PROCEEDINGS OF SCIENCE

Integral Scientific Data Analysis on Blade Architectures: Storage and Resource Management

Bruno Luigi Martino [1]* **and Memmo Federici** [2]†

[1] *CNR-IASI: Istituto di Analisi dei Sistemi ed Informatica*
Via dei Taurini 19, 00185 Roma, Italy
bruno.martino@iasi.cnr.it

[2] *IAPS-INAF: Istituto di Astrofisica e Planetologia Spaziali*
Via Fosso del Cavaliere 100, 00133 Roma, Italy
memmo.federici@iaps.inaf.it

This paper presents the migration of the data processing system for the Imager on Board the Integral Satellite (IBIS) instrument from an architecture based on a cluster of machines interconnected by an high speed local network to a new one based on a set of SGI UV2000 Blade units. Several issues are tackled including the obsolescence of the hardware used and the effective management of resources, a fast access to the data required for the processing and the possibility of modular system growth in the context of its possible reuse in new mission environments

XI Multifrequency Behaviour of High Energy Cosmic Sources Workshop
25-30 May 2015
Palermo, Italy

*Speaker.

†Our dear friend Memmo died while we was working on the publication of this article.

PoS(MULTIF15)073

1. Introduction

The AVES computing system [1], located in the IAPS Institute (INAF), is based on an "Cluster" multi node architecture; it is a fully integrated low cost computing facility devoted to the storage and analysis of INTEGRAL data [2] and has been operating since 2008. AVES is a modular system that uses a software resource manager named Simple Linux Utility for Resource Management (SLURM) and allows very high expandability (65,536 nodes and hundreds of thousands of processors); the current configuration is composed of 30 Personal Computers equipped with Quad-Cores CPU's for a combined computing power of 300 Giga Flops (300×10^9 Floating point Operations Per Second), 120 GB of RAM and 7.5 Tera Bytes (TB) of storage memory. AVES was designed and built to cope with problems raised by the ever increasing amount of data provided by INTEGRAL mission (currently around 14 TB) growing by more than 1 TB each year. The analysis software used is the OSA package, distributed by the ISDC in Geneva. This is a very sophisticated package composed by a number of programs that cannot be used in parallel computing environments. To overcome this limitation we developed a set of programs capable of sharing the workload analysis on the various nodes making AVES able to automatically split the whole analysis into jobs assigned to different cores. This solution produces results close to those which can be obtained using parallel configurations. Inorder to simplify its use we developed tools so as to enable a flexible use of the scientific software and providing quality control of the on-line data storage.

2. AVES issues

AVES, although very powerful and still operational, highlighted some weaknesses related to long-term reliability. The system dates back to some time ago and the hardware with which it was built is now outdated; for this reason the replacement of individual faulty nodes has become increasingly complex (in some cases impossible). Furthermore, the 32 bit operating system and the cluster queue manager impose hard limits on the maximum quantity of RAM usable by software making it impossible to use more recent versions of the support tools [3]. The architecture defined in the original AVES project is not easily reusable in the context of new missions, with particular reference to issues related to confinement of resources to be allocated to end users. These considerations have led to the design and implementation of a new computing system devoted to INTEGRAL mission data analysis named AVES2.

2.1 AVES2 design

To overcome the limitations of the current AVES architecture, it has been necessary to resort to entirely different hardware and software platforms.

As the cost of machines based on BLADE technology decreased considerably they have become a viable alternative to large clustered systems. A blade-server system is a self-contained system including a number of computer units designed for high-performance and high-capacity computing; a machine of that kind can be regarded as a bare-bones stripped-down computer [4]. Trying to summarize the advantages of this architecture:

- reduced energy consumption

- improved power management
- simplified cabling and wiring requirements
- greater processing power in less space
- consolidated resources (such as storage and networking)

The above considerations make this platform capable of effectively supporting the INTEGRAL mission data analysis requirements and of becoming the starting point for future computing applications for the IAPS distributed computing laboratory (LCD) such as the study of the background noise of the detectors on board the ATHENA mission (Figure 1).

