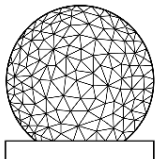


# Mark6 Operations

**12<sup>th</sup> IVS TOW Workshop**  
**Chester “Chet” Ruszczyk**  
**[chester@mit.edu](mailto:chester@mit.edu)**



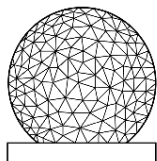
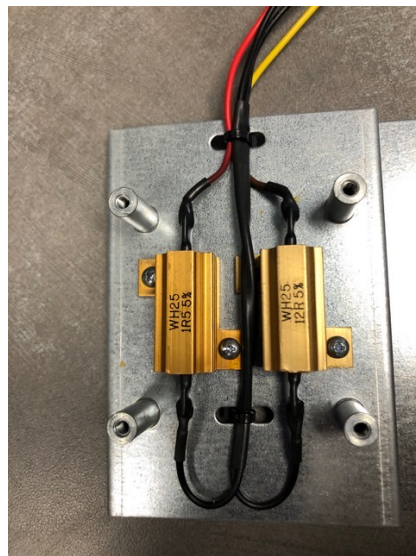
**MIT**  
**HAYSTACK**  
**OBSERVATORY**

# Objective

- Mark6 General Information
- Mark6 Applications
- Disk Modules
- Recording
- Play Back / Prepping for e-transfer
- Next Steps

## Mark6 Expansion Chassis Note

- How many folks have updated the Mark6 expansion chassis shunt resistor configuration?
- [https://www.haystack.mit.edu/wp-content/uploads/2023/02/010\\_MARK6.pdf](https://www.haystack.mit.edu/wp-content/uploads/2023/02/010_MARK6.pdf)

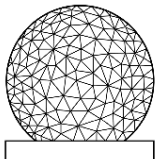


# Mark6 General Information

- Two versions of OS in the wild
  - Debian
  - CentOS7
- Debian's have been end of life (EOL) for many years
- Newer Mark6's have CentOS7
  - EOL (June 30, 2024)
- Presently evaluating new OS's distributions due to security concerns and impact on software / performance
  - Ubuntu 22.04 is the main candidate
  - Supported in LTS till 2032

## Mark6 General Information

- Setup
  - Cabling for SAS controllers
  - Order is not critical but important
    - Why?
      - Individual disk information using the ***disk\_info*** command is based upon certain order.
      - If a disk fails, poor performance there is not a one to one correspondence unless cabling is consistent.
      - You will have to determine it by probing additional ***disk\_info*** states.
        - A disk detective
      - Only on older HBA version 2 cards, V3 cards behave differently in bringing up HDDs in module.



# Cabling for HBA Controller Cards



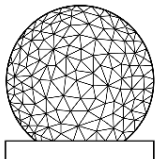
Version 2



Version 3

- Dependent on Version 2 vs. Version 3 HBA cards
  - Cable connectors are different
- Yellow / Red Dots to aid in connection cables
- We put stickers on the cables / disk modules
- If you do not use stickers there is a rule of thumb to follow
  - White label on cable is always on top
    - Represents the red dots

# Cable Connection



**MIT**  
**HAYSTACK**  
**OBSERVATORY**

## On Boot Up

- SAS controller cards bios executes before motherboard bios
  - Enter and disable boot up from disks attached to Controllers.
    - Now if the system reboots with disk modules keyed on
    - It will not look for a master boot record on the disk modules
    - It will boot normally and not hang since no OS is found



