StratusLab

Enhancing Grid Infrastructures with
Virtualization and Cloud Technologies

Installing and operating a production grid site in the StratusLab cloud: Experience and issues

Technical Note TN-GridOverCloud (V1.0)
2 June 2011

Abstract

StratusLab provides a complete, open-source solution for deploying an "Infrastructure as a Service" (IaaS) cloud infrastructure. Deployment and operation of a grid site on top of an IaaS cloud poses a number of challenges if we wish to take full advantage of the cloud service capabilities. In this technical note, we report our experiences from the installation of a production grid site on top of StratusLab's reference cloud service, and we provide various suggestions for areas that need improvement.

StratusLab is co-funded by the
European Community's Seventh
Framework Programme (Capacities)
Grant Agreement INFSO-RI-261552.

Contributors

Name	Partner	Sections
Evangelos FLOROS	GRNET	All
Athanasia ASIKI	GRNET	All
Kyriakos GINIS	GRNET	All
Cal LOOMIS	LAL	Review and corrections

Document History

Version	Date	Comment
0.1	07 May 2011	Initial version for comment.
0.2	19 May 2011	Reviewed internally in GRNET
1.0	17 May 2011	Review and comments from LAL

Contents

List of Tables

1 Introduction

StratusLab provides a complete, open-source solution for deploying an “Infrastructure as a Service” (Iaas) cloud infrastructure. One of the main use cases we wish to support is the operation of grid sites on top of IaaS cloud services. During the initial phases of the project we experimented extensively with the installation and operation of grid sites on top of cloud services. In parallel, we prepared a number of Virtual Machine (VM) appliances for the basic machine types of a gLite-based grid site [3], namely: the Computing Element (CE), the Storage Element (SE), the Worker Node (WN), the User Interface (UI) and the APEL service (used for site accounting). All these images are available from the appliance repository.[1] These appliances currently follow the evolution of gLite middleware; with every new gLite release, a new image snapshot is created and is uploaded to the repository.

In order to fully evaluate the capability of cloud services to support the operation of grid sites, we deployed a production grid site on top of the project’s reference cloud service running in GRNET. The site named HG-07-StratusLab was certified within the GRNET NGI (the Greek National Grid Initiative) and has joined the Greek national grid infrastructure (HellasGrid). The site offers a CE and 8 dual-core WNs thus providing a total capacity of 16 cores for job submission. (Complete details are available from the site’s GStat page.[2]) The site supports MPICH-2 and OpenMPI parallel jobs. Each WN is configured with 4GB of main memory. The site also provides an SE that offers a total storage space of 2TB. The storage is currently configured directly as an NFS mountpoint from the local storage server and is not yet virtualized (i.e. it cannot be managed as a persistent block storage service from the StratusLab command line tools).

This technical note summarizes our findings from the installation of the HG-07-StratusLab site and operations of it within the EGI pan-European grid infrastructure. In particular, we consider issues related to the installation, configuration and daily operation of the site. We identify issues that, in our opinion, impede the optimal exploitation of cloud technologies for the provision of grid services.

[1] http://appliances.stratuslab.eu
[2] http://gstat-prod.cern.ch/gstat/site/HG-07-StratusLab/

2 Installation

The installation process in a cloud environment depends on the availability of appropriate pre-configured VM images. Typically grid software will come pre-packaged in virtual appliances. It is expected that the appliance providers will maintain images with the base OS required by grid services (e.g. Scientific Linux 5.5 or CentOS 5.5) as well as the RPM packages needed for a specific grid node like a Computing Element, Storage Element, Worker Node, etc. Additionally the appliance provider should have configured the necessary yum repositories that enable the quick update of the VM instance installed software. As a good practice, it is recommended that the grid administrator using these VMs should run the command to update packages upon image instantiation at the first boot of the VM.

In the context of StratusLab we have prepared appliances for the basic grid site machine types (CE, SE, WN, UI and APEL). We expect that this task will be eventually taken over by the cloud middleware providers (e.g. EMI) or the grid infrastructure supervisors (EGI, NGIs, etc). These appliance providers should make sure that the VMs follow the evolution of grid middleware and that the appliances are validated (for security, functionality, etc.) before their release.

