Hybrid architecture for intensive data analysis

O. Terzo, L. Mossucca, P. Ruiu, G. Caragnano
Advanced Research on Computing Architecture and Security (ARCAS)
Istituto Superiore Mario Boella
Via P. C. Boggio 61, Torino, Italy
{*terzo, mossucca, ruiu, caragnano*}*@ismb.it*

***Abstract*—The new Italian GPS receiver for Radio Occultation has been launched from Satish Dhawan Space Center (Sriharikota, India) on board of the Indian Remote Sensing OCEANSAT-2 satellite. The Italian Space Agency has established a set of Italian universities and research centers to develop an infrastructure based on hybrid architecture, that is implemented for the overall processing Radio Occultation chain. The algorithms adopted can be used to characterize the temperature, pressure and humidity. In consideration of great deal of data to process, in case of saturation of physical resources, the system is able to start virtual machines on demand in order to solve temporary peak processing due saturation grid system.**

***Keywords*-radio occultation; grid computing; hybrid architecture; virtualization; scheduling.**

I. Introduction

The GPS Radio Occultation (RO) is a remote sensing technique for the profiling of atmospheric parameters (first of all refractivity, but also pressure, temperature, humidity and electron density, see [1] and [2]). It is based on the inversion of L_1 and L_2 GPS signals collected by an ad hoc receiver placed on-board a Low Earth Orbit (LEO) platform, when the transmitter rises or sets beyond the Earth's limb. The relative movement of both satellites allows a "quasi" vertical atmospheric scan of the signal trajectory and the profiles extracted are characterized by high vertical resolution and high accuracy. The RO technique is applied for meteorological purposes (data collected by one LEO receiver placed at 700 km altitude produce 300÷400 profiles per day, worldwide distributed) since such observations can easily be assimilated into Numerical Weather Prediction models. Anyway, it is also very useful for climatological purposes, for gravity wave observations and for Space Weather applications [5]. The system implements well consolidated RO algorithms through a processing chain which is subdivided into seven different software modules (namely Data Generators DG): these are executed in a sequential mode. Figure 1 represents a simple diagram of the processing chain and of the corresponding dataflow. Input data is ROSA observations (occultation and navigation data) that is made available by the ASI acquisition centre (namely CNM, Centro Nazionale Multimissione) and by the Indian counterpart, together with observations carried out by the International Geodetic Service (IGS) and other support data. This version implements RO state-of-the-art algorithms and, for the first time, it was developed and it runs on a distributed hardware and software infrastructure exploiting a grid computing strategy, which is called Web Science Grid (WSG). During the 2009 autumn season, the Indian OCEANSAT-2 mission carrying on-board the Italian ROSA (Radio Occultation Sounder of the Atmosphere) GPS receiver was launched. The Italian Space Agency [3] funded a pool of Italian Universities and Research Centers for the implementation of the overall RO processing chain, which is called ROSA-ROSSA (ROSA-Research and Operational Satellite and Software Activities). The ROSA-ROSSA is integrated in the operational ROSA Ground Segment it is operating in Italy (at the ASI Space Geodesy Center, near Matera) and in India (at the Indian National Remote Sensing Agency [4], near Hyderabad) starting from the 2009 autumn season. The paper is structured as follows: Section 2 explains the related work. Section 3 describes the project background. Section 4 presents the system architecture. Section 5 contains scheduling description. Section 6 draws the conclusions and future works.

II. Related work

The existent system is managed by integrated software, called Grid Processing Management (GPM), devoted to handle and process data of the OCEANSAT-2 on board sensor. This system consists of the following modules: nodes, repository, relational database, scheduler, agents and applications [6][7]. The physical nodes are located geographically in Italy, for accuracy to: Istituto Superiore Mario Boella (Turin), Polytechnic University of Turin (Turin), University of Padua (Padua), Sapienza University (Rome), University of Camerino (Macerata), International Center of Theoretical Physics (Trieste), INNOVA Information and Technology Consortium Group(Matera) and Institute for Complex System (Florence). The observed data, once acquired by the receiving ground station, are processed to produce refractivity, temperature and humidity profiles. The Radio Occultation (RO) events data processing consist of seven main steps, named Data Generators (DGs). All DGs are executed in series, these are SWOrD, DG_BEND, DG_BDIF, DG_BISI,

