Cluster Computing and Preprocessing of the native format data into a universally recognized file format.

Archit Sakhadeo, Roll no 3258

Abstract

High-resolution astronomical imaging at sub-GHz radio frequencies has been available for more than 15 years, with the VLA at 74 and 330 MHz, and the GMRT at 150, 240, 330 and 610 MHz. Recent developments include wide-bandwidth upgrades for VLA and GMRT, and commissioning of the aperture-array-based, multibeam telescope LOFAR. A common feature of these telescopes is the necessity to deconvolve the very many detectable sources within their wide fields-of-view and beyond. This is complicated by gain variations in the radio signal path that depend on viewing direction.

Here, SPAM (Source Peeling and Atmospheric Modeling), a set of AIPS-based data reduction scripts in Python that includes direction-dependent and direction-independent calibration, and imaging, has been used. Since its first version in 2008, SPAM has been applied to many GMRT data sets at various frequencies.

We have automated the SPAM pipeline using a High Performance Computing Cluster, to convert the raw data obtained at GMRT, to FITS images, cycle wise. The primary stage of processing consists of conversion of the raw, native data to a rather popular archive and transport format. The importance and the need of cluster computing in the data processing pipeline is explained.

1 Introduction

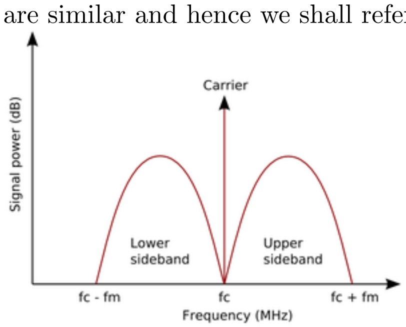
The Giant Metrewave Radio Telescope (GMRT), located near Pune in India, World's Largest Radio Telescope is an array of thirty fully steerable parabolic radio telescopes of 45 metre diameter, observing at metre wavelengths. It is operated by the National Centre for Radio Astrophysics, a part of the Tata Institute of Fundamental Research, Mumbai. At the time it was built, it was the world's largest interferometric array offering a baseline of up to 25 kilometres.

NCRA has set up a unique facility for radio astronomical research using the metre 110 wavelengths range of the radio spectrum, known as the Giant Metrewave Radio Telescope (GMRT), it is located at a site about 80 km north of Pune. GMRT consists of 30 fully steerable gigantic parabolic dishes of 45m diameter each spread over distances of upto 25 km.The metre wavelength part of the radio spectrum has been particularly chosen for study with GMRT because man-made radio interference is considerably lower in this part of the spectrum in India. Although there are many outstanding astrophysics problems which are best studied at metre wavelengths, there has, so far, been no large facility anywhere in the world to exploit this part of the spectrum for astrophysical research.

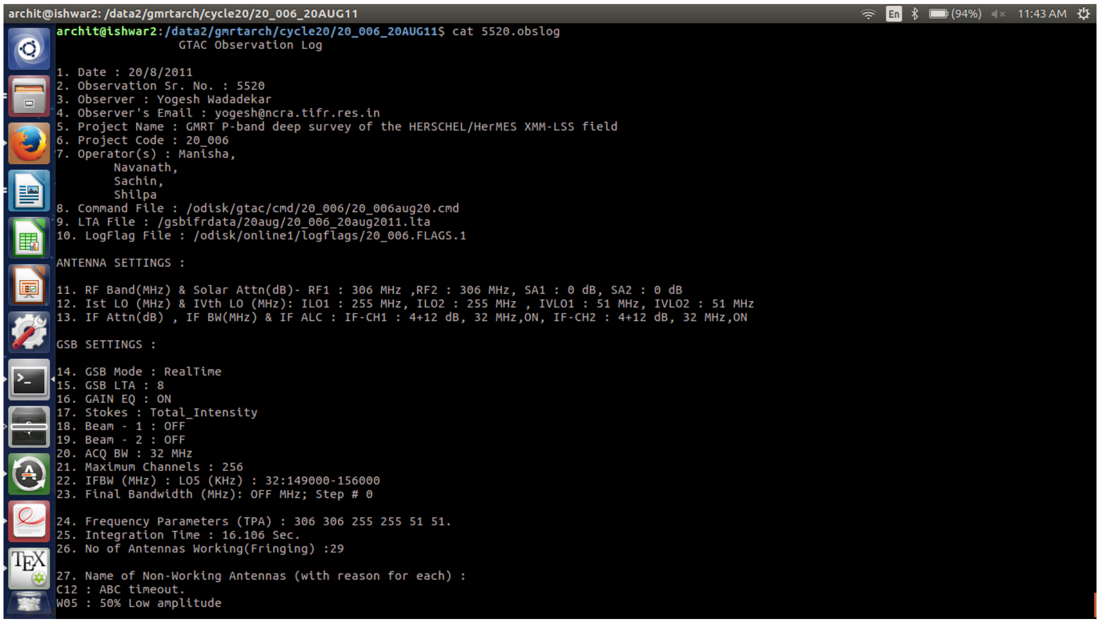
This report studies the extra-galactic radio data collected by GMRT telescope, how it is converted in a format accepted universally and explains the need for cluster computing and its usage in our pipeline.

