Mem. S.A.It. Vol. 83, 377

Memorie della

An high availability data storage subsystem for the INTEGRAL data analysis

B.L. Martino[1] and M. Federici[2]

[1] Istituto di Analisi dei Sistemi ed Informatica "Antonio Ruberti", IASI-CNR Viale Manzoni 30, 00185 Roma , Italy e-mail: brunolmartino@gmail.com
[2] Istituto Nazionale di Astrofisica – IASF-ROMA, Via Fosso del cavaliere 100, 00133 Roma, Italy

Abstract. The article shows how the Cluster Aves (Federici et al. 2009) manages the data provided by the satellite INTEGRAL (Winkler et al. 2003). The Cluster is hosted at IASF/INAF center in Rome. In particular it describes the adopted strategies to optimize the management of a data-packadge synced with the ISDC central data storage in Geneva. This data storage collect the information sent by the INTEGRAL satellite. Today the data collection reach 10 terabytes and it will grow up to 15 terabytes at the end of the satellite mission in 2014. The files contained in the data package have an order of magnitude of 10^8. These two parameter reflect the scale of the complexity of the data analysis procedures. The strategy shown above has its silver bullets in the optimization of the control quality of the data-package files and in a database that store the data-package file status. Before optimization, the data checking was very time consuming, it was closely similar to the computational time of the analysis

Key words. AVES: Data Storage – INTEGRAL

1. Introduction

The astronomic satellite INTEGRAL (INTErnational Gamma-Ray Astrophysics Laboratory) developed to make observation of sky windows in the gamma ray field. It was selected by scientific committee of the European Space Agency (ESA) in June 1993 as medium size mission. It was launched on the 17th of October 2002 and designed to have a nominal lifetime of five years. Because its on board instruments were in 2009 still perfectly working, the ESA extended the mission until 2014.

Send offprint requests to: BL. Martino

The satellite is equipped with two main instruments: a spectrometer (SPI) (Vedrenne et al. 2003) and a gamma telescope (IBIS) (Ubertini et al. 2003).

These two instruments work in the same time complementary. The data catched by the telescope and by the spectrometer are integrated by other two kind of equipment: a X-ray detector (JEMX) (Budtz-Jrgensen et al. 2004) and an optical telescope (OMC) Mass-Hesse et al. (2003).

INTEGRAL provide an huge quantity of scientific data. A forecast for its whole providing data for the entire mission has an amount of about 15 TB to store and analyze.

2. The INTEGRAL data analysis

The INTEGRAL data analysis it done using the software package (OSA) (Goldwurm et al. 2003), developed and made available by ISDC (INTEGRAL Science Data Centre at Geneva University). OSA is composed by a compound of complex programs then research institutes and agencies that analyze the data have been provided with increasingly large and powerful computers to process them. In this complex reality is born inside the Rome IASF headquarter the AVES project. This project aim to obtain the highest possible computational performance by OSA through the development of the following points:

- HPC (high performance computing): development of a cluster dedicated to scientific computing (AVES).
- Smart management of slices of data to analyze: replication in the local memory of the nodes of the cluster involved in the calculation of the needed data.
- Smart management of satellite data: optimization of the data storage subsystem.

2.1. The AVES cluster system

AVES is a computer-based on a "cluster" architecture, i.e. a set of computers connected together via a fast local area network. The hardware part of AVES is low-cost and easily expandable, composed by 34 commercial computer (for a total of 120 processors, 120 Gbytes of RAM and 7.5 Tera bytes of shared memory) housed on a metal structure. The software that manages the cluster is SLURM (Simple Linux Utility for Resource Management) Yoo et al. (2003). SLURM is an open source resource manager which can handle up to 65536 nodes. Among others, SLURM was developed by Lawrence Livermore National Laboratory LLNL. Because of its size and its complexity OSA package is not easily modifiable to run in parallel computing mode. To allow OSA to operate in a parallel like mode on the 34 AVES computer, have been developed a set of 50 original programs based on Linux bash shell. The Fig. 2 shows the current computer system configuration.

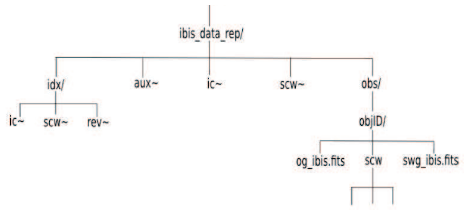

Fig. 1. The IBIS Data structure.

2.2. User data management (data slice management)

To minimize the time for retrieval of user data to be processed, only the information needed are transferred in local area memory. This mode of operation requires the transfer of data to be processed from user memory to a temporary area located within the computing nodes and, at the end of the calculation, back to user memory inside the local area network. In this way, the analysis software accesses user files that are in their local disk. Calculating the processing time as the sum of the time needed to the transfer file plus the CPU's time to process the data we obtain, with this approach, a total computing time that is faster in the worst case of two times, in the best case, up to four time.