3 Configuration

3.1 Service configuration

Grid services are configured using the YAIM tool [5]. YAIM uses a set of configuration files in order to fine-tune all the aspects of a grid site such as the number of worker nodes, the creation of user accounts, the number of VOs supported by the site and the information that the site broadcasts to the central grid information system.

Overall the YAIM system is very static and assumes a very homogenous hardware setup where all Worker Nodes expose same capabilities and hardware characteristics. The administrator has to complete a large amount of information and to make configuration decisions at the setup of the grid site that remain unchanged. In order to make any changes in the existing configured setup, the grid admin has to edit the respective configuration files and re-issue the yaim command.

3.1.1 Site locality

Geographical information defined by SITE_LAT and SITE_LONG macros might not be available to the grid site admin due to lack of knowledge about the location of the cloud datacenter. Even if this information is available it might not remain the same during the lifetime of the grid site. If we consider an architecture of federated cloud providers distributed in different regions in the same country or across Europe it might be very probable that, due to various reasons like physical node maintenance or in order to move workload to different locations, this VM might be migrated to a different location. In this case the information reported by the above variable will be outdated. The grid site administrators might not be even aware of this incident, thus they are not capable of updating these values manually. Thus, if site coordinates are still considered an important information in a virtualized environment, a viable alternative approach would be to use the location of the site administrator's home institute as the site locality.

3.1.2 Hardware information

CE_CPU_MODEL, CE_CPU_VENDOR, CE_CPU_SPEED, CE_OS_ARCH might not be known, may be difficult to define or finally may not be relevant at all in a cloud environment. In the case of VM migration and considering a scenario that a datacenter is built from non-homogenous physical nodes (different vendor, different architecture) these values may change and in some cases change frequently.

CE_MINPHYSMEM, CE_MINVIRTMEM, CE_PHYSCPU, CE_LOGCPU and CE_SMPSIZE assume a static, homogenous cluster where all the WNs share the same physical characteristics. Notice that this has been an issue also in the past with traditional grid cluster, but now with the cloud this assumption is even more restrictive since in the cloud the grid admin has the ability to customize very easily WNs with different hardware profiles. Latest versions of YAIM configuration support the glite-CLUSTER node type which allows a more fine grained separation of the physical resources. Still the nodes in a cluster are considered to be of the same type.

CE_PHYSCPU and CE_LOGCPU are difficult to be separated in a virtualized environment. In particular CE_PHYSCPU may not be easy to be defined at all since there is always a chance that a VM is not assigned a dedicated CPU but a subset of its time (e.g. 80%). Moreover the static definition of the above macros makes it very cumbersome to take advantage of the elasticity characteristics of the cloud. This values should be able to change on the fly whenever either the grid site administrator adds WNs by hand to the site or when an automated mechanism is used to adjust the size of the site adjusting to workload fluctuations and forecast resource requirements from grid jobs.

3.2 End user software

Currently grid sites follow a rather inelastic way of installing and advertising available software from site admins and VO managers. Users should be able to create their own WN VM images with their software pre-installed and attach it on-demand to an existing grid site that supports their VO. The StratusLab Marketplace could come into play in this scenario. Grid sites would act as endorsers of these VMs based on the specific VOs they support. This of course requires the establishment of the appropriate policies for VM endorsement on a EU-wide level. Again EGI could play an important coordination role in this area.

4 Operations

We consider four aspects of grid operations that are immediately impacted by the underlying cloud layer: the initial certification of the grid site, the related topics of monitoring and accounting, and the capability of grid site elasticity.

4.1 Site certification

Currently a site is certified through a formal process within the hosting NGI. It remains an open question who will be responsible for implementing the certification process in the case that the virtualized grid site is hosted in a cloud provider residing outside the grid site's NGI. Will it be the NGI of the cloud provider? Will it still be the responsibility of the grid site NGI? Will it be delegated to a centralized authority (a team within EGI)?

One other form of certification is the one required for issuing the digital certificates required by most grid services (e.g. CE, SE). Typically the grid site administrator will have to generate a certificate request for the service and email it to the Certification Authority responsible for his/her country. One of the requirements that the CA will check is that the domain name of the service lies within its area of authority (e.g. .gr for Greece). In the case of grid sites over clouds it is very probable that the cloud service might reside in a different country thus the allocated virtual machines will have a top level domain in a country different than the one in the area of responsibility of the CA. This will probably forbid the CA from signing the certificate. Obviously for this to work the CA policy has to be altered to allow signing of certificates for servers residing in foreign countries. Otherwise grid sites can take advantage of only same-country cloud providers.