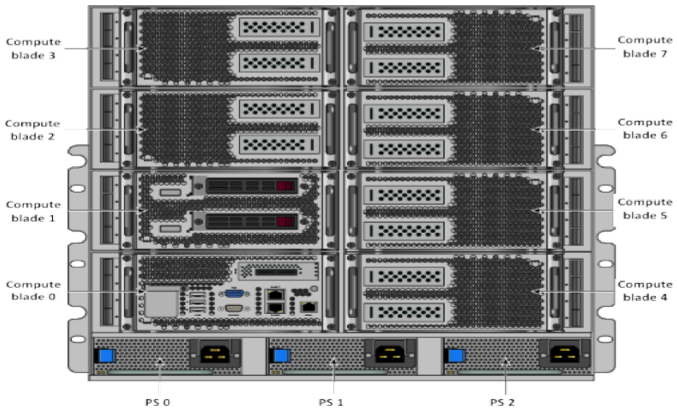

Figure 1: Current AVES2 IAPS configuration

2.2 UV 2000 overview

The SGI UV 2000 series is a family of multiprocessor distributed, shared memory (DSM) computer systems that can scale from 16 to 2,048 Intel processor cores as a cache-coherent single system image (SSI). The basic node on the UV 2000 has two single-socket servers with a vertical extender card sandwiched between the two stacked motherboards and linked together with a NUMAlink 6 hub chip (Figure 2).

	SGI UV 2000
CPU Speed (Cores)	Intel Xeon processor E5 - 4600 product family 2.4 - 3.3 GHz
Min/Max Sockets	4/256
Min/Max Cores (Threads)	32/2048 (4096)
Max Memory	64TB
Interconnect	NUMAlink 6
Enclosure	10U rackmount
Rack Size	Standard 10' Rack

Figure 2: UV 2000 highlights

3. OSA porting

The official analysis software for INTEGRAL (OSA 10.1) is not compilable on parallel systems [5]. We developed, with the AVES system, custom scripts capable of parallelizing serial analysis instances to OSA; heavy modifications of such procedures has also allowed their use on AVES2. The resource manager used in AVES project is SLURM (Simple Linux Utility for Resource Management) an open source, fault-tolerant, and highly scalable cluster management and job scheduling system for Linux OS. AVES2 is equipped by the OPEN LAVA [6] queue manager capable of allocating system resources to user jobs in a very granular mode (RAM, Cores number, CPU Time etc.) Openlava is a limited open source fork of LSF which is now owned by IBM. Openlava allows job scheduling on multiple nodes or cores; it does not come with a GUI but there is another project, Openlava web which enables the management via a web interface. Openlava enables one to defined critical jobs, users or groups when ensures that resources to these are available when they are needed. Furthermore, docker support is available for sofisticated future developments. The transition from SLURM to Openlava has not been particularly complex and the structure of the launch scripts remained virtually unchanged.

4. UI porting

The graphical interface is a crucial resource for users who need to use a computing system. As in the previous AVES system, users are driven to the insertion of the analysis parameters without having to type in long and complex text using the command line. The GUI uses a free and cross-platform program which provides GTK + dialog boxes inside bash scripts (Zenity); this program takes advantage of simple programming and a reduced consumption of CPU resources. After authentication and a choice of the working environment, the program assigns system resources to the user the according to the specified profile.

5. Resources optimization

To allocate to the users the necessary resources we used a less well known component of Linux kernel: Control groups (Cgroups). Cgroups is a feature that allows the limiting and insulation of resources usage (CPU, memory, disk I/O, network, etc.) of a collection of processes. Resources are assigned to each user preforming analysis, in particular:

