An attempt to integrate Clouds in Grids

Giuseppe Andronico[1]

Italian National Institute of Nuclear Physics, Division of Catania
Via S. Sofia 64, 95123 Catania, Italy
E-mail: *giuseppe.andronico@ct.infn.it*

Roberto Barbera

Department of Physics and Astronomy of the University of Catania and INFN
Viale A. Doria 6, 95125 Catania, Italy
E-mail: *roberto.barbera@ct.infn.it*

Andrea Fornaia

Italian National Institute of Nuclear Physics, Division of Catania
Via S. Sofia 64, 95123 Catania, Italy
E-mail: *andrea.fornatia@ct.infn.it*

Salvatore Monforte

Italian National Institute of Nuclear Physics, Division of Catania
Via S. Sofia 64, 95123 Catania, Italy
E-mail: *salvatore.monforte@ct.infn.it*

Over the last decade, Grid computing has become an increasingly hot topic in the world of new technologies and European scientific research has benefited from the increasing availability of computing and data infrastructures with unprecedented capabilities put together in the framework of large scale projects and initiatives. However, during the last few years, interest has gradually shifted, especially in the world of enterprises, from Grid computing to an independent and complementary new computing paradigm: Cloud computing. Both Grid and Cloud computing provide access to a large compute or storage resource but Clouds massively exploit virtualization to provide uniform interface to the underlying resource thus hiding physical heterogeneity, geographical distribution and faults. Although Grid technology continues to dominate the public academic sector and scientific computing environments, new interests have raised in deploying cloud technology on Grid-enabled resources to improve the management and reliability of those resources via the virtualization layer. The aim of this paper is to present an attempt to integrate both technologies in a use case where the virtualization layer of the Cloud is "seen" by the Grid as a resource.

---

[1]   Speaker

1. Introduction

During the last few years, interest has gradually shifted from Grid Computing to an independent and complementary paradigm: the Cloud Computing. Both Grids and Clouds provide access to large, distributed, compute and storage resources but Clouds massively exploit virtualization to provide uniform interface to the underlying resource thus hiding physical heterogeneity, geographical distribution and faults. While there are many similarities between the two computing models, the differences are those that matter most. Grid computing is better suited for organizations with large amounts of data being analysed by a small number of users, or few but large allocation requests, whereas Cloud computing is better suited to environments where there is a large number of users requesting small amounts of data, or many but small allocation requests.

In other words, one could say that Grids are well suited for complex works performed by Virtual Organizations while Clouds are well suited for simple works performed by single users or small groups. Another key difference between the two technologies is that Grids mostly perform batch job scheduling and have sophisticated policies for job allocation while Clouds, by their nature, do not do this and are more used for interactive jobs [1].

Although Grid technology continues to dominate the public academic sector and scientific computing environments, due to the collaborative nature of such communities and the need to go across resources' organizational boundaries, new interests have recently been raised in deploying cloud technology on Grid-enabled resources to improve the management and reliability of those resources via the virtualization layer

The aim of this paper is to present an integration example of both technologies where the virtualization layer of the Cloud is interconnected to the Grid infrastructure owned and operated by the Consorzio COMETA [2]. The proposed architecture allows users to increase their degree of choice between various software and hardware systems by enabling submission of Virtual Machines (VMs) as canonical Grid jobs thus extending the Cloud paradigm to benefit from Grid concepts by integrating federated access control and distributed resource sharing, as well.

The paper is organised as follows: Section 2 provides some background information and the motivations that triggered the exploitation of Cloud technologies to support the integration of Grid and Cloud computing within the COMETA infrastructure whereas Section 3 presents a detailed description of the whole architecture as well as the interactions among its components.

2. Background and motivations

The interest in working on this attempt is maturated in the reference frame of two separated entities in which the authors are involved; the EPIKH project and the Consorzio COMETA.

The EPIKH Project

The EPIKH [2] Project (Exchange Programme to advance e-Infrastructure Know-How), funded by the European Union Seventh Framework Programme as a Marie Curie action, is

performing an activity of knowledge dissemination in several areas of the world, from Latin America to Africa and Asia, on the field of Grid computing. The strategic aims of the EPIKH project are to reinforce the impact of e-Infrastructures in scientific research defining and delivering and stimulating program of educational events, including Grid Schools and High Performance Computing courses, as well as to broaden the engagement in e-Science activities and collaborations both geographically and across disciplines. These goals translate into the following specific actions:

- Spreading the knowledge about the "Grid Paradigm" to all potential users: both system administrators and application developers through an extensive training programme;
- Easing the access of the trained people to the e-Infrastructures existing in the areas of action of the project;
- Fostering the establishment of scientific collaborations among the countries/continents involved in the project.