Moreover, if we consider a federated cloud environment in which resource providers from different countries collaborate to provide cloud services, there is always a chance that part or all of the grid site might migrate to a different country or be split between two or more countries. Who will be the hosting NGI in this case? Will the digital certificates have to be re-issued from a different CA? Obviously, we have to reconsider to certification and monitoring procedures in order to take into account the characteristics of cloud environments.

4.2 Site monitoring

Currently information about a specific site is collected by a service running in the site itself and broadcast to a centralized service in EGI. These services are using LDAP to collect, organize and provide access to information. LDAP, by design, is a system optimized for infrequent updates and frequent queries. LDAP is not an adequate solution as has already been noticed in existing grid infrastructures, for example in updating the information about available CPUs, the total available storage, etc. In a cloud environment, this problem is amplified since a site may be structurally altered (virtually expanded or contracted) exploiting the elasticity and flexibility of the underlying cloud.

4.3 Accounting

Thus far, grid computing resources have been offered to scientists free of charge or at least with no direct charging, rather the costs where managed centrally by the government authorities. Consequently, the grid accounting system was designed mainly to support the collection of statistics and the centralized workload management systems of grid infrastructures. In the case of cloud e-Infrastructures it is not clear yet how the costs will be handled and who will be responsible for paying them. If we consider though a typical scenario where commercial cloud providers will offer resources to scientists or hybrid clouds where government funded clouds will burst to commercial clouds in order to handle peek workloads, it is crucial that the accounting system collects detailed usage information on the VO and individual user levels.

As mentioned, in our production site we have used glite-APEL[1] for site accounting. APEL collects only a limited amount of information such us the number of jobs submitted or total CPU time consumed per site/user/VO. Integration with cloud services will require a much more detailed report, including network bandwidth, storage space, IP addresses, and potentially use of software licenses (wherever applicable). On the other hand the cloud layer should be able to re-use this information in order to charge costs or/and to enforce quotas.

An alternative solution to APEL is DGAS (Distributed Grid Accounting System)[2] developed by INFN. DGAS offers the ability to collect accounting information for a broader range of metrics including Economic Accounting. According to our knowledge this last capability of DGAS has not been exploited so far by any of the production EGI sites. It may still be the case that this functionality will be useful in the cloud computing context and would be worth investigating in the context of StratusLab.

Finally, relevant work is also being carried within the VENUS-C project [4] which is working on a broad accounting solution that could cover grid service use cases and job workload execution models in general.

4.4 Site elasticity

Resource elasticity and flexibility is one of the most well known benefits that cloud computing brings to e-Infrastructures. Grid sites should be able to capitalize on this by being able to dynamically adjust their dimensions based on fluctuating demands. Typical dimensions of a grid site are:

- Processing capacity: being able to modify the size the cluster by adding or removing WNs on demand.
- Processing capability: being able to modify the profile of the WNs by adding more CPU cores, local storage and memory.
- Storage capacity: being able to modify the available storage provided by the SE node.

Currently StratusLab is working on grid site elasticity functionality. The idea is to integrate the Service Manager (Claudia) with the LRMS (Local Resource Manager System - e.g. Torque) in order to modify the size of the site based on rules defined by a grid administrator. For example:

- Increase the size of the site by 10% if the job queues become 80% full.
- Decrease processing capacity (remove WNs) by 20% if the utilization of the job queue falls below 20%.

This dynamic behavior of grid sites on the other hand may cause inconsistency on the global level if the information about the site's new capabilities are not announced promptly to the top level information systems (e.g. top-level BDII), causing job management services like the WMS (Workload Management System) to make inappropriate job scheduling decisions.

5 Summary and Conclusions

In this technical note we've presented our experience from the deployment of gLite grid sites on top of the StratusLab IaaS cloud. On first view, the provision of grid services on top of cloud technologies could be considered just another application for cloud services. Indeed, one can operate a grid site using virtualization technologies without any particular integration between the two layers. Nevertheless, we believe that in order for grid services to fully exploit the potential of cloud computing there should be a bridging of these two worlds on a technical level and operational level. In order to achieve this we've identified a number of issues both from the grid site and cloud service point of view, that pose important or less important impediments that forbid their optimal co-operation. Table 5.1 summarizes the issues identified, suggests potential solutions and pinpoints the actors involved.