2 Introduction to Radio Signals

Every antenna has a telescope at its end. There are 30 such antennas under the GMRT project which collect radio signals when pointed in a particular direction. These signals have Upper side band and Lower side band containing Fourier components of modulated signal. And with each band - USB and LSB, a different image is generated since both may not always be same. The image generated from USB is LTA and the one from LSB is LTB. These two images are similar and hence we shall refer to both of them as LTA.



3 Features of observation



4 LTA File Format

LTA is the native file format used at GMRT. It is simply the data collected by the antenna and stored in the most basic, raw form. It is a binary file with a peculiar header which denotes several fields like
1) The scan number
2) The object at which the telescope was pointed to
3) Observation mode
4) Baselines
5) Channels
6) Date and time of observation
7) Radio Frequency at which it is observed
8) Number of records
9) Coordinates of the object

VERSION: COR30x2 LTA1.10 DAS1.10 HST03

OBS_MODE: INDIAN_POLAR   CORR_HOST: GSB   BASELINES: 930   CHANNELS: 256

SCN	OBJECT	RA(MEAN)	DEC(MEAN)	DATE	IST	RF(MHz)	CW(kHz)	Nre
0	3C48	01h38m21.29	+33d13'07.24"	20/Aug/2011	00:05:22	306.00	130.208	61
1	0054-035	00h54m43.99	-03d30'08.80"	20/Aug/2011	00:23:57	306.00	130.208	17
2	XMMLSS01	02h24m50.27	-03d06'51.71"	20/Aug/2011	00:30:23	306.00	130.208	63
3	XMMLSS02	02h21m05.21	-03d31'49.48"	20/Aug/2011	00:48:09	306.00	130.208	64
4	XMMLSS03	02h17m20.18	-03d51'47.31"	20/Aug/2011	01:05:49	306.00	130.208	63
5	0054-035	00h54m43.99	-03d30'08.80"	20/Aug/2011	01:24:22	306.00	130.208	17
6	XMMLSS04	02h26m05.08	-04d16'52.46"	20/Aug/2011	01:30:43	306.00	130.208	62
7	XMMLSS05	02h24m05.06	-04d26'51.26"	20/Aug/2011	01:48:04	306.00	130.208	62
8	XMMLSS06	02h21m05.02	-04d46'49.48"	20/Aug/2011	02:05:37	306.00	130.208	62
9	0054-035	00h54m43.99	-03d30'08.80"	20/Aug/2011	02:24:02	306.00	130.208	15
10	XMMLSS07	02h18m35.00	-04d56'48.03"	20/Aug/2011	02:29:47	306.00	130.208	63
11	XMMLSS08	02h27m19.89	-05d21'53.22"	20/Aug/2011	02:47:19	306.00	130.208	63
12	XMMLSS09	02h23m34.86	-05d41'50.96"	20/Aug/2011	03:05:01	306.00	130.208	64
13	XMMLSS10	02h19m49.81	-06d06'48.75"	20/Aug/2011	03:22:54	306.00	130.208	64
14	0054-035	00h54m43.99	-03d30'08.80"	20/Aug/2011	03:42:15	306.00	130.208	17
15	XMMLSS01	02h24m50.27	-03d06'51.71"	20/Aug/2011	03:49:27	306.00	130.208	63
16	XMMLSS02	02h21m05.21	-03d31'49.48"	20/Aug/2011	04:07:16	306.00	130.208	62
17	XMMLSS03	02h17m20.18	-03d51'47.31"	20/Aug/2011	04:24:46	306.00	130.208	63
18	0054-035	00h54m43.99	-03d30'08.80"	20/Aug/2011	04:43:47	306.00	130.208	16
19	XMMLSS04	02h26m05.08	-04d16'52.46"	20/Aug/2011	04:50:31	306.00	130.208	62
20	XMMLSS05	02h24m05.06	-04d26'51.26"	20/Aug/2011	05:07:48	306.00	130.208	64
21	XMMLSS06	02h21m05.02	-04d46'49.48"	20/Aug/2011	05:25:47	306.00	130.208	64
22	0054-035	00h54m43.99	-03d30'08.80"	20/Aug/2011	05:44:43	306.00	130.208	17
23	XMMLSS07	02h18m35.00	-04d56'48.03"	20/Aug/2011	05:50:57	306.00	130.208	63
24	XMMLSS08	02h27m19.89	-05d21'53.22"	20/Aug/2011	06:08:40	306.00	130.208	63
25	XMMLSS09	02h23m34.86	-05d41'50.96"	20/Aug/2011	06:26:20	306.00	130.208	62
26	XMMLSS10	02h19m49.81	-06d06'48.75"	20/Aug/2011	06:43:39	306.00	130.208	64
27	0054-035	00h54m43.99	-03d30'08.80"	20/Aug/2011	07:02:29	306.00	130.208	17
28	XMMLSS01	02h24m50.27	-03d06'51.71"	20/Aug/2011	07:08:45	306.00	130.208	63
29	XMMLSS02	02h21m05.21	-03d31'49.48"	20/Aug/2011	07:26:17	306.00	130.208	62
30	XMMLSS03	02h17m20.18	-03d51'47.31"	20/Aug/2011	07:43:50	306.00	130.208	64
31	0054-035	00h54m43.99	-03d30'08.80"	20/Aug/2011	08:02:45	306.00	130.208	16
32	0521-207	05h22m08.85	-20d47'00.82"	20/Aug/2011	08:10:53	306.00	130.208	16
33	XMMLSS04	02h26m05.08	-04d16'52.46"	20/Aug/2011	08:18:07	306.00	130.208	63
34	XMMLSS05	02h24m05.06	-04d26'51.26"	20/Aug/2011	08:35:50	306.00	130.208	63
35	XMMLSS06	02h21m05.02	-04d46'49.48"	20/Aug/2011	08:53:45	306.00	130.208	63
36	0521-207	05h22m08.85	-20d47'00.82"	20/Aug/2011	09:13:04	306.00	130.208	16
37	XMMLSS07	02h18m35.00	-04d56'48.03"	20/Aug/2011	09:19:35	306.00	130.208	63
38	XMMLSS08	02h27m19.89	-05d21'53.22"	20/Aug/2011	09:37:21	306.00	130.208	40