3. Scientific data management

The scientific data are stored on two different storage units, configured in RAID-6, each of capacity of 48 TB. The data analysis programs constituting OSA requires to work properly a rigid folder structure as shown in Fig. 1. The main repository at the ISDC Geneva contains three current versions of the preprocessed data. Each version consists of four main folders: IDX, AUX, IC, SCW. The total number of files in each version is in the order of 10^8. The storage subsystem and AVES is composed by four servers shown in Fig. 3:

- A firewall to protect the connection with the ISDC.
- A primary storage server.
- A secondary storage server dedicated to cluster data analysis.

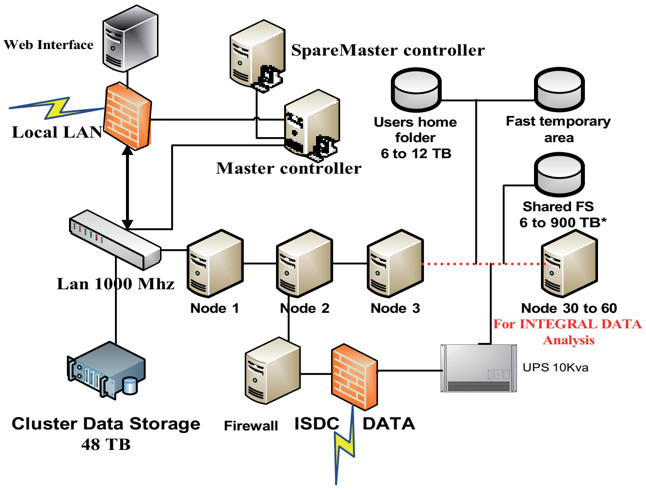

Fig. 2. The block diagram of the hardware structure of AVES. For this version: 480 TB of UFS capability for 30 nodes configuration.

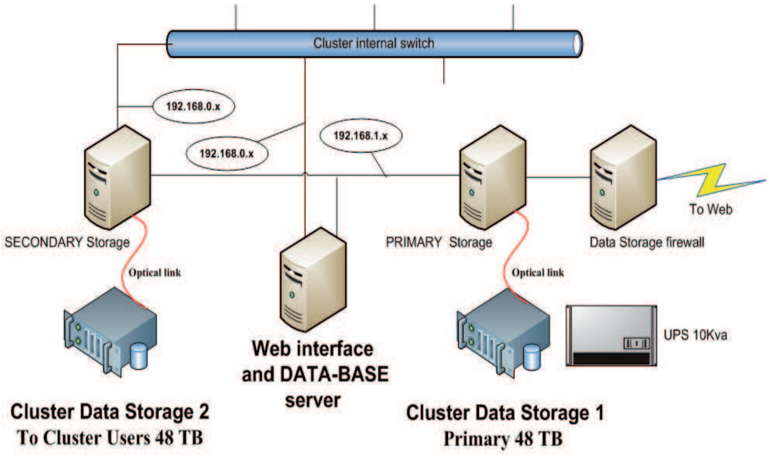

Fig. 3. Block diagram of the hardware structure of Data Storage system for AVES cluster.

– A server equipped with an user interface web based dedicated to the analysis of clustered data.

The tasks assigned to the subsystem are the following:

– Automatic download of the latest data from the site ISDC
– Correctness check of downloaded data
– Synchronization between the primary storage server and one dedicated to the cluster for scientific analysis
– Storing information in a database of state of the entire data store

The Fig. 4 shows the three processes that govern the management of data. In the current version of the subsystem the three main functions (download, check and sync) are im-

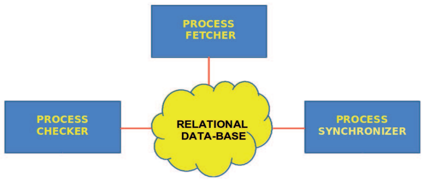

Fig. 4. The three process for the data management.

plemented by three independent processes. Processes are activated in asynchronous mode by the system utility ”crond”. The behavior of each process depends on its status and the status of other processes. The information is stored in a relational database avoiding the need to use mechanisms of interprocess communication (IPC) like message queues, fifos, etc.

