PROCEEDINGS OF SCIENCE

Distributed Cloud Development for e-Science

Eric Yen [1]
1. *Academia Sinica Grid Computing Centre (ASGC)*
2. *Department of Management Information System, National Cheng-Chi University*
Taipei, Taiwan
E-mail: eric.yen@twgrid.org

Simon C. Lin
Institute of Physics, Academia Sinica
Taipei, Taiwan
E-mail: Simon.Lin@twgrid.org

The e-Science infrastructure is established to accelerate scientific discovery and innovation by integrating global resource federation and collaboration. The Worldwide LHC Computing Grid (WLCG) is the perfect example of advantages of immense scalability and multidisciplinary collaborations. Meanwhile, Cloud technology is expected to reduce the cost of exploiting research infrastructure and minimize the effort of data and IT management for the scientific community. In this research, ASGC integrated the service-oriented Cloud technology with the WLCG and built the ASGC Distributed Cloud so that wider range of services and more delicate customization for e-Science applications can be supported. For Infrastructure as a Service (IaaS), the site level virtual infrastructure management was established, together with the sharable repositories of virtual machine images and virtual appliances. And for Platform as a Service (PaaS) which enables easy reconfiguration of sharable services and tools according to different research workflows, ASGC developed the Grid Application Platform (GAP). It is composed of generic domain toolkits and interfaces for underlying distributed infrastructure and is also used in supporting platform level web portal development. Currently in ASGC Distributed Cloud, the on-demand services are available through thousands of virtual machines with the performance of reaching 98% of the physical machine when being used in running e-Science applications. ASGC will continue to enhance the Distributed Cloud for e-Science and work on decreasing the cost and effort of data management, elevating the performance, advancing the intelligent adaptive mechanism, and supporting the sustainability.

The International Symposium on Grids and Clouds (ISGC) 2012
Academia Sinica, Taipei, Taiwan
February 26 – March 2, 2012

[1] Speaker

I. Introduction

The Worldwide LHC Computing Grid (WLCG) is the largest production distributed system and has been supporting heavy usage of Large Hadron Collider (LHC) experiments and many e-Science applications, processing 1.3 million jobs a day [1, 2]. From the users' point of view, how to efficiently use the available resources for conducting big data analysis is always the main concern. Through the WLCG, users could make use of various resources,with the right data model, from hundreds of resource centers around the world over the Internet. The challenge today is how to enable the automation of data analysis, management and research workflow by taking advantage of new IT and Distributed Computing Infrastructure (DCI) like the grids and clouds.

When considering the service requirements from both users' and resource centers' end, Cloud technology enhances the service-oriented architecture, including infrastructure, platform, software and any specific layer valuable to the users as well as overall application design. Broadly speaking, both end user and the system administrator are all Cloud users. In terms of generic e-Science application requirements, scientific user community needs an environment where the research workflow is easily implemented and analysis is efficiently conducted. For resource center (and service provider), the flexibility of resource arrangement and agile system is always the target. In Table I, the requirements for these two user groups are categorized in the typical service layers of infrastructure, platform and software. Computing model on the DCI and data services are the fundamental challenges for the users. User environment and data flow services have to be supported at the platform layer. In addition, the workflow management and the repurpose of sharable software tools are the basic building blocks in the software layer. From the resource center's perspectives, providing on-demand service through virtualization and distributed resource federation is the first goal of DCI.

Requirement/Service		Infrastructure	Platform	Software
Scientific User	Complex Modeling & Simulation	HPC/HTC over Elastic scaling DCI	Efficient deployment & reconfigure of user environment	Workflow and SW tools repurposing
	Huge data analysis	Dist. storage management; fast data access	Data flow services; dist. data discovery & management	Configurable research applications; data repository
Resource Provider	Flexible resource arrangement on-demand	On-demand services; Site level VMM	Typical & customized app. Env. by virtual appliance	Application-specific services by Web portal
	Resource Fed	World Wide Grid	Dist. Services	Services API

Figure 1. Generic requirements of e-Science by Infrastructure, Platform and Software Layers

When examining all the above requirements, it is clear that the Distributed Cloud can maximize the advantages of both Grid infrastructure and Cloud services. Based on the e-Science development in the past decade, the distributed architecture used in data-intensice science is invaluable for future infrastructures. As scalability and sustainability being essential to e-Science, Distributed Cloud is the best solution to create a flexible infrastructure that is capable of serving massive amount of data and users anytime, anywhere. The target user communities, strategy and Distributed Cloud architecture are described in this section. Then the implementation approach is presented. The evaluation of real e-Science applications on Distributed Cloud is then discussed, followed by a discussion of the lessons learnt and related works.