5 UVFITS file format

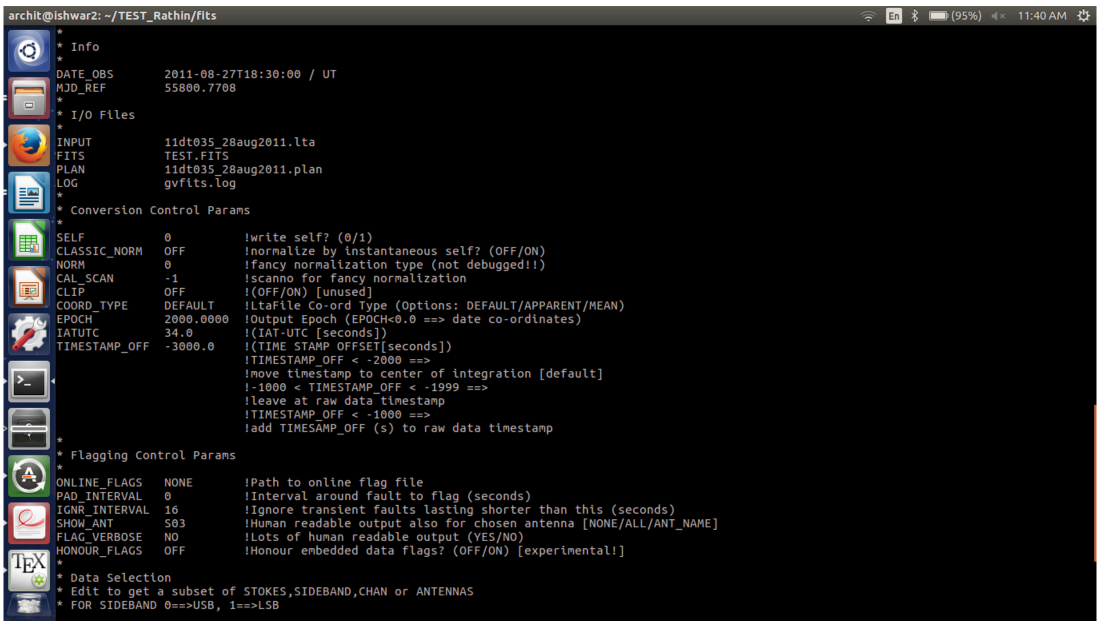
LTA files are not used globally as every observatory has their own file format. The image processing system that is universally used - AIPS, accepts input only in UVFITS format. UVFITS file also consists of similar parameters but in a different form.

