From bsahr Tue Apr 17 12:09:58 2001
To: mrupen
Subject: correlator interfaces
Content-Type: X-sun-attachment
X-Lines: 2222
Status: RO
Content-Length: 134273

----- Begin Included Message -----

From bsahr Mon Apr 16 18:28:10 2001
To: Brent.Carlson@hia.nrc.ca, bsahr, ksowinski, bwaters, jbenson, bclark, jjackson, ghunt, rperley
Subject: correlator interfaces
Cc: tcornwell, cjanes, wkoski
Content-Type: X-sun-attachment
X-Lines: 2204
Status: RO
Content-Length: 133925

Brent & I had two email exchanges and one phone conversation last week on the subject of correlator interfaces. I will summarize the phone conversation in this note, and include the emails as attachments. The attachment entitled bkend02.pdf should be printed in color. If, like me, you seldom have the need to do so, and if you print it before CUPS comes online, then the printer to use is pstek560lp.

The attachments bkend02.txt & bkend02.pdf are Brent's initial text and diagram for 1) a correlator backend interface, and 2) correlator monitor and control. Briefly, Brent is suggesting the use of FPDP to get the data off the baseline boards, and the use of multiple Beowulf clusters for monitor and control and to move the data out of the correlator. Station boards are not shown in the diagram. Both they and the baseline boards would be configured via the monitor and control system. I'm not entirely clear on how needed info that is not usually considered to be monitor data would be obtained from the station boards. Brent does briefly mention this issue in bkend02.txt, but I am unsure if it would be packaged as monitor data, or if an additional 1 or more Beowulf clusters are involved.

For monitor and control of the baseline boards - 1 cluster consisting of 16 slaves & 1 master. For output from the baseline boards - 4 clusters, each consisting of 16 slaves & 1 master. Total, not including any additional slaves or masters for the station boards - 85 nodes. Each node would be a computer of some sort, probably a PC.

This cluster of clusters would be a deliverable. It would be treated as part of the correlator. Just who would program it to provide the needed functionality is unclear to me, but Brent does seem to imply that DRAO would supply it to NRAO with some portion of the software already completed.

From the phone conversation -

The requirements driving the baseline board output configuration are 1) the requested recirculation capability, and 2) the requested fast pulsar phase binning capability. For recirculation as speced (or as requested), a 1 ms readout time is needed for each correlator chip. This requirement has driven the design in the direction of placing an LTA controller (readout controller) behind each correlator chip. For phase binning, the requested capabilities require 64 readout controllers, with an arbitrator, per baseline board. The desired hot swap capabilities also add to the problem and drive the design in the direction of more interfaces per baseline board.

FPDP is attractive for output from the baseline boards because it is both fast enough to handle the data rates, and simple enough to be handled entirely by an FPGA. The required I/O rates do not permit of processor intervention at the level of the baseline boards.

Brent had much more to say on these matters, but to offer his comments at length from my notes and memory would guarantee inaccuracies. At some point a phone or video conference would be appropriate. DRAO now has a compatible video conferencing system.

Bill

----- End Included Message -----

Email of 4/13/2001, from Brent Carlson

Hi Bill:

I looked at your write-up and it looks correct to me. I'm not a Beowulf/Linux administrator so I only see our cluster as an application programmer might. To me, the Beowulf cluster (in our case) consists of 1 master and 15 slaves. The master has the name "master" and the slaves have the names "slave1" through "slave15". All of them are running Linux—which to me the programmer seems exactly like Unix. The master and the slaves are all connected by 100 Mbps Ethernet to a switch. The master has two network cards: one connects to the switch, and one connects to the outside world "DRAO net". Once you "rlogin" to the master, then from there, you can "rlogin" to the slaves. The master and the slaves all see the same directory structure—which is the disk on the master. If you want a slave to just use its own disk (for number crunching, temporary storage) then the "/tmp" directory is used and so a disk access will not occur over the network.

So, with that simple model (and not concerning myself with IP details which I let someone else worry about), I developed a network diagram of the "baseline subsystem" which I've attached as a .pdf file. I've shown this to Tony Willis who is developing the ACSIS software using the Beowulf system described above.

Some nomenclature that's used in the drawing:

MCC-M—monitor and control computer master.
MCC-S—monitor and control computer slave.
DHC-M—data handling computer master.
DHC-S—data handling computer slave.
...any other acronyms you should recognize.

In the drawing I represent the racks, boards, and computers in a quasi-physical fashion to give an idea of what some of the cable routing and racks actually look like. For computers (MCC, DHC), I used a desktop box since it's my favourite choice for this application (low cost, high performance, easy to replace). Because of PCI slot limitations, each DHC-S has data from only 4 Baseline Boards going into it. If a rack-mount computer is used, more PCI slots are available and so fewer computers can be used for lower performance at higher cost (!!!???)

