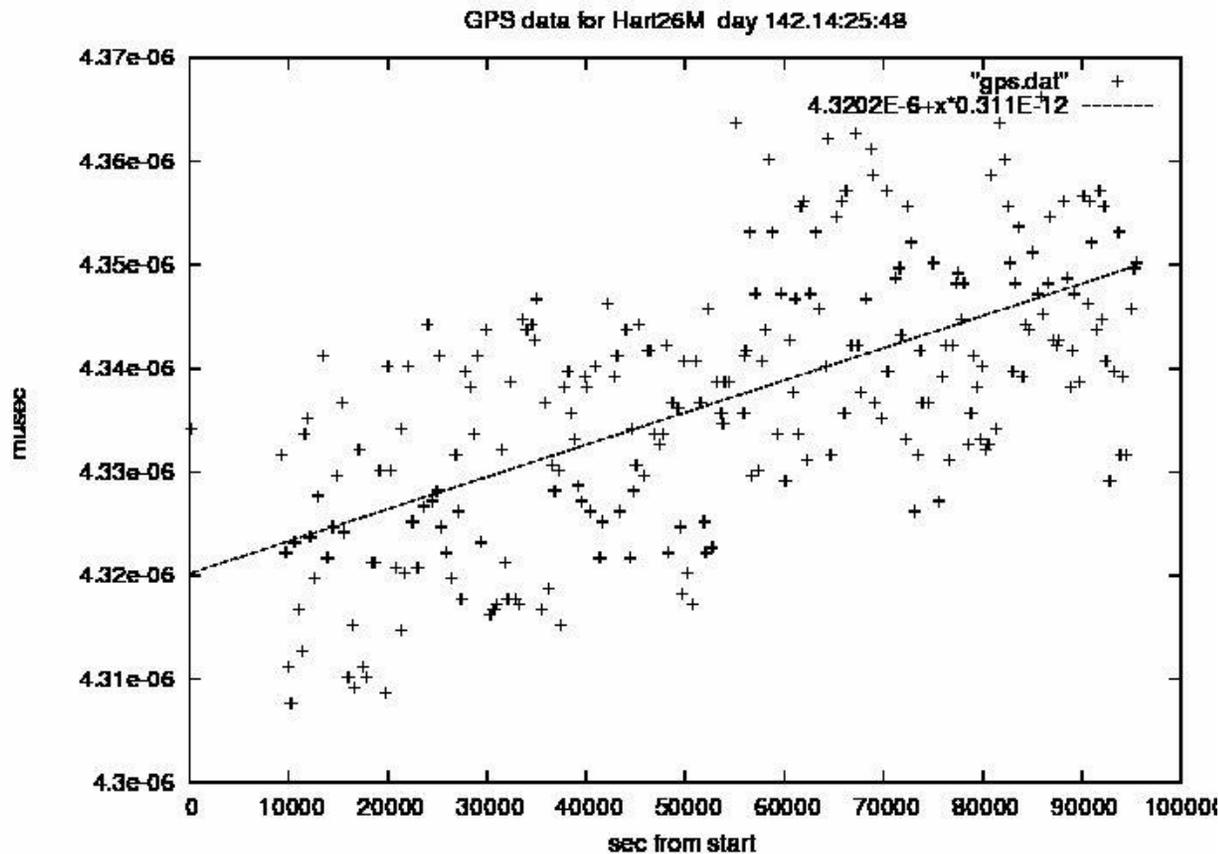
2011.101.16.58.12.99/gps-fmout/+4.3502E-006

gps-fmout is monitored very often.

Take the first value and insert it in the vex.



Clock rate calculation:



Normally correlator operators perform linear fits to the clock data before calculating drifts.



Copy the value of the first gps-fmout in the log at the place of the 0.

Calculate the drift and insert it in the file at the place of the 0.

Note the drifts are in s/s, while the clock in us !

```
$CLOCK;  
def Ny;  
clock_early=2012y142d17h00m : 0 usec :2012y142d17h00m0s : 0;  
enddef;  
def Wz;  
clock_early=2012y142d17h00m : 0 usec :2012y142d17h00m0s : 0;  
enddef;
```



DiFX does not read VEX file directly.