# General

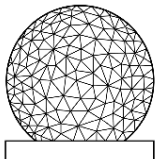
## • Setup

### • Ethernet Interfaces

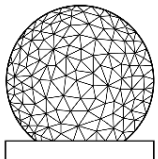
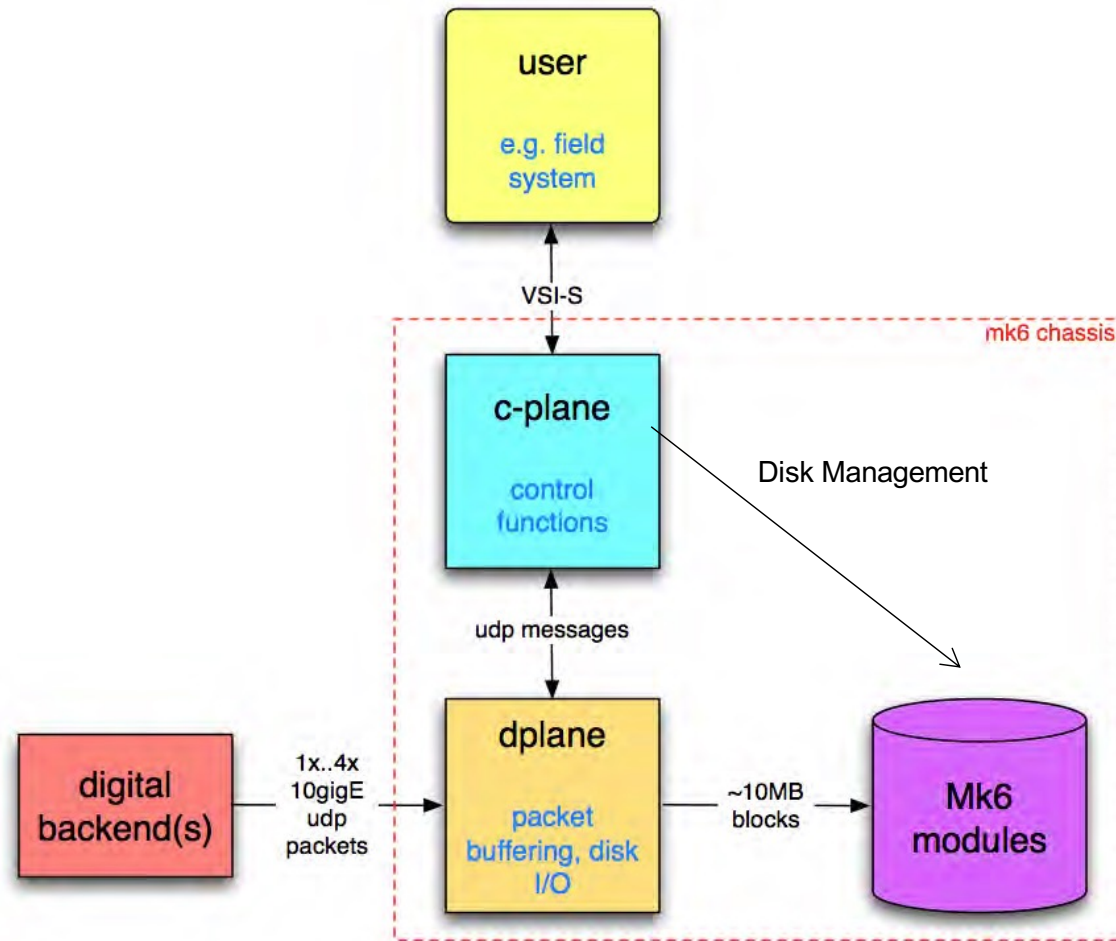
- Eth0 - Eth5 do not exist? What is happening with my system?
- OS disk was plugged in with different NIC cards
  - Linux assigned them eth0 - eth5
  - The new interfaces are eth6- eth11

### • How do I correct this?

- On CentOS7 systems hardcode MAC address in individual ifcfg-eth2-5
- On old Debian machines as root remove the following file and reboot
  - `rm /etc/udev/rules.d/70-persistent-net.rules`
  - This file will be automatically regenerated on bootup