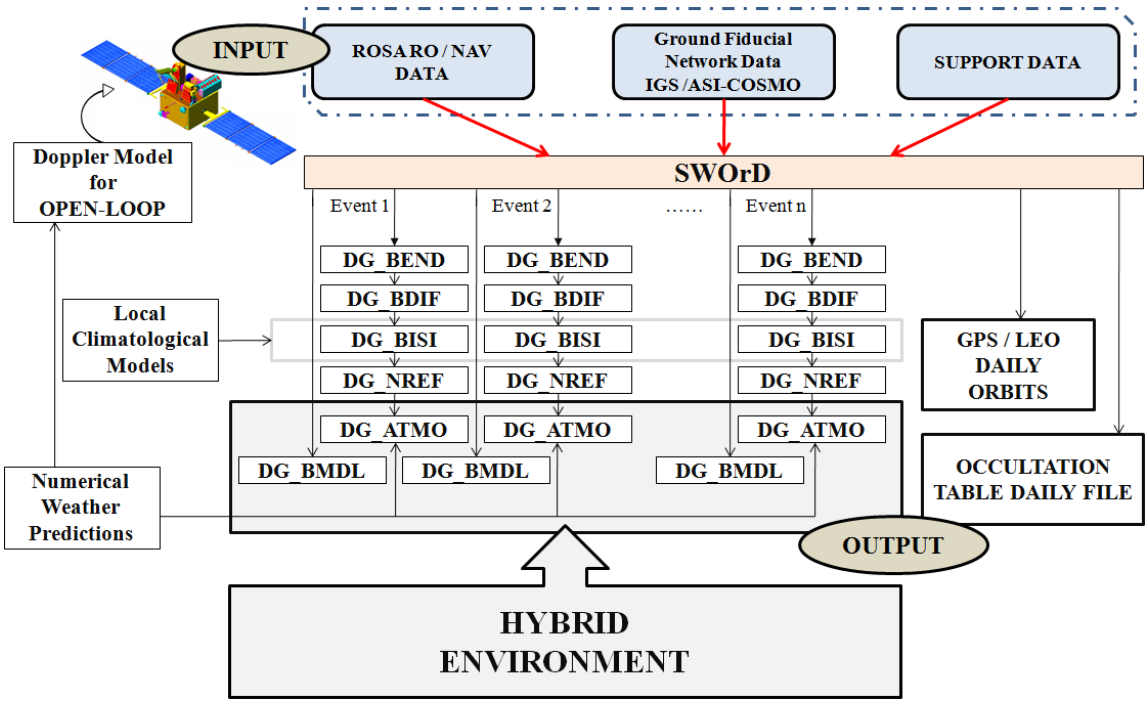

Figure 1. ROSA-ROSSA chain processing.

DG_NREF, DG_ATMO and DG_BMDL (see Fig. 1). The input and output files cover a 24 hours time interval. The output are daily data therefore for each day about 256 files event need to process in sequential way, for further details see [11]. In this context, where one needs to elaborate an enormous amount of data, using a grid architecture, there is already a great saving of time. But in some cases the system fills up reducing processing chain time. A solution to solve this problem can be to use virtual machines. The architecture proposed allows to create a virtualized environment, which allows to activate virtual machines on demand, in order to increase the computational power and to solve temporary peak processing.