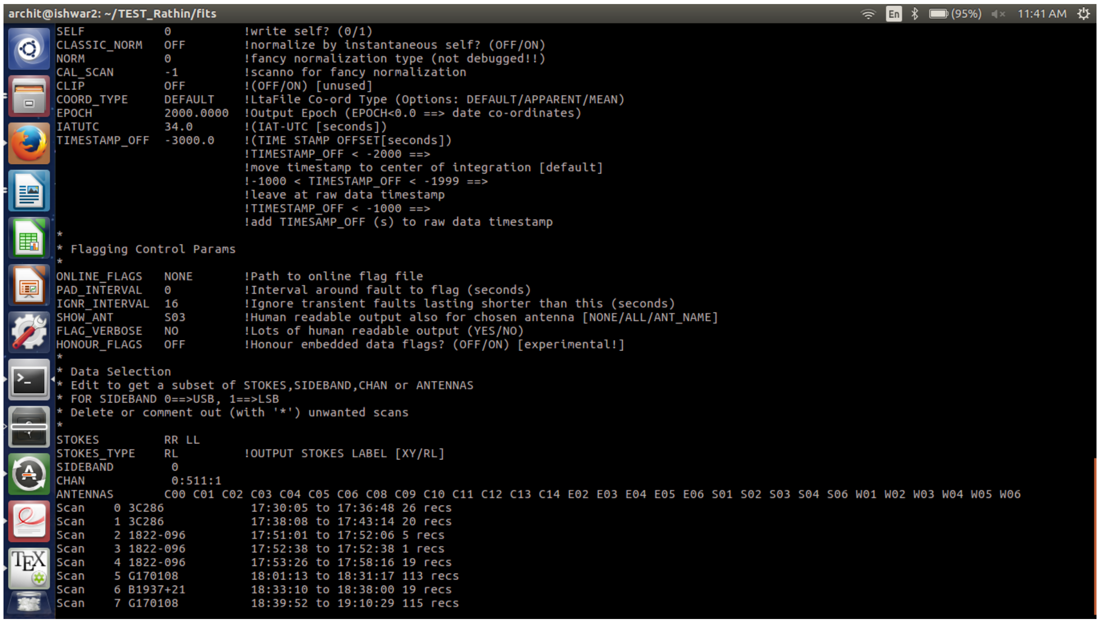
```
SIMPLE  =                    T / file does conform to FITS standard             BITPIX  =                  -32 / number of bits per data pixel
                 NAXIS   =                    7 / number of data axes                            NAXIS1  =                    0 / length of dat
a axis 1                 NAXIS2  =                    3 / length of data axis 2                            NAXIS3  =
2 / length of data axis 3                  NAXIS4  =                  256 / length of data axis 4                            NAXIS5  =
                 1 / length of data axis 5                  NAXIS6  =                    1 / length of data axis 6
     NAXIS7  =                    1 / length of data axis 7                            EXTEND  =                    T / FITS dataset may contain
extensions              GROUPS  =                    T / random group records are present              PCOUNT  =                    8 / number o
f random group parameters              GCOUNT  =               597690 /                                            COMMENT   FITS (Flexible
 Image Transport System) format is defined in 'AstronomyCOMMENT   and Astrophysics', volume 376, page 359; bibcode: 2001A&A...376..359H OBJECT
 = 'MULTI   '                                                                 TELESCOP= 'GMRT    '
          INSTRUME= 'GMRT    '                                                                 OBSERVER= '        '
                           HISTORY AIPS SORT ORDER = 'TB'                                                               HISTORY Created by GVFITS Version  1
.17                                        HISTORY GMRT CORR Version COR30x2 LTA1.10 DAS1.10 HST03                                        HISTORY GMRT Projec
t Code 20_004                                                   HISTORY Input LTA file 20_004_03jul2011.lta                                                        PR
OJECT = '        '                                                                 DATE-OBS= '2011-07-02'
               DATE-MAP= '2011-07-02'                                                                 BSCALE  =    1.00000000000E+00
                              BZERO   =    0.00000000000E+00                                                                 BUNIT   = 'UNCALIB '
                                            EPOCH   =       2.000000000E+03                                                                 ALTRPIX =
 1.000000000E+00                                                       CTYPE2  = 'COMPLEX '
   CRVAL2  =      1.0000000000E+00                                                       CDELT2  =       1.000000000E+00
                    CRPIX2  =       1.000000000E+00                                                       CROTA2  =      0.0000000000E+00
                                   CTYPE3  = 'STOKES  '                                                       CRVAL3  =     -1.0000000000
E+00                                                       CDELT3  =      -1.000000000E+00                                                       CRPIX3  =
      1.000000000E+00                                                       CROTA3  =      0.0000000000E+00
        CTYPE4  = 'FREQ    '                                                       CRVAL4  =      6.2900000000E+08
                         CDELT4  =      -1.302083281E+05                                                       CRPIX4  =       5.000000000E-01
                                         CROTA4  =      0.0000000000E+00                                                       CTYPE5  = 'IF      '
                                                       CRVAL5  =      1.0000000000E+00                                                       CDEL
T5  =       1.000000000E+00                                                       CRPIX5  =       1.000000000E+00
             CROTA5  =      0.0000000000E+00                                                       CTYPE6  = 'RA      '
                            CRVAL6  =      0.0000000000E+00                                                       CDELT6  =       1.000000000E+00
                                          CRPIX6  =       1.000000000E+00                                                       CROTA6  =      0.
0000000000E+00                                                       CTYPE7  = 'DEC     '
 CRVAL7  =      0.0000000000E+00                                                       CDELT7  =       1.000000000E+00
                  CRPIX7  =       1.000000000E+00                                                       CROTA7  =      0.0000000000E+00
                                 PTYPE1  = 'UU---SIN'                                                       PSCAL1  =    1.00000000000E+
00                                                       PZERO1  =    0.00000000000E+00                                                       PTYPE2  = '
VV---SIN'                                                       PSCAL2  =    1.00000000000E+00
     PZERO2  =    0.00000000000E+00                                                       PTYPE3  = 'WW---SIN'
                      PSCAL3  =    1.00000000000E+00                                                       PZERO3  =    0.00000000000E+00
                                    PTYPE4  = 'BASELINE'                                                       PSCAL4  =    1.00000000
000E+00                                                       PZERO4  =    0.00000000000E+00                                                       PTYPE5
```