# Mark6 Software Architecture

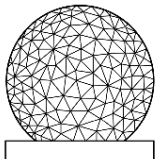


## Mark6 Applications

- cplane (control plane) application
  - 1.0.26(geo - Debian) / 2.1.1-0 (CentOS7)
- dplane (data plane) application
  - 1.22 (geo) / 1.22 (astro)
- End Stations
  - Need **both** applications / services to be running
- Correlators
  - Need only **c-plane** application / service is running

## Mark6 Applications (cont)

- cplane / dplane started as services on bootup
- CentOS7
  - `sudo sysconfig cplane {status, start, stop}`
  - `sudo sysconfig dplane {status, start, stop}`
  - To disable:
    - `sudo sysconfig disable cplane/dplane`
- Debian
  - `sudo /etc/init.d/cplane {restart , stop, start }`
  - `sudo /etc/init.d/dplane {restart , stop, start }`
- Configuration file
  - `/etc/default/mark6` (Next slide)
    - Sets the Interrupts / smp affinity / CPU Cores
    - Critical for performance (recording)



# Configuration File

# This file is sourced by /bin/sh from /etc/init.d/dplane

***Defined in file /etc/default/mark6***

# Options to pass to mark6 which take effect with restart.

# This specifies the ethernet ports to be used for incoming traffic.

# (Up to 4 ports are supported; You **must** list only the ones actually to be used.)

**MK6\_OPTS=eth2:eth3:eth4:eth5**

MK6\_DRVR=myri10ge

# Specifies the running directory--both planes log by default there.

**MK6\_RDIR=/var/log/mark6**

# dplane log level

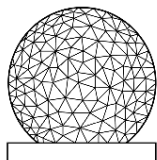
**MK6\_DLOG=2**

# cdplane log level (Information, level 0 is debug)

**MK6\_CLOG=1**

# process umask

MK6\_MASK=0002

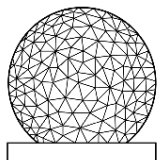


## Mark6 Application (cont)

- Where are the log files?
  - /var/log/mark6
  - dplane-daemon log
  - cplane-daemon log
  - M6-2015-DOY-HH-MM-SS.log
- For CentOS7
  - dplane-daemon log us used
  - cplane-daemon has no information
    - Moved to the journal files systems of sysconfig service

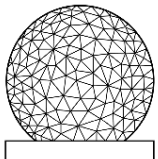
# Disk Modules

- Configured as RAID0 or scatter gather
  - Recommend using scatter gather mode for recording
- How to initialize a new module
  - `mod_init = slot : number disks : MSN : sg : new`
- How to remove a module from a group
  - `mod_init = slot : number disks : MSN : sg : null`
- How to erase
  - `group = unprotect : slot`
  - `group = erase : slot`
  - or `mod_init` the module:
    - `group=unmount:<slot>`
    - `mod_init = slot : 8 : MSN: sg : new`



## Disk Modules (cont)

- Insert module in slot
- Connect cables
- Power -Turn key
  - Takes about 25 secs for module to be recognized by Linux kernel
    - Watch lights on module
  - Wait before querying on the module status
    - `mstat ? all`
    - `mstat ? slot`
- Requires 8 disks in module
  - cplane will not be happy with less
  - Note some say this is a bug, we say require good modules
    - Revisiting philosophy based on 2 years of operation





## Disk Modules (cont)

- Removing disks
  - group = close : slot
  - group = unmount : slot
    - Can verify using linux command df to see if modules are truly unmounted
  - turn key to remove power
  - query the module status
    - mstat ? all
    - mstat ? slot
  - Bug if you mstat? before turning off power
    - The meta data of disk 0 will be remounted

# Recording

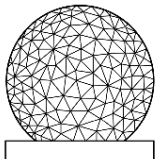
- Setup
  - `input_stream` command (next slide)
- Recording assumptions
  - Time is inspected in every header for all input streams defined
  - Only interfaces that are expecting data to be recorded should be defined
    - If a interface is defined and no data dplane will not close the files for it is expecting “ALL” streams specified to have valid data.
    - `record=off` must be issued to close files

# Recording

- Problems encountered
  - Data is not being recorded
    - input\_streams declarations do not match data on wire
      - Use wireshark to capture a few packets and make sure
        - packet length and offsets are correct
    - vdif headers do not have proper time
      - dplane uses vdif time to determine how much data to record based on record command
    - vdif packets received have different reference epochs
      - dplane expects all streams to transmit the same reference epochs.