The COMETA Grid

The COMETA Grid infrastructure is owned and managed by the Consorzio COMETA, a not-for-profit Organization established in Catania (Italy) in 2005 and formed by the Universities of Catania [4], Messina [5] and Palermo [6], the Italian National Institute of Nuclear Physics (INFN) [7], the Italian National Institute of Astrophysics (INAF) [8], the Italian National Institute of Geophysics and Volcanology (INGV) [9] and the SCIRE Consortium [10] which brings together both public research institutions and private enterprises. Figure 1, shows the locations of the COMETA sites in the three towns of Catania, Messina and Palermo.

Figure 1 – The COMETA Grid infrastructure.

The aims of this infrastructure are to:
- Create a virtual laboratory in Sicily, both for scientific and industrial applications;
- Connect the Sicilian e-Infrastructure to those already existing in Italy, Europe and outside;
- Foster the adoption of Grid computing for massive computations, improve the competitiveness of e-Science and e-Industry "made in Sicily";
- Trigger the start-up of spin-off in the ICT area;
- Contribute to the reduction of the endemic problems of the "digital divide" and the "brain drain" of brilliant people to other parts of Italy and beyond.

The Sicilian Infrastructure has been funded with almost 15 M€ over three years by two projects: the TriGrid Virtual Laboratory [11], funded by the Regional Operating Program (POR) [12] managed by the Department of Industry of the Sicilian Regional Government and PI2S2 [13], funded by the National Operating Program (PON) and the Italian Ministry of Education, University and Research (MIUR).

The computational and storage resources of the e-Infrastructure are exposed to the end users through the virtual services of the gLite middleware [14], developed in the context of the European Commission funded project EGEE [15]. The adoption of this middleware makes the Sicilian e-Infrastructure fully interoperable with the others already existing in Italy and abroad.

A bit more than 120 applications were ported on the Consorzio COMETA infrastructure, several of them relevant for the industrial world.

Motivation of the work

Both EPIKH and Consorzio COMETA need a demonstrative solution able to add a Cloud computing interface to the e-Infrastructure.

In the case of EPIKH, this requirement comes from some partners, claiming that inserting such a solution in the curricula of the schools increases the interest of students and researchers in participating to the project events.

Consorzio COMETA, instead, is proposing itself as a computing and storage service provider for the industrial world where the interest for Cloud computing is high. Also in this case at least a demonstrative solution is needed, with the objective to customize it with requirements coming together with agreements.

3. The proposed Grid/Cloud integration

Nowadays, Grid's users are experiencing incompatibility problems between their applications and the software available on the Grid infrastructure which support only specific software installed on its Worker Nodes. Moreover, the Grids lack in supplying their users with a standard software installation mechanism. In other words, due to the intrinsic heterogeneous and distributed nature of the Grids, users may find difficulties when trying to submit some specific application. It is possible to basically overcome these limitation exploiting two possible

solution based on respectively on resource or machine virtualization. The proposed architecture is based on the "machine virtualization" approach and allows users to increase their degree of choice between various software and hardware systems by enabling submission of Virtual Machines as canonical Grid jobs thus extending the Cloud paradigm to benefit from Grid concepts by integrating federated access control and distributed resource sharing, as well.

One of the key points of the proposed architecture is that it is decoupled from the local resource manager systems available at Grid sites. More precisely, the Cloud infrastructure is "seen" by the Grid middleware as a local resource manager system thus becoming a Grid resource by itself, handling entire Virtual Machines rather than normal Grid jobs.

Architecture overview

The Grid components involved in our preliminary integration test-bed are basically those already deployed and working in the European infrastructure relevant to the gLite middleware architecture.

The OpenNebula [16] FrontEnd has been installed on a CREAM (Computing Resource Execution And Management) Service [17] node, which will be responsible for Virtual Machine management operations at the Computing Element (CE) level.

Installing the OpenNebula front-end on the same machine where a CREAM CE service is up and running has the advantage of demanding authentication and authorization requests directly to the CREAM service itself. Moreover, the CREAM CE allows to forward, via the BLAH component, requirements to the batch system. To such an aim the *CERequirements* attribute, described in the CREAM JDL guide can be used.

For direct submissions to the CREAM CE (e.g. jobs submitted to the CREAM CE using the CREAM CLI glite-ce-job-submit command) the *CeRequirements* attribute is supposed to be filled by the end-user whereas, for jobs submitted to the CREAM CE via the WMS, the *CeRequirements* attribute is instead filled by the WMS.

