Reading Mark6 data

Use vdifuse in order to mount recorded diskpacks.

Preparation

vdifuse needs a mount point for the diskpack content (e.g. /mnt/diskpack).

As user root:

```
mkdir /mnt/diskpack
mkdir /mnt/diskpack/stream1
mkdir /mnt/diskpack/stream2
mkdir /mnt/diskpack/group
chmod -R a+rwx /mnt/diskpack
```

Stopping the schedule

Not sure if this is neccessary (Geoff?) but before mounting the modules it is probably safer to stop the recording schedule and close the module group:

- schedule execution can be stopped by ^C the relevant program (M6_CC or start_eht.py)
- in da_client:

```
group=close:1234
```

Mounting the module group

the module content can be mounted in different ways:

- mounting the whole 4-module group. This will result in a single vdif file per scan containing 2 vdif threads.
- mounting stream-based subgroups. This will result in two single threaded vdif files per scan.

mounting the 4 module group

as user oper:

```
vdifuse -a test.cache -xm6sg /mnt/diskpack/group /mnt/disks/?/?/data
```

the 2-thread vdif files will be visible under

```
/mnt/diskpack/group/sequences/{expname}/{stationcode}
```

mounting subgroups

as user oper:

```
vdifuse -a test.cache -xm6sg /mnt/diskpack/stream1 /mnt/disks/[12]/?/data
vdifuse -a test.cache -xm6sg /mnt/diskpack/stream2 /mnt/disks/[34]/?/data
```

the single-thread vdif files will be visible under

```
/mnt/diskpack/stream1/sequences/{expname}/{stationcode}
/mnt/diskpack/stream2/sequences/{expname}/{stationcode}
```

Unmounting

as user oper:

depending on the way the modules were mounted:

```
fusermount -u /mnt/diskpack/group
```

or

```
fusermount -u /mnt/diskpack/stream1
fusermount -u /mnt/diskpack/stream2
```

Restart the schedule

in da_client:

```
group=open:1234
```

then restart your schedule execution.