II. The Implementation Approach

Distributed Cloud is a highly reconfigurable system composed of loosely coupled autonomous resource centers. Services can be dynamically scaled up or down and can be delivered according to key metrics such as the locality of the data and providers, etc. The service access barrier and the time needed to finish jobs without scalability limitation are the major concerns of an e-Science infrastructure.

ASGC Distributed Cloud is constructed based on WLCG infrastructure. In the service-oriented conceptual model, the user (consumer), service provider and resource broker are the primary actors in the typical Distributed Cloud scenario depicted in Figure 2. Information system provides required information in resource status, discovery, match making, virtual organization, application, and tracking of job lifecycle where static and dynamic information of the whole infrastructure could be found. Resource broker finds the best matched service providers for the submitted jobs according to the user preference, application requirements, and pre-defined service-level agreement (SLA) of the resources. Repositories of virtual machine (VM) images and virtual appliances (VA) serve as the authority control of shared VM images and appliances as well as the key to a fast user environment deployment at any site. A more delicate authentication and authorization mechanism could be imposed to the repositories for better security, collaboration or even commercial purposes.

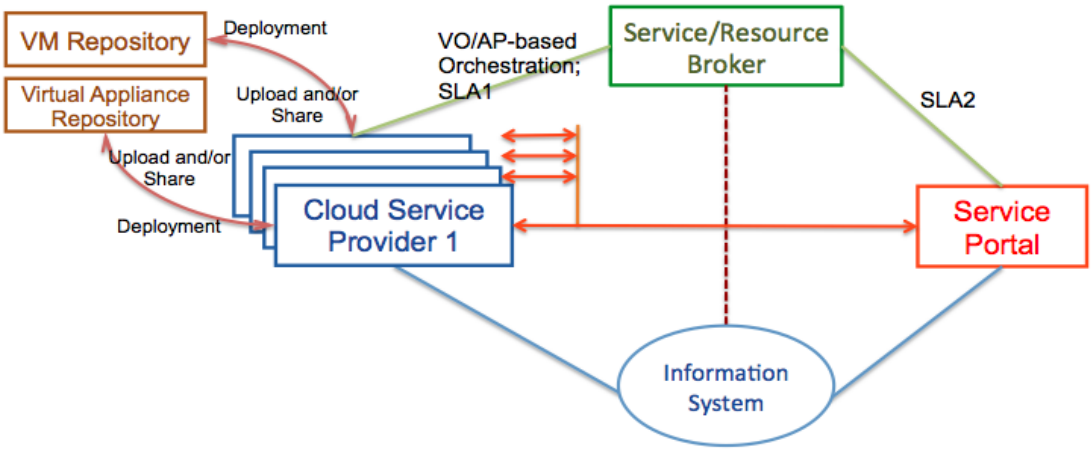

Figure 2. ASGC Distributed Cloud Model Based on User Requirements

WLCG Infrastructure contains basic grid middleware components, the federation of Internet resources, the support of distributed computing and data services. Pragmatic security and trust mechanism as well as well-defined information system are also in production. In terms of Cloud Computing, the only extra efforts are the site level virtualization, finer granularity of service layers, and the sharable VM and VA environment. Our approach is to integrate the site level virtual infrastructure with OpenNebula [3], utilizing the current gLite-based e-Infrastructure [4]. VM image repository is based on HEPiX Virtual Machine Image Catalog (VMIC) [5] to achieve trusted VM sharing across sites. CernVM [6] is used as a software appliance to support fast deployment of the application environment. The resource on-demand services and the reconfigurable storage services are the basic infrastructure level of Cloud services. Platform layer service is achieved by combining the pilot factory of PanDA, a distributed software system from ATLAS, and the Grid Application Platform (GAP2) developed by ASGC. The logical architecture of ASGC Distributed Cloud is as Figure 3.