6 Preprocessing initial step

NCRA has designed two specific commands which do the conversion from LTA to UVFITS format. They are
1) LISTSCAN

- listscan : List the scan in the ltafile for converting to UVFITS format. This will create a file with .log extension. The parameters you might need to change are the output UVFITS file name, CLASSIC_NORM (normalisation by self), removing unwanted antennas or scans, giving path to the FLAG file etc.





```
archit@ishwar2: ~/TEST_Rathin/fits
SELF            0           !write self? (0/1)
CLASSIC_NORM    OFF         !normalize by instantaneous self? (OFF/ON)
NORM            0           !fancy normalization type (not debugged!!)
CAL_SCAN        -1          !scanno for fancy normalization
CLIP            OFF         !(OFF/ON) [unused]
COORD_TYPE      DEFAULT     !LtaFile Co-ord Type (Options: DEFAULT/APPARENT/MEAN)
EPOCH           2000.0000   !Output Epoch (EPOCH<0.0 ==> date co-ordinates)
IATUTC          34.0        !(IAT-UTC [seconds])
TIMESTAMP_OFF   -3000.0     !(TIME STAMP OFFSET[seconds])
                            !TIMESTAMP_OFF < -2000 ==>
                            !move timestamp to center of integration [default]
                            !-1000 < TIMESTAMP_OFF < -1999 ==>
                            !leave at raw data timestamp
                            !TIMESTAMP_OFF < -1000 ==>
                            !add TIMESAMP_OFF (s) to raw data timestamp
*
* Flagging Control Params
*
ONLINE_FLAGS    NONE        !Path to online flag file
PAD_INTERVAL    0           !Interval around fault to flag (seconds)
IGNR_INTERVAL   16          !Ignore transient faults lasting shorter than this (seconds)
SHOW_ANT        S03         !Human readable output also for chosen antenna [NONE/ALL/ANT_NAME]
FLAG_VERBOSE    NO          !Lots of human readable output (YES/NO)
HONOUR_FLAGS    OFF         !Honour embedded data flags? (OFF/ON) [experimental!]
*
* Data Selection
* Edit to get a subset of STOKES,SIDEBAND,CHAN or ANTENNAS
* FOR SIDEBAND 0==>USB, 1==>LSB
* Delete or comment out (with '*') unwanted scans
*
STOKES          RR LL
STOKES_TYPE     RL          !OUTPUT STOKES LABEL [XY/RL]
SIDEBAND         0
CHAN             0:511:1
ANTENNAS        C00 C01 C02 C03 C04 C05 C06 C08 C09 C10 C11 C12 C13 C14 E02 E03 E04 E05 E06 S01 S02 S03 S04 S06 W01 W02 W03 W04 W05 W06
Scan    0 3C286              17:30:05 to 17:36:48 26 recs
Scan    1 3C286              17:38:08 to 17:43:14 20 recs
Scan    2 1822-096           17:51:01 to 17:52:06 5 recs
Scan    3 1822-096           17:52:38 to 17:52:38 1 recs
Scan    4 1822-096           17:53:26 to 17:58:16 19 recs
Scan    5 G170108            18:01:13 to 18:31:17 113 recs
Scan    6 B1937+21           18:33:10 to 18:38:00 19 recs
Scan    7 G170108            18:39:52 to 19:10:29 115 recs
```

2) GVFITS

- gvfits : Convert GMRT (lta) data to UVFITS format. Run gvfits on the above log file (eg: gvfits test.log). This will take some time depending on the size of the input file.