III. Project Background

The Hybrid Architecture consists of a number of virtualized nodes integrated into a grid composed of physical nodes. With physical node, we mean a machine that runs an operating system that has the exclusive use of the underlying hardware. The virtualized node is instead an instance of a virtual machine that can share resources with other nodes, managed by the hypervisor (see Fig. 2). The project stems from need of computational power in case of an unexpected burst of calculation that the physical infrastructure would not be able to respond on its own. In these cases, where the physical grid has saturated its resources, the system asks to the hypervisor for new virtualized nodes which are according on rules sets in the scheduler that will be discussed later. This architecture allows to profit from the grid (increasing computing power through the pooling of resources calculation, etc.) and from the virtualization (flexibility, scalability, cost reduction, etc.). The use of virtual nodes in addition to the physical nodes in the grid has considerable advantages. Virtual machines can be blocked shots and quickly reboot every time you need, without loss of information or problems to the chain flow of execution during the processes on the machine. Tests were performed on virtual machine to estimate startup time and the result is about 9500 ms. In addition to reduced startup time, the use of VM in the grid brings other benefits including load balancing and high availability. The load balancing allows migration of virtual machine from a physical box to another, in order to balance system performances; an high available system ensures migration of virtual machine when maintenance shall be paid on the server, avoiding possible (and usually lengthly) discontinuity in service provisioning. Startup time is the difference in milliseconds from the time when hypervisor receives a request to start the VM to the time when the VM is accessible on the network, and ready for a job execution.

IV. Virtualization Technology

The Xen hypervisor is a layer of software that replaces the operating system running directly on the hardware of the computer. It is released under GPL license for x86-compatible and was developed at the University of Cambridge. It was decided to use Xen as virtualization platform because it is an open source product and is one of the

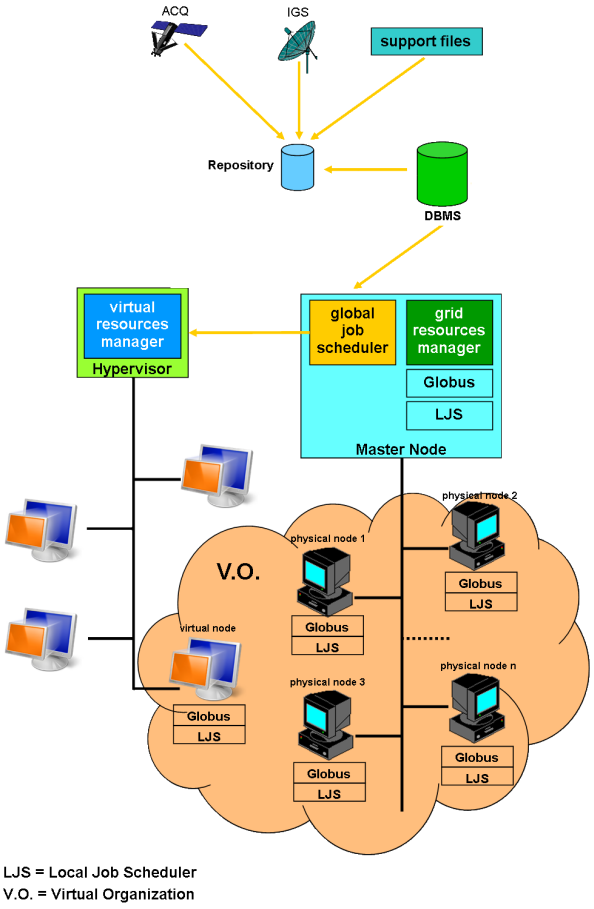

Figure 2. Hybrid architecture schema.

few hypervisor that supports both paravirtualization and full virtualization. Xen Hypervisor is the direct interface between virtual machines and the hardware, and receives all requests for CPU, I/O and disk usage. Due to the separation between the OS and hardware, the hypervisor can run multiple operating systems safely and concurrently. The Domain 0 (Dom0) is the only domain that can access to the hypervisor and can manage (create, shutdown, suspend, etc.) the virtual machines. The Domain Guests (DomU) are controlled by the Dom0 and can operate independently in the system. The DomU can be of two types: paravirtualized or fully virtualized, also called Hardware Virtual Machine (HVM).The paravirtualized systems are aware to have no direct access to the hardware and they use special instructions to interact with the kernel. In this case, the virtual machine operating system must be adapted for the purpose. HVM machines are not aware that they are virtualized and interact with the hardware as if they were running directly on it: in reality all messages are filtered by the hypervisor. The xm comands are the main interface for managing the guest domain. Thanks to them, it is possible to create, pause, and shutdown the VMs, but also enable or pin VCPUs and attach and detach virtual block devices. The commands are listened by a daemon called xend.