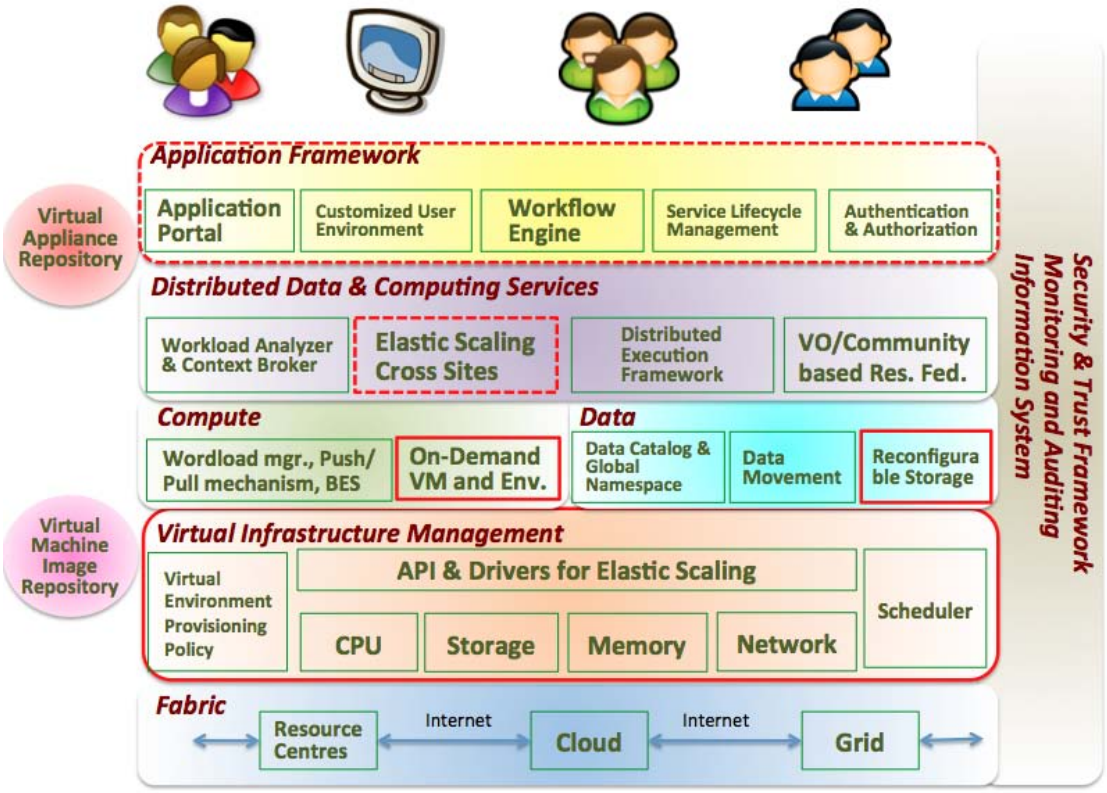

Figure 3. Logical Architecture of ASGC Distributed Cloud

WLCG laid the foundation of a scalable e-Science infrastructure. Cloud enforces service-oriented applications in all layers. Providing both IaaS and PaaS, ASGC Distributed Cloud is an adaptive system which combines the WLCG and Cloud for e-Science. Its architecture is illustrated as Figure 4.