These two commands are embedded in the SPAM function convert_ lta_ to_ uvfits(LTA_ FILENAME,UVFITS_ FILENAME)

```
20_064_05sep2011.lta.UVFITS  l2u_precal.log  lta_to_uvfits_log.txt
[root@gadpu 20_064_05sep2011_lta]# cat l2u_precal.log
     ----- LISTSCAN version 1.17  -----
Please edit 20_064_05sep2011.log and then run
gvfits 20_064_05sep2011.log
 Scan  0 ::  43/ 43 recs
 Scan  1 ::  18/ 18 recs
 Scan  2 :: 115/115 recs
 Scan  3 ::  17/ 17 recs
 Scan  4 :: 116/116 recs
 Scan  5 ::  18/ 18 recs
 Scan  6 :: 116/116 recs
 Scan  7 ::  18/ 18 recs
 Scan  8 :: 116/116 recs
 Scan  9 ::  18/ 18 recs
 Scan 10 :: 116/116 recs
 Scan 11 ::  18/ 18 recs
 Scan 12 :: 116/116 recs
 Scan 13 ::  18/ 18 recs
 Scan 14 :: 116/116 recs
 Scan 15 ::  18/ 18 recs
 Scan 16 :: 115/115 recs
 Scan 17 ::  17/ 17 recs
 Scan 18 :: 116/116 recs
 Scan 19 ::  18/ 18 recs
 Scan 20 :: 116/116 recs
 Scan 21 ::  18/ 18 recs
 Scan 22 :: 117/117 recs
 Scan 23 ::  17/ 17 recs
 Scan 24 ::  18/ 18 recs
 Scan 25 :: 116/116 recs
 Scan 26 ::  18/ 18 recs
 Scan 27 ::  51/ 51 recs
 Scan 28 ::  41/ 41 recs
     ----- GVFITS  version 1.17  -----
FITSIO version number = 3.030000
Created 20_064_05SEP2011A.UVFITS
See gvfits.log for details
```

7 Total Data

GMRT has NCRA Archive and Proposal Management System (NAPS) to manage all the data. GMRT has been recording data since 2002. The total data collected is divided into multiple cycles.
Each cycle lasts for 6 months. Thus all the data collected by all these 30 telescopes within these 6 months is collectively called as 1 cycle.

Each cycle has a total of about 1-3 TB of data. And all cycles combined have 60 TB data.

Our task is to automate the processing of the pipeline on this 60 TB data for which we used 1 TB data from cycle 20 as our test set.

8 Processing time

The time taken to process one LTA file into one FITS file on an average is 16 hours. It can last from 11 hours to sometimes even 24 hours. One LTA file which is generally of the size of 1 GB takes 16 hours, then our guess being that all the data from 31 cycles till data should take 983040 hours i.e. 112 years if run on one core and one machine.

9 Need for more computing power

Since, this is not feasible at all, we make use of a cluster of computers. This is because, there is a need for more computing power as well as more RAM, more hard disk memory, swap memory and caches.

10 Cluster Computing

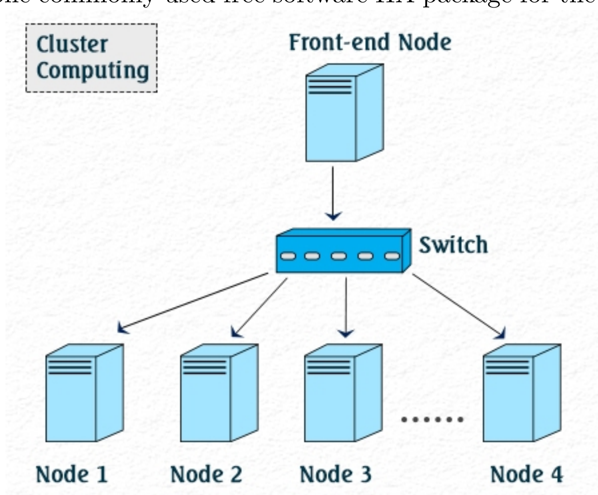
A computer cluster consists of a set of loosely or tightly connected computers that work together so that, in many respects, they can be viewed as a single system. Unlike grid computers, computer clusters have each node set to perform the same task, controlled and scheduled by software.

The components of a cluster are usually connected to each other through fast local area networks ("LAN"), with each node (computer used as a server) running its own instance of an operating system. In most circumstances, all of the nodes use the same hardware and the same operating system, although in some setups (i.e. using Open Source Cluster Application Resources (OSCAR)), different operating systems can be used on each computer, and/or different hardware.

They are usually deployed to improve performance and availability over that of a single computer, while typically being much more cost-effective than single computers of comparable speed or availability.

Computer clusters are used for computation-intensive purposes, rather than handling IO-oriented operations such as web service or databases. For instance, a computer cluster might support computational simulations of vehicle crashes or weather. Very tightly coupled computer clusters are designed for work that may approach "supercomputing".