In this drawing there are 5 Beowulf clusters. Each cluster (orange, red, green, magenta, blue) consists of 1 master and 16 slaves configured identically and in the manner described above. The orange cluster is for monitor and control and the master (MCC-M) provides the monitor and control interface to the outside world. The red, green, magenta, and blue clusters are for data handling. Each cluster sees a particular set of baselines across all sub-band correlators. Thus, the cluster size is always the same and performance depends on how many clusters you care to populate the correlator with.

Starting at the Baseline Boards the data flow is as follows:

1. Data leaves the Baseline Board via a dedicated P2P FPDP into a FPDP PCI card plugged in a given DHC-S. The DHC-Ss are given some configuration information from the MCC-M so they know how to tag the data with observation code, frequency ids etc.

2. The DHC-S performs the FFTs and creates FITS file fragments (UV tables) for the baselines it is processing. Normally, the DHC-Ss can operate without communication with other DHC-Ss in the same cluster. However, in those modes where a Baseline Board does only one part of a bigger lag chain, then the DHC-M will inform each slave where (i.e. which DHC-S) its data for particular baselines gets deposited for the FFT to occur. This way, it will be possible to share the FFT load amongst DHC-Ss. In this mode there is a performance hit because of the DHC-S-to-DHC-S communications over the network within the cluster. C'est la vie.

3. The DHC-S writes the FITS file fragments on to its "Network RAID Server" for the cluster. In my simple-minded thinking this would be done using NFS...but a more sophisticated message passing mechanism could be used. (Although, NFS may have some advantages when it comes to hot-swapping DHC-Ss???)

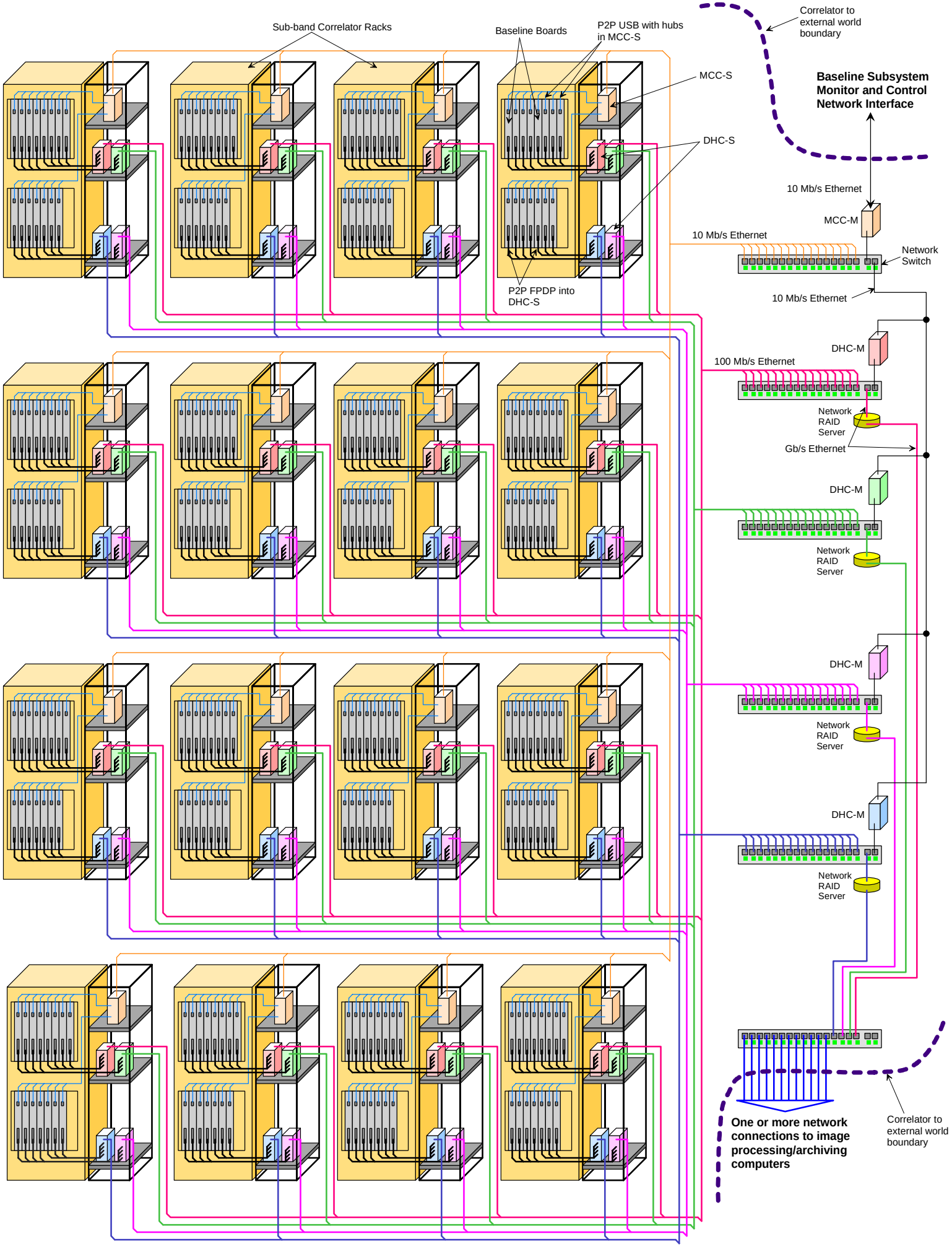
4. External image processing/archiving computers then go along and vacuum up the data from the Network RAID servers via a network switch to produce real-time images and assemble all of the FITS file fragments for writing out to permanent media. A similar operation would happen with the station subsystem file server so that calibration data (quantizer statistics, FIR filter powers, FIR filter bandshapes, noise diode measurements) can be put in the final amalgamated FITS files as well. If desired, raw data archiving (although its not completely raw at this point because the FFT has been done) could be done at the same time by the Network RAID Server -- or it could be done by the downstream image processing/archiving computers.