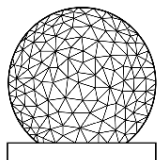
# Recording

- Data is not being recorded (cont)
  - No data is being received on the interfaces
    - `/sbin/ifconfig | grep -i "rx packets"`
      - to see if the receive packet counters are incrementing
  - A group is not open for recording
- Why does cplane commands return two status fields?
  - The first is the vsi-s return code
  - The second is a cplane specific return code
    - Specified in command set
    - (see next slide)



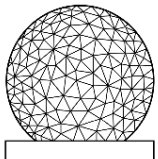
# cplane return codes

| Mk6 return code | Command      | Description   |
|-----------------|--------------|---|
| 2               |              | Specified group not open  |
| 10-19           | delete       |   |
| 20              | execute      | Invalid Action  |
| 21              | execute      | No filename provided  |
| 22              | execute      | Inconsistent filename used for append/finish process  |
| 23              | execute      | Duplicate filename  |
| 24              | execute      | Invalid upload sequence   |
| 25              | execute      | Attempted removal of non-existent xml file  |
| 30              | group        | Attempted open of multiple groups   |
| 31              | group        | Attempted open of incomplete group  |
| 32              | group        | 'unprotect' not issued immediately before 'erase'   |
| 33              | group        | 'auto' option failed, only supports module types initialized as scatter / gather and not RAID |
| 34              | group        | Attempted group open does not match subgroup defined in 'input_stream' configuration          |
| 40-49           | gsm          |   |
| 50-59           | gsm_mask     |   |
| 60              | input_stream | Invalid subgroup declaration (group already open)   |
| 61              | input_stream | Writing of subgroup meta data to disc failed  |
| 62              | input_stream | Adding stream label failed, it already exists   |
| 63              | input_stream | Specified stream label cannot be deleted it was not configured                                |
| 64              | input_stream | Committing configuration to dplane failed, not in an  |
| 65              | input_stream | Commit failed, invalid sub-grouping compared to the open group_ref                            |



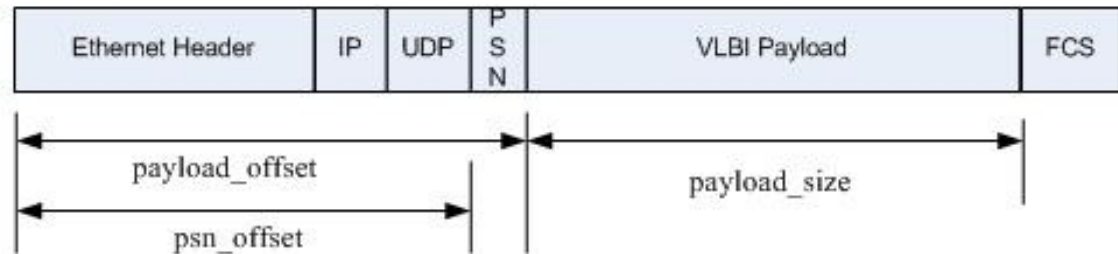
## Recording (cont)

- Our data does not have PSN's how do I turn off checking?
  - set `psn_offset` to 0, this disables checking
- How can I check what `vdif` time is being received by `dplane`
  - use `dpstat` utility
  - turn on debug level logging on `cplane` and look at the log files
- Can you abort a recording?
  - Yes, `record=off`
  - Will close any open files



