MPIfR Correlator Report

Helge Rottmann

Bonn, 28th March 2011
Cluster Hardware

**Compute Nodes:**
60 Dual-Quadcores = 480 Cores (16 GB memory each)  
40 computers used for DiFX

**Datastream Nodes:**
14 Mark5 units (6 Mark5C, 1 Mark5B, 7 Mark5A)

**Headnodes**
1 Dual Quadcore (16 GB memory)  
+ 1 backup system

**Storage**
5 RAID-6 Units (20 TB each)  
5 RAID-6 Units (40 TB each)  
1 RAID-6 Unit (8 TB)  
5 RAID-6 Units (40 TB each)  
1 RAID-6 Unit (8 TB)  
**total ≈300TB**

80 TB reserved for DiFX  
(file-based correlation)

**Interconnect**
Infiniband connection (Cluster interconnect)  
2x 1GB Ethernet to Mark5 units
Mark5 Units

Total of 14 Mark5 units (mixed A/B/C flavors)

OS: CentOS 5.5 (= Redhat Enterprise Linux 5.5) pending upgrade...until playback bug is fixed (Roger ?)

SDK: 9.1

Interconnect: 2 x 1Gb Ethernet

NOTE: OpenMPI can utilize multiple networks automatically (no trunking needed)
• Low-Cost alternative for faster transfer speeds (compared to Infiniband etc.).
• No Problem with long cable lengths ( Infiniband: < 10m)

Planned additions for this year:

• Switchable PDU for Mark5 units (allows remote power off/on)
• Infiniband (over fibre) connection from Mark5s to cluster
Operations switched from using fuseMk5 to native mode in 12/2010:

=> higher playback speed (also more comfort!)

fuseMk5 (1x 1GB ethernet): 500-750 Mb/s
native mode (1x 1Gb ethernet): \( \approx 1.0 \) Gbit/s
native mode (2x 1Gb ethernet): \( \approx 1.4 \) Gbit/s
infiniband: ??? (Walter B. ?)

But: limited by maximum playback rate of the Mark5 Streamstor hardware:

PCI-816XF2 (Mark5A): 1.4 Gb/s
Amazon (Mark5C): 1.6 Gb/s
• currently no GUI/database in use (planned within 2011)
• using standard NRAO command-line applications

mkdaemon
mk5dir
mk5control
mk5erase
startdifx
genmachines
mk5mon
cpumon
....
Modifications to the standard NRAO programs
( mostly not yet submitted to the SVN repository 😞 )

**mk5daemon**
--user option
allows execution of system commands from mk5daemon
(e.g. mk5dir) with the given user privileges.
--isMk5
forces this machine to be a Mark5 regardless of its hostname

**genmachines**
properly treat file-based datastreams in creation of machine file

**startdifx**
run in loop until data is available (overcome „close“ problem)
Other planned modifications:

**startdifx / genmachines**
--use-machines
explictely give DIFX_MACHINES file, overiding „DIFX_MACHINES“ environment. Useful for starting multiple correlations at the same time.

**DIFX_MACHINES / genmachines**
add more options to DIFX_MACHINES file, e.g. for file-based correlation, allow certain machines to always serve a particular datastream path.

<table>
<thead>
<tr>
<th>Service</th>
<th>Priority</th>
<th>Process ID</th>
<th>Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>fxmanager</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>mark5fx01</td>
<td>0</td>
<td>1</td>
<td>serves=/data3</td>
</tr>
<tr>
<td>io03</td>
<td>6</td>
<td>0</td>
<td>serves=/data4</td>
</tr>
<tr>
<td>io04</td>
<td>5</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>
Mixed 32/64-bit operations:

Cluster is running 64-bit
Mark5 machines are running 32-bit (no 64-bit streamstor API by Conduant)

⇒ compile OpenMPI with option --enable-heterogeneous

Needs 32-bit and 64-bit versions of DiFX

MPIfR largely follows NRAO scheme:

See presentation „mixing 32 and 64-bit cluster members“
of Walter B. at:
Sample setup.sh

```bash
if [[ `uname --hardware-platform` == "x86_64" ]]; then
    export DIFXBITS=64
    export IPPROOT=/cluster/intel/ipp/6.1.2.051/em64t/
    export OPENMPIROOT=/cluster/openmpi/1.4.3/x86_64/
    export DIFXROOT=/cluster/difx/DiFX-trunk_64
else
    export DIFXBITS=32
    export IPPROOT=/cluster/intel/ipp/6.1.2.051/ia32
    export OPENMPIROOT=/cluster/openmpi/1.4.3/i386/
    export DIFXROOT=/cluster/difx/DiFX-trunk_centos55_sdk91_32
fi

export DIFX_VERSION=DiFX-trunk
export MPIXCC=${OPENMPIROOT}/bin/mpicxx
export PATH=${DIFXROOT}/bin:${OPENMPIROOT}/bin:${PATH}
export PKG_CONFIG_PATH=${DIFXROOT}/lib/pkgconfig:${PKG_CONFIG_PATH}
export LD_LIBRARY_PATH=${DIFXROOT}/lib:${IPPROOT}/sharedlib:${OPENMPIROOT}/lib:$
{LD_LIBRARY_PATH}
export DIFX_MESSAGE_GROUP=224.2.2.1
export DIFX_MESSAGE_PORT=52525
export DIFX_BINARY_GROUP=224.2.2.1
export DIFX_BINARY_PORT=52526
export CALC_SERVER=fxmanager
export DIFX_MACHINES=/cluster/difx/machines
export MARK5_DIR_PATH=/cluster/difx/directories
```
Main problems encountered during DiFX operations:

1) Hanging Mark5 units
   Mark5 computers frequently crash & freeze during correlation
   (typically once every few hours).
   => *unattended correlation impossible*
   Crashes occur more frequently in native mode than using fuseMk5
   (higher datarates? Larger reads?)

2) „Invalid data“ on kernels < 2.6.18
   First saw this problem in Jan 2011. Degugging and finding kernel-link was time-consuming. Thanks to Roger, problem is now fairly well understood!

3) Problems with data
   DiFX is less tolerant towards “weird” data than MK4 hardware correlator.
   Occasions were data could be correlated with MK4 but was refused by DiFX.
   Requires: Fixing the data by the analysts.

   EVN correlators probably see “weird” data more frequently than VLBA.