It takes the information from the VEX and produces other files.

The program `vex2difx` does this operation.

`vex2difx` needs one ancilliary file called `r1572.v2d` and is run:

```
vex2difx r1572.v2d
```

Example of v2d file cont.:

```
RULE clock{  
    scan = scan name  
    setup = r1572
```

```
}
```

```
ANTENNA AB
```

```
{
```

```
    file = path/ab_data
```

```
}
```



Task 6: Get familiar with the syntax of the v2d.

<http://cira.ivec.org/dokuwiki/doku.php/difx/vex2difx>

R1572 was observed using only one polarization (RCP) =>
doPol = False

We have only one scan => **singleScan = True**

Scan name → DOY-HHMM => **043-1700b**

File: DiFX must know whether the data are on Mark 5 module or in a file. If 'file =' is in v2d then data comes from files.

so **file = path/onsala_data ...**

(**path = /mnt/2/R1572/On/r1572_on_043-1700b and**
/mnt/6/R1572/Wz/r1572_wz_043-1700b)

Note both stations require an **ANTENNA** block with the respective file.

Task 7: Insert the tInt and nChan in the v2d file.

As start use tInt = 1 s and nChan = 32

Note that no of lag = no. of spectral channel (in DiFX!).

Task 8: Enter the remaining parameters described above in the v2d file for r1572.

Task 9: Run vex2difx. Which files have been created?

Task 10: Run the correlator =>

```
calcif2 <r1572_1.calc>
```

```
genmachines <r1572_1.input>
```

```
startdifx -n r1572_1.input
```

Task 11: Run `difx2mark4` to create the files for inspecting the data using Haystack software: *fourfit*

Task 12: Fringe fit the correlator output by running *fourfit*. Change directory to 1234, and run *fourfit*:

`fourfit -pt -c cf_1234 043-1700b`

Task 13: Check the SBD. Is it at zero μs ?

SBD $\neq 0$ \Rightarrow non modelled station-based errors. But we can correct them!

Task 14: Change the clock_early entry in the vex file.

SBD is baseline-based \Rightarrow take one of the two station as reference and correct the other.

Suppose we have baseline AB

Suppose SBD = $0.6 \mu\text{s}$ on baseline AB.

Suppose clock_early A = $1.5 \mu\text{s}$ and clock_early B = $-3.0 \mu\text{s}$

If A is chosen to be the reference station \Rightarrow

clock_early B = $-3.0 + 0.6 = -2.4 \mu\text{s}$

If B is reference \Rightarrow clock_early A = $1.5 - 0.6 = 0.9 \mu\text{s}$

We have changed the clock model. Need to check it.

Remove the old *.input, calc, im, difx files

Re-run vex2difx - startdifx - difx2mark4 - fourfit.

Task 15: Is the SBD better?

Now we break things!

Task 16: Change the station position or source position or clock models. Increase/reduce tInt and nChan. Run the correlation and look at the data with fourfit. What happens?

Edit fourfit control file:

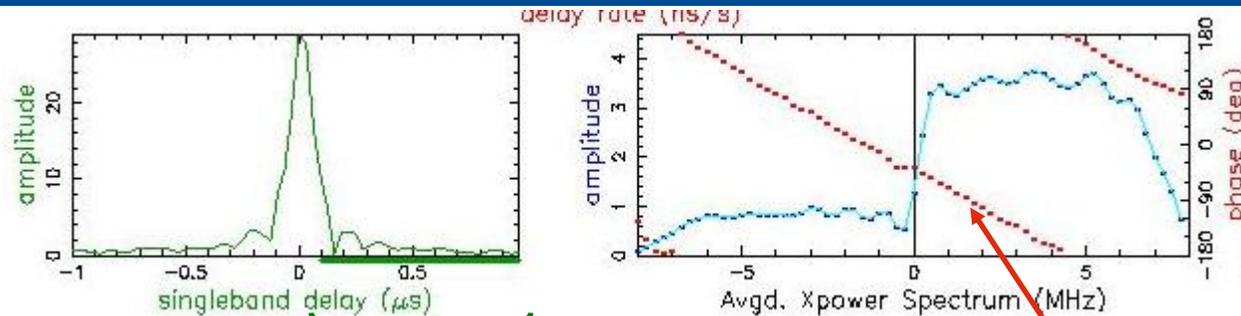
Task 17: Change the `sbd_win` to force the SBD off the peak. What happens to the phase in the spectrum?