## Mark6 Data Payload Definition and Parsing

Received by 10G Ethernet NIC



The “input\_stream” command from the Mark6 command set specifies how to treat the incoming data on a specific Ethernet interface:

```
input_stream = <action> : <stream_label> : <data_format> : <payload_size> : <payload_offset> : <psn_offset> :  
[<interface_ID>]: [ <filter address> ] : [<port>] : [<sub group ref>];
```

acton – {add, delete, commit}

delete with no stream label removes all labels defined

data\_format – “m5b” for mark5B, and “vdif” for vdif VLBI payload format.

payload\_size – VLBI Data Frame length in bytes, the length **must** be divisible by 8

payload\_offset – number of bytes into the received packet to find the start of the VLBI Data Frame.

psn\_offset – number of bytes into the received packet to find the start of the packet serial number

“0” represents no PSN in the incoming stream

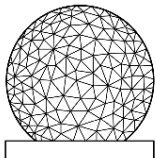
“non-zero value” represents the location of the PSN in the incoming stream

NOTE: Since the PSN can be the first word in the VLBI Data Frame or embedded in a VLBI header (e.g. word 5 of the vdif header) specifies the number of bytes to locate the PSN.

Interface\_id – {eth2, eth3, eth4, eth4, eth5}

Filter address and port not used

Sub group ref - sub-group (of open group) to which this data stream “interface\_ID” should be written to.



**MIT**  
**HAYSTACK**  
**OBSERVATORY**

# Play Back / Prepping for e-transfer

- Mount the disks
- group\_members? slot
  - Number of disks in the group\_ref
  - The associated disks eMSN in the group\_ref
- When mounting, does order have to be preserved?
  - No you can place them in any slot of the Mark6's
- gator – Wrapper program for gather464 and gather416.
- What does gathering the data do?
  - Takes the 4 thread IDs from the DBEs that are scattered gathered over the disk module and writes to a single file of either 64 channels in a single threadID, or 4 threads of 16 channels in a file.
  - This task is completed at the correlator when you send a S/G module.



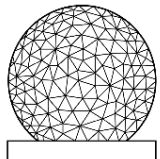
# Play Back / e-transfer

- Why gator if e-transfer
  - multi-thread was not originally supported
  - required 4 passes on the correlator (in-efficient)
  - Even today there is a performance gain if data gathered before playback vs multi-thread
- Problems with gator seen
  - Starts the gather and just stops but in 464 mode (with -t option)
  - Duration of gather

## Play Back / e-transfer

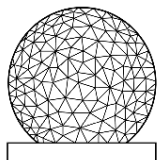
- Vdifuse

- Scatter / Gather Fuse interface for VDIF
  - process the data directly from the disk modules to DiFX
- Version specific for e-transfer under development
- Gathering of data no longer required



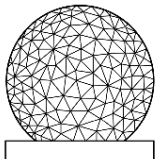
# RAID'd modules

- CentOS7 notes:
  - Automatically assembles a RAID'd module if keyed on.
  - `cat /proc/mdstat`
    - Will provide you the device assembled to.
  - If you receive a RAID you can convert it to s/g and the steps are:
    - `sudo mdadm --stop /dev/mdXXX`
    - This is not automated yet due to differences in how OS treats RAID's.
    - `mod_init`
- Debian:
  - is not an automatic process and treat as a standard module.