Table 5.1: *Summary of issues and proposed solutions for grid/cloud interoperation*

Process	Issue Type	Description	Possible Solution	Applies to
Installation	Site installation and upgrade	Properly configured VM appliances are required for grid site deployment	Procedures for the creation and certification of grid node VMs must be defined as well as who is responsible for each action. These appliances should be updated with every grid software release.	Cloud layer (StratusLab), Grid middleware (EMI), Grid operations (EGI)
Configuration	Service Configuration	YAIM configuration files and tools are too static for the dynamic nature of clouds	Re-structure information that have to be defined in YAIM configuration files. Disregard those that can be provided automatically from the infrastructure layer (e.g. number of Worker Nodes in a grid site)	Grid middleware (EMI)
Configuration	Site locality	Geographical information defined by SITE_LAT and SITE_LONG macros might not be available to the grid site admin due to lack of knowledge about the location of the cloud datacenter.	Define as site locality the home institute of the grid site administrator	Grid operations (EGI)
Configuration	Hardware information	Low level hardware information like CPU type, vendor etc, might not be known, may be difficult to define or finally may not be relevant at all in a cloud environment.	Introduce the notion of Virtual CPU. Information regarding architecture (e.g. x86_64) and capabilities (e.g. GHz) could be potentially extracted from the VM instance upon startup.	Grid operations (EGI)

Process	Issue Type	Description	Possible Solution	Applies to
Configuration	Provision of end-user software	Currently grid sites follow a rather inelastic way of installing and advertising available software from site admins and VO managers	Users should be able to create their own WN VM images with their software pre-installed and attach it on-demand to an existing grid site that supports their VO. The StratusLab Marketplace could come into play in this scenario.	Grid operations (EGI)
Operations	Site certification	Who will be responsible for implementing the certification process in the case that the virtualized grid site is hosted in a cloud provider residing outside the grid sites NGI?	Could be performed by a central team in EGI. If workload too high, the NGI were the site admin resides could take over this responsibility	Grid operations (EGI, NGIs)
Operations	Site service digital certificates	Typically the grid site admin will have to generate a certificate request for the service and email it to the Certification Authority responsible for his/her country.	Certificate requests for virtualized sites might be handled by a catch-all cloud-oriented VO. Alternatively the CA of the country where the administrator resides should handle the process. In this case the CA should be able to sign requests for machines whose IP domain resides outside of the country.	Grid operations (EGI), EUGridPMA, IGTF.
Operations	Monitoring	Information are kept in a centralized LDAP data base. Apart from introducing a central point of failure LDAP is not appropriate for keeping information of constantly changing resources.	If aggregation of monitoring information is needed, implement a distributed monitoring solution. Replace LDAP with a more appropriate DB solution	Grid operations (EGI)

Process	Issue Type	Description	Possible Solution	Applies to
Operations	Accounting	APEL collects very limited information of grid site usage.	Need additional information to be kept (e.g. storage usage, network I/O, VM type used etc). These information should be reused on the cloud level in order to apply billing and quota policies.	Grid middleware (EGI), Cloud services (StratusLab)
Operations	Elasticity	Grid sites should be able to capitalize on this by being able to dynamically adjust their dimensions based on fluctuating demands. The grid information system should be able to cope with the frequent changes of grid site capabilities.	Requires close integration with the grid layer. A solution using OVF and the Service Manager (Claudia) is being developed in the context of StratusLab. On the grid layer	Grid middleware (EGI), Cloud services, (StratusLab)

References

[1] APEL. Accounting Processor for Event Logs. https://wiki.egi.eu/wiki/APEL.

[2] DGAS. Distributed Grid Accounting System. http://www.to.infn.it/dgas/.

[3] gLite Consortium. gLite Middleware. http://glite.org/.

[4] VENUS-C Project. VENUS-C. http://www.venus-c.eu/.

[5] YAIM. YAIM Ain't an Installation Manager. http://yaim.info.