"High-availability clusters" (also known as failover clusters, or HA clusters) improve the availability of the cluster approach. They operate by having redundant nodes, which are then used to provide service when system components fail. HA cluster implementations attempt to use redundancy of cluster components to eliminate single points of failure. There are commercial implementations of High-Availability clusters for many operating systems. The Linux-HA project is one commonly used free software HA package for the Linux operating system.



11 Types of Cluster Computing

A computer cluster may be a simple two-node system which just connects two personal computers, or may be a very fast supercomputer. A basic approach to building a cluster is that of a Beowulf cluster which may be built with a few personal computers to produce a cost-effective alternative to traditional high performance computing.

Computer clusters may be configured for different purposes ranging from general purpose business needs such as web-service support, to computation-intensive scientific calculations. In either case, the cluster may use a high-availability approach. Note that the attributes described below are not exclusive and a "computer cluster" may also use a high-availability approach, etc.

"Load-balancing" clusters are configurations in which cluster-nodes share computational workload to provide better overall performance. For example, a web server cluster may assign different queries to different nodes, so the overall response time will be optimized. However, approaches to load-balancing may significantly differ among applications, e.g. a high-performance cluster used for scientific computations would balance load with different algorithms from a web-server cluster which may just use a simple round-robin method by assigning each new request to a different node.

12 Architecture used at NCRA

There are always two options to choose from - a homogeneous cluster or a heterogeneous cluster. The problem with a heterogeneous cluster is to maintain each machine individually if they have different architecture or OS or any feature. It is significant trouble to care for each and every small feature in a heterogeneous cluster if there are thousands of nodes. Hence in huge clusters, it is preferred to use a homogeneous cluster.

The cluster at NCRA is a homogeneous one, where the master and the slave-compute nodes are similar in all ways. This serves multiple purposes:
1) Easy to identify which node went wrong
2) Easy to rest any node by simply rebooting the machine in case one machine fails
3) Maintenance of the cluster is easy
4) If the whole cluster seems faulty, then most of the times there's some minor fault which is repeated everywhere. Correcting that fault can help bring the cluster to the original form.
5) Easy to grow the cluster by simply attaching more similar nodes.

Following is the exact architecture of the cluster at NCRA

1. Frontend

    - Nodes of this type are exposed to the outside world. Many services (NFS, NIS, DHCP, NTP, MySQL, HTTP, ...) run on these nodes. In general, this requires a competent sysadmin. Frontend nodes are where users login in, submit jobs, compile code, etc. This node can also act as a router for other cluster nodes by using network address translation (NAT).

      Frontend nodes generally have the following characteristics:
    - Two ethernet interfaces - one public, one private.
    - Lots of disk to store files.

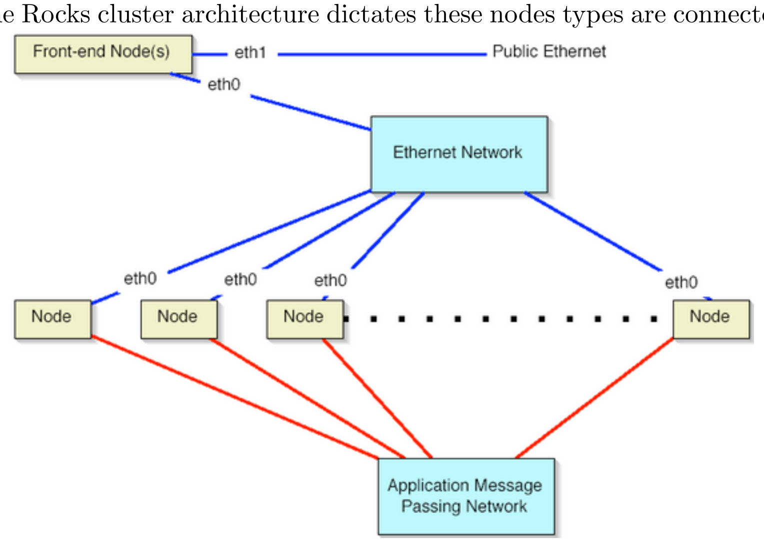
2. Compute

    - These are the workhorse nodes. They are also disposable. Our management scheme allows the complete OS to be reinstalled on every compute node in a short amount of time (10 minutes). These nodes are not seen on the public Internet.
    - Compute nodes have the following characteristics:
    - Power Cable

- Ethernet Connection for administration
- Disk drive for caching the base operating environment (OS and libararies)
- Optional high-performance network (e.g., Myrinet)

3. Ethernet Network
   - All compute nodes are connected with ethernet on the private network. This network is used for administration, monitoring, and basic file sharing.
4. Application Message Passing Network
   - All nodes can be connected with Gigabit-class networks and required switches. These are low-latency, high-bandwidth networks that enable high-performance message passing for parallel programs.