V. Resource Scheduler

As mentioned above, in case of saturation of physical resources, the system is able to automatically start virtual machines: it can support the grid taking in charge the execution of jobs. For this purpose, it has been designed and implemented a Resource Scheduler that decides to allocate new virtual machines, depending on specific dynamics. The scheduler is a process thet runs on the hypervisor and consists of several bash scripts allowing the monitoring and the collection of data which will be used for calculating system parameters and for the generation of log files [8][9]. The logic model determines the allocation of the VM, it is based on the observation of the status of the grid: specifically processes execution and machines availability. These information are retrieved by querying the database hosted on the master node of the grid. The nodes, belonging to the grid, send data to the database on the master node periodically. These data are information about the state and the workload of the system (RAM allocated, average CPU usage). In particular, combining these information it is possible to determine two fundamental elements: the state of the virtual machine and the parameter CUI. The possible states of the virtual machine are two: the first is the status of ”Available, which indicates that a virtual machine is connected to the grid and has resources available to be able to perform the job (this means that the value of the CPU is between 0% and 5% use). The second state is ”Running”, and indicates that a virtual machine has a queue of files to process c, where $c > 0$, and system resources are committed to process a job (the CPU is greater than 5% and above 70% for a long period of time). With these data is possible to calculate a parameter that is an indication level about use of Grid resources, called Computational Usage Index (CUI), which represents the ratio between the number of running nodes and the number of available nodes (Eq. 1).

$$CUI = \frac{\sum_{i=1}^{N} RunningNodes}{\sum_{i=1}^{N} AvailableNodes} \quad (1)$$

The CUI is compared with bound values called start threshold and stop threshold. The value of the two thresholds are static and were calculated on the basis of scientific considerations that are not covered by this study. The first value, start threshold, indicates the saturation of the grid. If the CUI is greater than this value it is necessary to instantiate a new virtualized node. The resource scheduler starts a VM and sends to the master node information about availability of new node just started. As soon as the master node, where resides the job scheduler, detects the new virtualized node it can begin to assign the job. The stop threshold is the value that shows grid resources are underused and is therefore

time to switch off the VM. The scheduler, once this limit is exceeded, tell to the master node that the virtualized node is no longer available to receive job. Then it will proceed to shutdown the machine, after to have verified that there are no file transfers in progress, there is not a queue of files to be processed, the processing job is actually completed.

VI. Performances test

During a test phase we evaluated the elaboration time of each Data Generators testing between two types of nodes: physical and virtualized node. The server used for testing is equipped with a dual-core Intel Xeon (4 CPU), 8 GB of RAM and 130 GB of storage. The operating system is Ubuntu server. The guest machines reside entirely on this server and therefore they share the resources (RAM, CPU, disk): each machine has 2 GB of RAM and 2 dedicated CPUs. Virtualized nodes are configured exactly like a physical node of the grid. It has been installed the softwares used for the chain processing and some system tools for the local job scheduling and monitoring of the resources. It was decided to use paravirtualized systems since it was shown that (in terms of network and I/O), they have better performances than the fully virtualized one [10]. In Fig. 3 a comparison of elaboration time is depicted. α is the weighted average elaboration time for each node belonging to the grid (Eq. 2).

$$\alpha_i = \frac{\bar{t}_i}{\sum_{j=1}^{N} \bar{t}_j} \tag{2}$$

For each Data Generators, W represents a ratio between the virtual machine processing time and physical machine processing time (Eq. 3).

$$W_i = \alpha_i \frac{t\_virt_i}{t\_phy_i} \tag{3}$$

The execution time of algorithms DG_BDIF, DG_BISI, DG_NREF executed on the virtualized machine are almost comparable to the execution time on the physical machine. However, if the algorithm is executed on the virtualized machine DG_ATMO has a slight delay, estimated in 5% , compared with physical machines. The most important result is noticeable by observing DG_BEND: the execution time of this algorithm on the virtual machine is more quickly of about 23.5% than its execution on the physical machine (see Fig. 4).