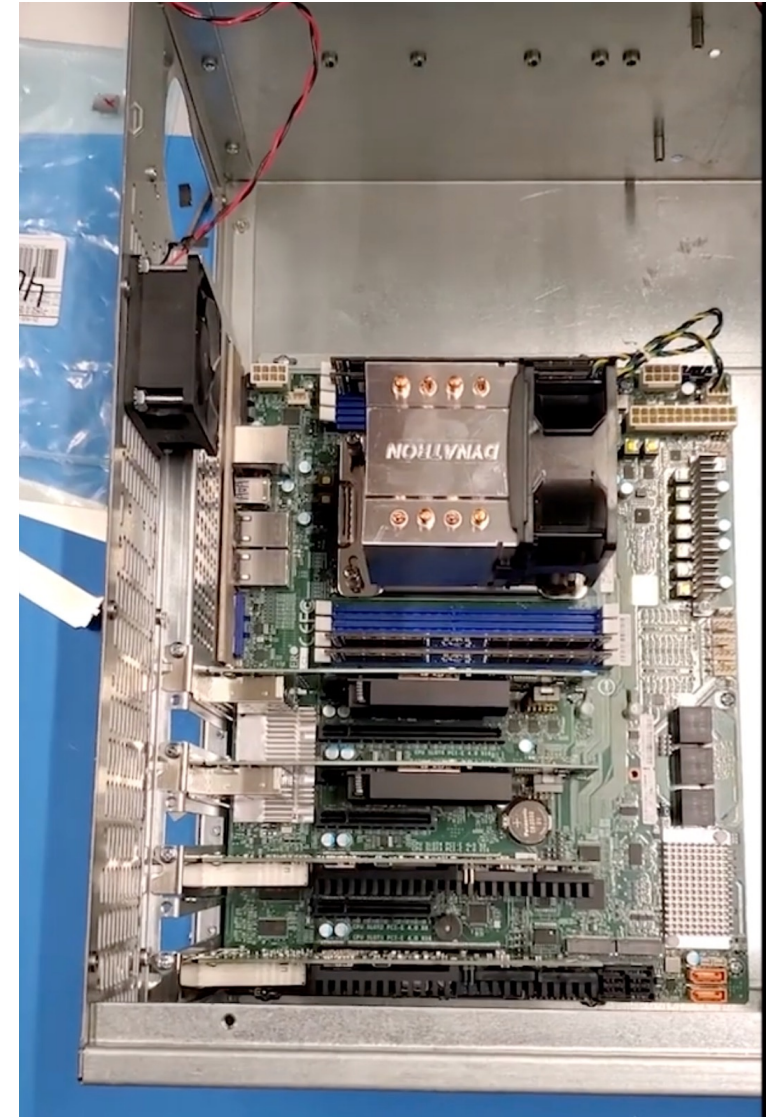
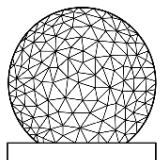
# Mark6 Next Steps

- Gatherize
  - gather464, gather416 replacement
  - Complex or real treated the same
  - supports 4-128 and even 4-256 (1GHz 32 channels / pol)
- OS Upgrade path
  - NASA is requiring us to move to a new OS
  - CentOS 7 Support thru 2024 but has fallen out of favor with the US federal government
  - Ubuntu FIPs LTS release (paid distro) is recommended
  - We will move to Ubuntu LTS 22.04 as the target
- Impact
  - cplane / dplane update required and utilities



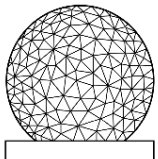
# Mark6+ - 32 Gbps recorder

- Hardware - same form factor:
  - AMD 16-core CPU
    - (EPYC) 128 PCIe 4.0 lane
  - PCIe capable 4.0 motherboard
    - x2 on-board NVMe slots.
    - 64 GB RAM (128GB possible)
  - x2 NIC Intel XXV710-DA2
    - Interface SFP+ 1/10/25g
    - PCIe 3.0 x8
  - x2 HBA: Atto 12Gb/s
    - Interface SFF-8644 SAS3
    - PCIe 4.0 x8
    - Backward compatible with SAS2 (modules are SFF-8088)
  - Additional cooling fans/deflectors
  - Total cost \$6k (2022) (excluding chassis/case and media).



# Questions / problems to discuss?

**May the Forth be with you!**



**MIT**  
**HAYSTACK**  
**OBSERVATORY**