- number of CPU cores
- maximum RAM memory
- maximum job lifetime

5.1 Cgroups and Namespaces

Cgroup is a Linux kernel feature which enables fine-grained resource partitioning between competing processes on the same machine. Its two priamry parts are, a controller set and a central core. which creates a hierarchical classification of processes running on the system. As an example,

the memory controller limits this resource for a group of processes; the block controller limits bandwidth for disk input/output, and similarly there are other controllers for other system resources [7]. From an application point of view, one can *detach* a set of processes from a specific layer of the kernel and assign them to a new one named *namespaces*. The chroot system available on UNIX/Posix systems is a primal form of namespaces: a process sees a completely new file system root and has no access to the original one. Linux extends this concept to the other OS layers (PIDs, users, IPC, networking etc.), so a specific process can live in a *virtual OS* with a new group of pids, a new set of users, a completely unshared IPC system (semaphores, shared memory etc.), a dedicated network interface and its own hostname. In a nutshell: Cgroups limits how much you can use, Namespace limits what you can see (and use).

5.2 Shepherd daemon

The nature of the multi-user and multitasking system, one must ensure that none of the authorized users saturate the resources of the machine. We ensure this by:

- limiting the number of processes launched by the user throughout their sessions
- maintaining a database of resource states
- updating the quantity of a given resource allocated to a user and ensuring profile restrictions using a timed daemon

This procedure is called *Shepherd daemon*.

6. Storage system

The storage system consists of a Linux server that automatically downloads the satellite data from the ISDC site, checks the consistency of downloaded files and tries, in case of failure or bad data, to download them again [8].

Figure 3: Qsan F600Q-D316 main controller

The main unit is composed of a SAN unit Qsan F600Q-D316 (Figure 3) which manages two set of magnetic disks of 48 TB each. These units are currently shared by the two computing systems AVES and AVES2:

- AVES is connected by Ethernet at 1 GB speed

- AVES2 is connected by Fibre Channel at 8 GB speed

AVES2 is equipped with a large amount of fast RAM (1TB). We decided to use some of it as a high speed virtual disk and the analysis time has been reduced considerably because a temporary file system (TMPFS) using local memory is faster than magnetic disks improving system performance of reading and writing files The RAM disk has improved performance over a SSD (Solid State Disk) by a factor of about 10.

7. Conclusions

This system through the creation of virtual machines from physical machines in active service, provides a greatly increased reliability. The ease and speed with which VMs are made active greatly improves maintenance time for the hosted systems. The low cost of individual VM makes it extremely interesting for the implementation of virtualized host systems over the many servers that make up the calculation facility for small and medium sized structures. The portability of VMs over different virtualization platforms makes the unavoidable transition to new generation architectures (because of the well known and continuous technological improvements) easy and economic from the management point of view.

Acknowledgments

Pietro Ubertini (IAPS Director), Franco Giovannelli and LOC

References

[1] M. Federici, B. L. Martino, P. Ubertini, *AVES: A high performance computer cluster array for the INTEGRAL satellite scientific data analysis*, *Experimental Astronomy Volume 34, p. 105-121* [2012]

[2] C. Winkler et al., *The INTEGRAL mission*, *A&A 411* [2003]

[3] M. Matsui, *How Far Can We Go on the x64 Processors?*, *13th international workshop, FSE 2006, Graz, Austria, March 15-17* [2006]

[4] Q. Haiping, et al., *Algorithms and Architectures for Parallel Processing*, *Springer* [2010]

[5] A. Goldwurm et al., *The INTEGRAL/IBIS scientific data analysis*, *A&A 411* [2003]

[6] A. Reuther et al., *Scheduler Technologies in Support of High Performances Data Analysis"*, *MIT* [2015]

[7] *LXC, linux conainers*, *URL http://linuxcontainers.org/.* [Accessed: 2015-05-20]

[8] B. L. Martino, M. Federici, *An high availability data storage subsystem for the INTEGRAL data analysis*, *Mem. S.A.It. Vol. 75, 282* [2008]

DISCUSSION

P. L. Biermann: Only 1TB for many users seems too little.

B. L. Martino: The actual configuration is: 2 x 48 TB SAN mass storage, 512 GB SSD (Operating System), 1TB of RAM memory. RAMdisk uses 256 GB of the total available RAM space, leaving 768 GB free for system and user applications.