VII. Conclusions

The ROSA-ROSSA software implements Radio Occultation technique, which run for the first time on a hybrid infrastructure. This paper want to be an improvement of a projects based on grid computing to solve temporary peak processing due saturation system. In frameworks such as Radio Occultation, where the amount of data to be processed is significant, the use of a hybrid architecture as the grid can

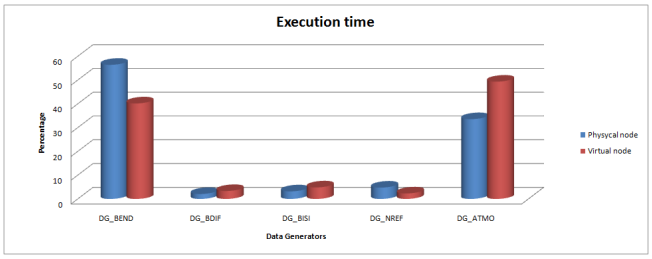

Figure 3. Execution time.

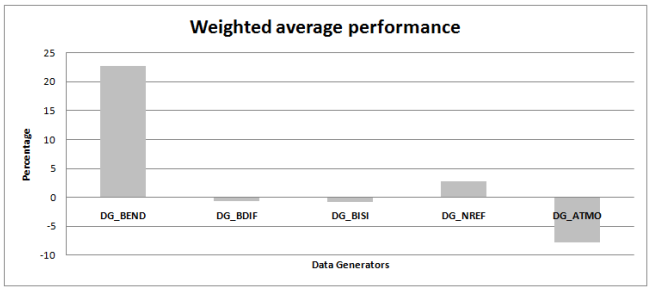

Figure 4. Weighted average performance.

be the best choice. We have focused on the implementation of a intelligent scheduler that can manage also the virtualized side of the infrastructure in order to assign jobs to nodes in an automatic way without any human interaction. When the algorithms are executed on virtual machines there is not a decline in system performance, but conversely, in the case of the algorithm DG_BEND, VMs have better performance than physical ones. Looking at tests results emerge that in a future scenario the Scheduler could assign the execution of DG_BEND algorithm only to virtual machines. By adopting this mechanism would be possible to reduce the execution times of the entire processing chain. Also, as future works we plan the extension of the proposed architecture to computer clusters available across the European Grid Infrastructure (EGI) and we are studying a solution for EC2 environment by Amazon to allow to further increase available computing power.

VIII. Acknowledgments

The authors are grateful the Italian Space Agency (ASI) for supporting this project within contract I/006/07/0 and to all the ROSA-ROSSA partners for their contributions.

References

[1] Melbourne, W., *The application of spaceborn gps to atmospheric limb sounding and global change monitoring*, JPL Publ, pp. 18-94, 1994

[2] Kursinski, E.R., *Observing Earth's atmosphere with radio occultation measurements*, Journal Geophys. Res., pp. 429-465, 1997

[3] Italian Space Agency (ASI), *http://www.asi.it/*, 2010

[4] Indian National Remote Sensing Agency (ISRO), *http://www.isro.org/*, 2010

[5] Wickert, J., *The radio occultation experiment aboard CHAMP: Operational data processing and validation of atmospheric parameters*, Journal Meteorol. Soc. Jpn, pp. 381-395, 2004

[6] Foster, I. and Kesselman, C., *The Grid2: Blueprint for a New Computing Infrastructure*, Morgan Kaufmann, pp. 38-63, 2003

[7] Berman, F., Fox, G. and Hey, G., *Grid Computing Making the Global Infrastructure a Reality*, Wiley, pp. 117-170, 2005

[8] Dimitriadou, S. and Karatza, H., *Job Scheduling in a Distributed system Using Backfilling with Inaccurate Runtime Computation*, The international conference on complex, intelligent and software intensive system, pp. 329-336, 2010

[9] Xhafa, F., Pllan, S. and Barolli, L., *Grid and P2P Middleware for Scientific Computing Systems*, The international conference on complex, intelligent and software intensive system, pp. 409-414, 2010

[10] Chierici A., Verald R., *A quantitative comparison between xen and kvm*, 17th International Conference on Computing in High Energy and Nuclear Physics (CHEP09), IOP Publishing Journal of Physics, 2010

[11] Mossucca L., Terzo O., Molinaro M., Perona G., Cucca M., Notarpietro R. *Preliminary results for atmospheric remote sensing data processing through Grid Computing*, The 2010 International Conference on High Performance Computing and Simulation (HPCS 2010), 2010