For monitor and control/configuration (of the baseline subsystem), the external world will only have to talk to a server on the MCC-M. The monitor and control at this point would be at a high level -- and potentially this will be the "virtual correlator interface" that we would provide. Similarly for the station subsystem and probably the phasing subsystem (although the phasing subsystem may be integrated with the baseline subsystem).

To upgrade performance, it is not necessary to make the Beowulf clusters bigger or add new software...simply add more identically configured Beowulf clusters...which adds Network RAID servers etc. The FITS file fragments will then contain fewer baselines but otherwise nothing has changed. If the software is designed correctly, adding more Beowulf clusters should be a simple field installation...likewise replacing MCCs and DHCs with newer computers.

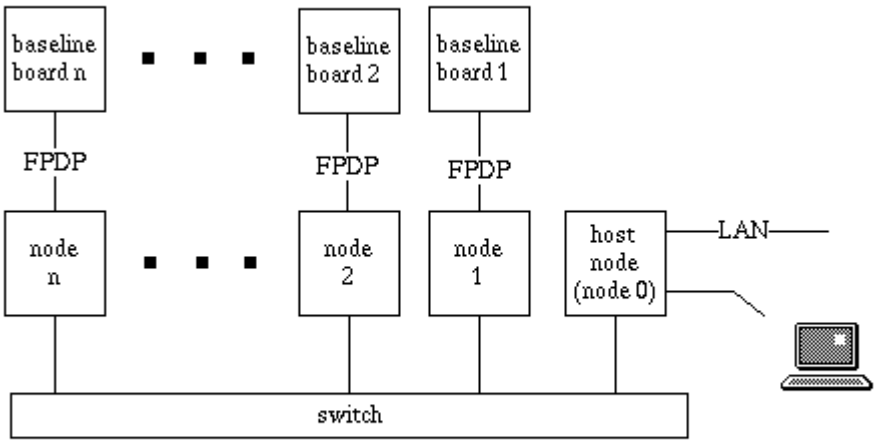
Anyway, this is preliminary... Comments? Questions?

Regards,
Brent.


Sub-band Correlator Racks
Baseline Boards
P2P USB with hubs in MCC-S
Correlator to external world boundary
MCC-S
Baseline Subsystem Monitor and Control Network Interface
DHC-S
10 Mb/s Ethernet
MCC-M
10 Mb/s Ethernet
Network Switch
P2P FPDP into DHC-S
10 Mb/s Ethernet
DHC-M
100 Mb/s Ethernet
Network RAID Server
Gb/s Ethernet
DHC-M
Network RAID Server
DHC-M
Network RAID Server
DHC-M
Network RAID Server
One or more network connections to image processing/archiving computers
Correlator to external world boundary

Brent,

A very modest beginning. It's not real to me until I see a diagram. This diagram is concerned with the correlator backend, and does not consider the station boards or monitor and control. I've spent all of an hour or so reading about Beowulf systems, and that constitutes the sum of my knowledge to date. So, I'll expose my ignorance …



The row along the bottom is the (Beowulf) cluster. I gather from my reading that all of the nodes other than the login node (node 0) use non-routable IP addresses. The line labeled "LAN" is the connection to the outside world. I'm assuming that the cluster will run some version of unix, i.e., some version of Linux. The cluster would be equipped with special software such as PVM or MPI for message passing, and, I suppose, would also include software for some sort of distributed memory system. Configuration would require someone with reasonably thorough knowledge of Linux. I suppose it is too much to ask that said person also have knowledge of Beowulf systems.

Of course, the nodes are connected to the switch via Ethernet and TCP/IP. 100 Mbps Ethernet? Gigabit Ethernet? Standard half duplex Ethernet or full duplex Ethernet (among the nodes)? The LAN would be whatever is used for the EVLA network inside the VLA control building. No decision on that yet. Of course, the most obvious possibility is standard Fast Ethernet, i.e. 100 Mbps, half duplex.

Other miscellaneous thoughts. An obvious one – placing more than one FPDP interface in each node. Again, as we discussed, using rack mounted, "server" style boards for the nodes.

Bill