The *CERequirements* expression received by CREAM is then forwarded to BLAH which, basically manages it by setting some environment variables are available and can be properly used by the $GLITE_LOCATION/bin/xxx_local_submit_attributes.sh scripts (e.g. pbs_local_submit_attributes.sh for PBS/Torque, lsf_local_submit_attributes.sh for LSF, etc.). These scripts must be properly created by the site admin and are easily customizable for implementing new features.

Using the attributes forwarding mechanism described above it had been possible to make CREAM accepts VM submission requests described with the same Job Description Language (JDL) used to describe jobs submitted to the gLite Workload Management System (WMS) [18] and other VM management requests (e.g., VM cancellation, VM monitoring, etc.) forwarding them to the attached OpenNebula cluster nodes. For this purpose, the following attributes have been introduced in the JDL language and should be added to the *CeForwardParameters* attribute in the WMS configuration file (i.e./opt/glite/etc/glite_wms.conf) to trigger the forwarding mechanism previously described:

• **ONE_VM_CMD**, describes the command which should be executed at the CREAM CE/OpenNebula node, *i.e. submit, hold, resume, cancel, status*;

- **ONE_VM_IMG**, specifies the VM disk image user has previously uploaded and registered on the SToRM service;
- **ONE_VM_NET**, describes network connectivity mode the VM will be using at the OpenNebula Cloud side (*e.g. bridged, NAT*);
- **ONE_VM_ID**, describes the unique identifier of an already submitted VM;
- **ONE_VM_CONTEXT**, specifies the ISO image containing configuration parameters that will be handled for later use inside the VM; this method is what OpenNebula provides to pass configuration parameters to a newly started Virtual Machine, as recommended by OVF.

From a user point of view, for each VM he/she wants to execute in the Grid environment, a JDL file should be prepared and submitted to the gLite WMS. Thus, submitting and a VM is as simple as submitting a job to the Grid.

The gLite CREAM CE service has been patched for handling the VM submission requests. Once a VM request arrives to CREAM, using the forwarded information to the batch system the corresponding OpenNebula deployment file is generated by the patched submission script and the requested command is then forwarded to the OpenNebula front-end for the actual execution. The patched CREAM scripts are also responsible for returning VM information back to the user (e.g., IP, ID, etc.) using the *onevm.info* file within the OutputSandBox of the job. Once they have retrieved the *onevm.info* file, using the *glite-wms-job-output* command, users may either connect to (e.g. via SSH) or control the just started VM. This last action is as simple as submitting new jobs having the ONE_VM_ID and ONE_VM_CMD attributes in the JDL respectively assigned with the unique identifier of the VM, as specified in the *onevm.info* file, and the desired VM control command, i.e. *status, hold, cancel*.

In this way, an OpenNebula cluster is basically "seen" at the Grid side as a Local Resource Manager System to control entire VMs rather than single and simple Grid jobs.

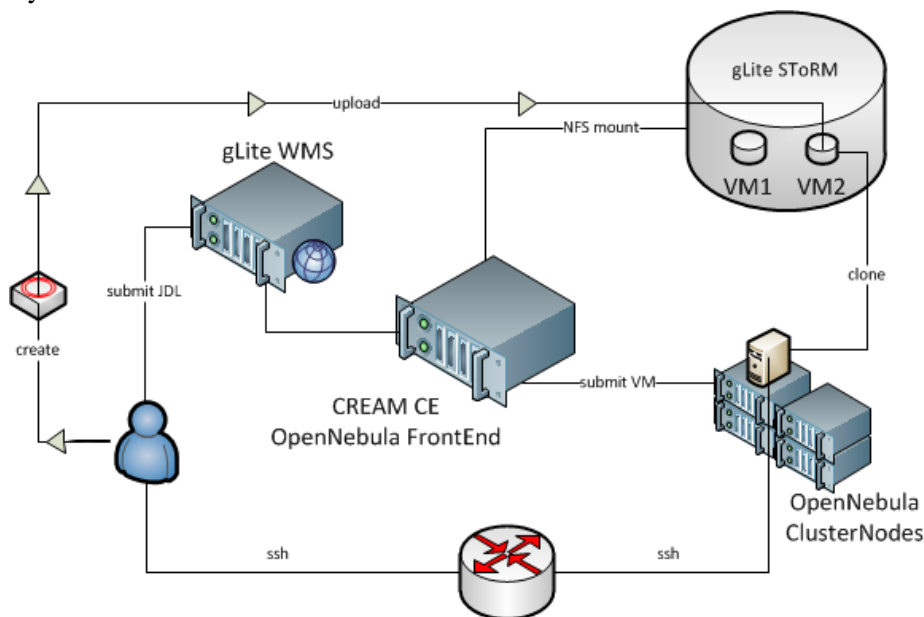The following figure describes the interactions among the various components of the proposed system.