3.1. The ”Fetcher” process

The Fetcher process takes care of downloading scientific data from the ISDC to the data storage subsystem of the IBIS data center. It behaves as follow:

- Checks if there is a new orbit to download;
- Downloads the orbit from ISDC
  * If the download is successful set the orbit state as ”UnCheckable”
  * If the download is unsuccessful:
    * Updates the log file
    * If the download has failed because of failure to read a file from ISDC, also updates the ”black list”
- The process makes a new connection to the data center and download a configurable number of orbits previus than the one just downloaded
- Change the status of the first orbit was no longer interested in the process of downloading as ”Checkable”

The use of black list avoid to download the unreadable files (e.g. because the permissions are inconsistent, the file is corrupt and so on). This means that the application will never get a ”process aborting” due uncorrect files analysis. The status of an orbit becomes ”Checkable” only when the orbit is considered established.

3.2. The ”Checker” process

The Checker process is responsible for monitoring integrity of files belonging to a given orbit. It behaves as follow:

- Selects the orbits that are in the ”Checkable” state
- Performs a structural control of the FITS files containing the data for the orbit
- Sets the orbit state in:
  * Good
  * Missing
  * Corrupted
- Writes in the database the name of the folder containing the orbit
- Retries the process for each selected orbit

3.3. The ”Synchronizer” process

The Synchronizer process takes care of maintaining the archive of scientific data synced with the secondary server (the server from which the cluster AVES takes the data to be analyzed). It behaves as follow:

- Selects the orbits that are in the ”Good” status
- Synchronizes the primary server with the secondary one
- If the process is successful, the synchronizer sets the status of orbit to ”Synced”
- Retries the process for each selected orbit

4. The necessity to check the orbits status

One of the main problems concerning the analysis of INTEGRAL scientific data is the fact that frequently they are downloaded corrupted from the repository of the ISDC. When the analysis software tries to use these files, the whole process gets error and stops. This event can also occur at the end of a long-term procedure (days, weeks, months ...), the damage in terms of time and economy is evident. Putting the status for each orbit into a database, we can perform the an analysis of a chosen science window without get corrupted data, because now we can process only the orbits checked as ”Good”, After completing the list of orbits to be analyzed, the user is able to submit the list to the software without further controls or checks on the correctness of the data.

5. Conclusion

The INTEGRAL data analysis is performed by providing a list of files to AVES containing information on the structure of data to be processed. By adopting the described strategies we achieved the following benefits:

- The orbits contained in that list should not be checked by a run-time application that require an execution time of the same order of magnitude as necessary to scientific analysis but simply compared with their status (good, corrupted or missing) provided in the database reference.
- The total treatment time of the data by the cluster, compared with that obtained from a single computer processor has been reduced by a factor of 200.
- With the adoption of the new storage subsystem, the speed of verification of the list was increased by a factor of more than 1000.

5.1. Future activities

Next steps to ameliorate the performances are already drawn. We want adopt a middleware able to distribute the data in a cloud and ”moving” computing activities where the data are. This paradigm, patented by Google in 2004, called Map-Reduce, will eliminate almost the network traffic. That will improve dramatically the performances of the total computing time. This kind of approach, due its architecture, will resolve automatically replication and redundancy of data. There are several open source implementations of this paradigm as middleware or for example embedded inside no-SQl databases. The first planned step is to study which one of these implementations will work better for us.

6. Discussion

FRANCO GIOVANNELLI: Is your Data Storage for the INTEGRAL Data Analysis easily adaptable for other missions?

BRUNO MARTINO: The solution is very general, so it is possible with few effort adapt it to other missions data.

FRANCO GIOVANNELLI: What is the cost of such a Data Storage Subsystem?

BRUNO MARTINO: The cost of the Data-Storage Subsystem is related to the hardware providing and to the customization of the software. For this release the cost was about 40,000 Euro. No cost for Software licensing because all the software adopted is under Open Source licensing.

Acknowledgements. The authors are grateful to Giuliano Sabatino purchasing manager of IASF-Rome for its valuable work. This system has been produced with the ASI contract. The IASF author acknowledge the ASI financial support via grant ASI-INAF I/008/07/0/. Special thanks goes to Fabio Guglietta for his kind contribution.

References

Budtz-Jrgensen, C., 2004 SPIE 5165, 139-150

Federici, M., et al. 2009, POS, Published online at `http://pos.sissa.it/cgi-bin/reader/conf.cgi?confid=96`, p.92

Goldwurm, A., et al. 2003, Astronomy and Astrophysics, 411, L223

Mas-Hesse, J. M., et al. 2003 Astronomy and Astrophysics, 411, L261

Ubertini, P., et al. 2003 Astronomy and Astrophysics, 411, L131

Vedrenne, G., et al. 2003, Astronomy and Astrophysics, 411, L63-L70

Winkler, C., et al. 2003, Astronomy and Astrophysics, 411, L1

Yoo, A., Jette, M. and Grondona, M. 2003, Job Scheduling Strategies for Parallel Processing, volume 2862 of Lecture Notes in Computer Science, pages 44-60