The Rocks cluster architecture dictates these nodes types are connected as such:



- On the compute nodes, the Ethernet interface that Linux maps to eth0 must be connected to the cluster's Ethernet switch. This network is considered private , that is, all traffic on this network is physically separated from the external public network (e.g., the internet).
- On the frontend, two ethernet interfaces are required. The interface that Linux maps to eth0 must be connected to the same ethernet network as the compute nodes. The interface that Linux maps to eth1 must be connected to the external network (e.g., the internet or your organization's intranet).
- Once you've physically assembled your cluster, each node needs to be set to boot without a keyboard. This procedure requires setting BIOS values and, unfortunately, is different for every motherboard. We've seen some machines where you cannot set them to boot without a keyboard.

13 Specifications of the architecture

RAM - 32 GB
Hard disk - 2 TB
SDD - 160 GB
OS - CentOS
Number of Cores - 8 (Octacore)
Processor - Intel i7
Cluster - Rocks Cluster Distribution

14 Speed up achieved

We achieved a speed up of about 56 times. By introducing a total of 100 compute nodes the speed up shall reach up to 600-700 times with optimum number of cores. Thus the work which would have required about 112 years, is now reduced to 1/600th the total time which is even less than two months for the processing of 60 TB of data.

15 Conclusion

Thus, using python wrappers on AIPS we used SPAM to get the native form data into a universally accepted format. Thus the first task in the preprocessing stage was completed giving way for the next task which is to split the sources and the calibrators and obtain separate UVFITS files. Also, a massive necessary speed up was obtained on the computation. Thus as per the need for the computation speed and resources, the required cluster was implemented. Having implemented it practically, it is certain that this cluster can be scaled for thousands of nodes with no problem since the installation is automated.

References

[1] Sirothia, S. K. 2009, MNRAS, 398, 853

[2] Sirothia, S. K., Lecavelier des Etangs, A., Gopal-Krishna, Kantharia, N. G., & Ishwar-Chandra, C. H.

[3] Sirothia, S. K., Saikia, D. J., Ishwara-Chandra, C. H., Kantharia, N. G. 2009, MNRAS, 392, 1403

[4] Smirnov, O. M. 2011, A&A, 527, A107

[5] http://www.astron.nl/citt/pybdsm/ Niruj Mohan and David Rafferty

[6] Arora, B. S., Morgan, J., Ord, S. M., et al. 2015, PASA, 32, e029

[7] Baars, J. W. M., Genzel, R., Pauliny-Toth, I. I. K., & Witzel, A. 1977

[8] Bagchi, J., Sirothia, S. K., Werner, N., et al. 2011, ApJ, 736, L8

[9] Baldwin, J. E., Boysen, R. C., Hales, S. E. G., et al. 1985, MNRAS, 217, 717

[10] Becker, R. H., White, R. L., & Helfand, D. J. 1995, ApJ, 450, 559

[11] Bilous, A., Kondratiev, V., Kramer, M., et al. 2015, ArXiv e-prints [arXiv:1511.01767]

[12] Bridle, A. H. Schwab, F. R. 1999, in Astronomical Society of the Pacific Conference Series, Vol. 180, Synthesis Imaging in Radio Astronomy II, ed. G. B.

[13] Taylor, C. L. Carilli, R. A. Perley, 371

[14] Brienza, M., Godfrey, L., Morganti, R., et al. 2016, A&A, 585, A29

[15] Cohen, A. S., Lane, W. M., Cotton, W. D., et al. 2007, AJ, 134, 1245

[16] Condon, J. J. 1997, PASP, 109, 166 II, ed. G. B. Taylor, C. L. Carilli, & R. A. Perley, 357

[17] de Gasperin, F., Intema, H. T., van Weeren, R. J., et al. 2015, MNRAS, 453, 3483

[18] Feretti, L., Giovannini, G., Govoni, F., & Murgia, M. 2012, A&A Rev., 20, 54

[19] Franzen, T. M. O., Jackson, C. A., Offringa, A. R., et al. 2016, MNRAS, 459

[20] Garrett, M. A. 2013, ArXiv e-prints [arXiv:1307.0386]

[21] George, S. J. & Stevens, I. R. 2008, MNRAS, 390, 741

[22] Ghosh, A., Prasad, J., Bharadwaj, S., Ali, S. S., Chengalur, J. N. 2012, MNRAS,

[23] Gopal-Krishna, Mhaskey, M., Wiita, P. J., et al. 2012, MNRAS, 423, 1053

[24] Greisen, E. W. 2003, Information Handling in Astronomy - Historical Vistas, 285, 109

[25] Hales, S. E. G., Baldwin, J. E., Warner, P. J. 1988, MNRAS, 234, 919

[26] Hales, S. E. G., Riley, J. M., Waldram, E. M., Warner, P. J., & Baldwin, J. E