Figure 2 – Architecture of the solution proposed in the paper.

As far as the storage is concerned, the gLite SToRM [19] service has been configured for registering users' virtual machine images and using it as shared storage repository for the OpenNebula cluster nodes. By default,for each supported VO, SToRM creates a subdirectory where users may store their own data (e.g. Virtual Machine images). In the preliminary prototype described in this paper the root folder containting the VO directories is exported and mounted via NFS by the OpenNebula ClusterNodes.

4.Conclusions and future works

In this paper the design principles and a preliminary prototype of a Cloud to Grid integration has been presented, as a tool to satisfy specific needs coming both from the EPIKH projects and the initiative the Consorzio COMETA is carrying on for enterprises. The proposed architecture allows users to increase their degree of choice between various software and hardware systems by enabling submission of Virtual Machine as canonical Grid jobs, without requiring any interaction with system/site administrators for deploying the VMs users wish to be executed. Moreover, the system is decoupled from the Grid local resource manager system as the Cloud infrastructure is "seen" by the Grid middleware as a resource manager itself handling entire Virtual Machines rather than single and simple Grid jobs.

Grid computing provides a security framework for identifying inter-organizational parties, managing data access as well as movement and utilization of remote compute and storage resources. Integrating a Cloud in a Grid infrastructure strengthen the security with the robust federated identity and access management architecture of the Grid infrastructure. Moreover, the use of a Storage Resource Manager such as SToRM helps identifying and consolidating the Transfer and Storage Policy concepts within a Cloud by exploiting ACLs on the underlying file system to enforce file access permission and providing guaranteed space reservation. This allows as well to take advantages of special features provided by the underlying file system implementation. Clearly some intensive testing sessions should be performed as a future work, in order to check whether SToRM is adequate enough to handle the i/o patterns characteristic of a shared Virtual Machine repository which differ from the most common data analysis ones.

In order to keep tracks of the resource consumed in terms of CPU cycles at the OpenNebula Cluster Nodes running VMs a tight integration between OpenNebula and the BLAH service should be provided as a future work. A new BLAH LogServer for OpenNebula should be written in order to retrieve information about VMs by parsing the log files created by the OpenNebula cluster. The obtained information will be then avalaible for accounting purposes.

Acknowledgments

References

[1] *Grid and Cloud Computing, A Business Perspective on Technology and Application*, X ed.: S.Slabeva Katarina, W.Thomas, R.Santi, 2010, ISBN:978-3-642-05192-0.

[2] *Consorzio COMETA [Online]* http://www.consorzio-cometa.it/

[3] *EPIKH Project [Online] http://www.epikh.eu/*

[4] University of Catania – www.unict.it

[5] University of Messina – www.unime.it

[6] University of Palermo – www.unipa.it

[7] The National Institute For Nuclear Physics (INFN) – www.infn.it

[8] The National Institute for Astrophysics (INAF) – www.inaf.it

[9] The Italian National Institute of Geophysics and Volcanology (INGV) – www.ingv.it

[10] The SCIRE Consortium – www.consorzioscire.it

[11] The TriGrid VL Project – www.trigrid.it

[12] the Regional Operating Program (POR) - www.dps.tesoro.it/documentazione/QSN/docs/PO/POR_Sicilia_FESR_SFC2007.pdf

[13] The PI2S2 Project – www.pi2s2.it

[14] gLite – Lightweight middleware for Grid Computing – glite.cern.ch

[15] The EGEE Project – http://public.eu-egee.org/

[16] OpenNebula: The Open Source Toolkit for Cloud Computing. [Online]. http://opennebula.org/

[17] P. Andreetto et al*., CREAM: A simple, Grid-accessible, Job Management System for local Computational Resources*, Proc. XV International Conference on Computing in High Energy and Nuclear Physics (CHEP'06), Feb 13-17, 2006, Mumbay, India, Macmillan, p. 831-835, ISBN 10:0230-63017-0, ISBN 13:978-0230-63017-8

[18] European Middleware Initiative project, [Online]. http://www.eu-emi.eu/

[19] The StoRM Project, [Online]. http://storm.forge.cnaf.infn.it