Task 18: Do as for task 18, but using the `mbd_win`. What happens to the phases vs time?

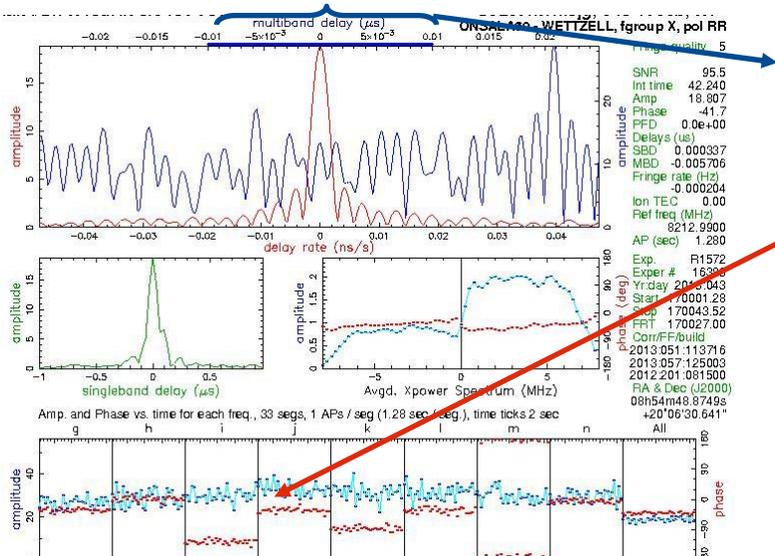
Task 19: Do as for task 18, but using the `dr_win`. What happens at the phase vs time?

Task 20: Reduce the integration time, by changing the start. What is the SNR at which fringes disappear?



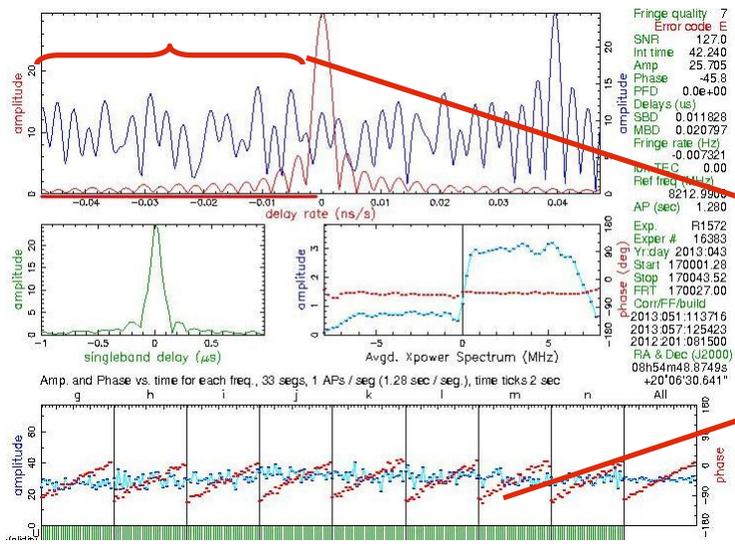


SBD window excludes the peak => phase has residual slope because correct solution is excluded by the window.



MBD window excludes the peak => phase/BBC is different, corresponding to a residual slope.

-
-
-
-
-
-
-



DR window excludes the peak => causing a residual slope of the phase vs time.

Change the correlator model.

Task 21: Correlate the scan changing the calc file (not by much).

Look in the fourfit plot at the residual delay and residual rate in both cases. Are they changed?

Now look at the total delay: are those changed?

Conclusions....

Game Over ...

Try Again?

