DIFX Monitor

DiFX has hooks to dump a copy of all accumulated visibilities down a TCP stream for online viewing, processing etc. Only single program can receive this stream. To allow multiple clients to simultaneously receive this monitoring data, a monitor_server can be run which receives a copy of ALL visibilities and allows multiple clients to connect. The clients request which of the visibilities they require and monitor_server only sends on the products required (so e.g. a client can be run over a moderately slow link if it only requires a handful of products).

Usage

Add the option `-Mhost:port` to the mpifxcorr options, where `host` is the host running monitor_server, and port is the TCP port to use. Note typically the host will be localhost

e.g.

```
 mpirun -machinefile machines -np 10 /home/vlbi/difx/bin/mpifxcorr
v999.input -Mlocalhost:9999
```

Either before or after mpifxcorr is started, monitor_server should be run. monitor_server would normally be run on the headnode running difx, but does not have to be (as long as mpifxcorr is told which host it is running on). monitor_server is designed to run persistently, ie it is not started and stopped each time a job is run. Note that two instances of monitor_server cannot be run on the same host, nor can monitor_server support two simultaneous instances of mpifxcorr. monitor server is simply run as `monitor_server PORT` where port is the same port mpifxcorr was told.

e.g.

```
monitor_server 9999
```

Clients

Clients connect to monitor_server to display a subset of the visibilities. Note that most clients require a copy of the difx .input file. If they are run on a different host than difx (and without a shared filesystem). The .input file and some other setupfiles (e.g. .uvw) will need to be copied.

difx_config

`difx_config` Prints the product numbers (used by the following clients) and the associated baseline, frequency, polarisation etc.

Usage:

```
difx_config <experiment.input>
```

e.g.

```
difx_config v999.input
0: CATW102-MOPRA   RR 1405 MHz (16 MHz)
1: CATW102-MOPRA   LL 1405 MHz (16 MHz)
2: CATW102-MOPRA   RR 1421 MHz (16 MHz)
3: CATW102-MOPRA   LL 1421 MHz (16 MHz)
```

mon_sample

`mon_sample` was originally a coding example for using the monitoring infrastructure, but has proved to be a useful program in its own right. The name may change with no notice! mon_samples plots amplitude and phase vs frequency and amplitude vs delay (ie lag function) for up to 4 products.

Usage:

```
mon_sample  <host> <inputfile> <product#> [<product#> ...]
```

where host is the host running monitor_server, inputfile is the difx input file *currently being used* and up to 4 product numbers can be specified.

e.g.

```
mon_sample localhost v999.input 0 8 16 24
```

difx_monitor

`difx_monitor` creates an html table and plots a .png plot for every visibility. Because the plotting is quite slow, it might be advisable to use a longer integration time if either the original integration time is short or a lot of products need to be plotted (e.g. large number of baselines). To be useful difx_monitor needs to be run on a host with a web server (and run in a directory visible to the web server).

Usage:

```
difx_monitor <inputfile> <host> [intergration]
```

where host is the host running monitor_server, inputfile is the difx input file *currently being used*. integration is an optional argument which gives a longer integration time (in seconds) to make plots for. The actual integration time sued will be the closest integral number of difx integrations to this number.

e.g.

```
difx_monitor v999.input cave00-ext 10
```

e.g.

```
mon_sample localhost v999.input 0 8 16 24
```

vcal

Vcal calculates the average delay of each station to a reference antenna then updates the input file to remove the offsets. DiFX needs to be restarted before these changes are seen. For each non-reference station, the delay offset for each baseline to the reference is computed then those above a SNR threshold averaged (unweighted). The algorithm for computing the delay is very simple and only good to 0.5 samples.

Usage:

```
vcal <inputfile> <host> [refant [# samples]]
```

where inputfile is the inputfile currently being used, host is the host running difx_monitor. `refant` is an optional argument to change the referenced antenna from the default (the first in the input file). By default a simple integration is used - by setting the 4th argument more can be averaged together.

mon3d

mon3d displays a 3D fringe plot of a single product (a 2D FFT of the visibilities, so showing the delay and rate). It is useful for public demos etc. It uses the S2plot 3D libraries.

Usage:

```
mon3d <product>
```

Building

monitor_server and the clients are not built by default. Change directory to applications/difx_monitor (and trunk/branches etc if appropriate) and type make. You will need to install by hand. You can also try:

```
install-difx --withmonitor
```

which will attempt to compile and install monitor_server and clients.