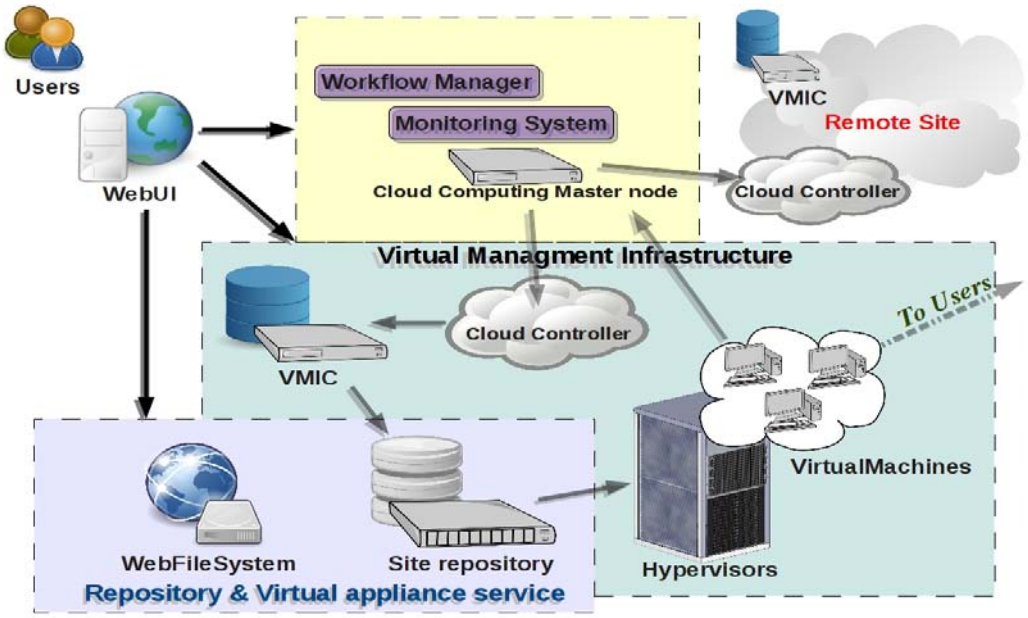

Figure 4. ASGC Site Implementation of Distributed Cloud Architecture

The site level virtual infrastructure management enables the Infrastructure as a Service (IaaS) to dynamically provide resources such as work nodes of gLite, system service or application server, or an independent computing cluster, etc. VMIC and VM repository would gurantee consistent system images in any environment. Linux hypervisor with KVM and Xen images in Qemu format is the basic template in current system for better interoperability. Disk Pool Manager (DPM) – the primary Grid storage management system of ASGC e-Infrastructure is integrated to serve as the storage of repository on Distributed Cloud. The CernVM for ATLAS, CMS, AMS and other e-Science applications is supported by the system and used for on-demand application platform services.. A general model of VM images and VA sharing and trading is under development, which is the foundation for Platform as a Service (PaaS) mechanism. Application environment could be easily deployed based on the VMIC and VA.

From the job processing workflow perspective, users access specific e-Science application or request resource through a web interface. Provided by IaaS, the on-demand resources that are implemented on vNodes include the Cloud VMs, gLite components, cluster computing environment as well as the pNFS file system.[7]. For e-Science applications, the online application portal is the typical solution for easy-to-use Grid User Interface and the platform services. Application workflow is implemented on the portal and jobs are processed in the job management framework. Both PanDA [8] and GAP Pilot job system are implemented to improve job brokerage efficiency and release user handling on job failure before execution finished successfully. Information System is vital for finding the best resources through matchmaking. Jobs could be executed at a remote site based on the result of the matchmaking or by the dynamic migration supplied by the underlying Grid infrastructure. Sample workflows for

applications such as the Grid Virtual Screening Service (GVSS) [9], BLAST,, and the Weather Simulation portal will be described in the next session.

Storage resource virtualization is more complicated than the computing resource as the storage space is rarely fully released once it is allocated. Also, it is challenging to efficiently aggregate fragmented space, especially in the block or file system devices. The focus of the discussion for the moment are the data sets and data transmission perspectives.

To tackle the two issues above, the storage system is designed as follows: first, the virtual storage block device could support protocols like iSCSI, Fibre Channel, and Fibre Channel over Ethernet. Then, for file-based storage, WebDAV, NFS/pNFS, CIFS and Lustre are supported by the DPM Grid storage. Currently, Dropbox-like file-based services are in operation under Web Distributed Authoring and Versioning (WebDAV) and x.509-based authentication. Virtual machines can easily get storage space through the pNFS file system. Thus, storage spaces could be shared among multiple servers through either block device or file-level access.

III. Platform As A Service (PaaS) for e-Science

In addition to IaaS, ASGC also provides PaaS. For many scientific users, having the platform services with required workflow to achieve data preparation, analysis, and visualization is more important than infrastructure services. In our design, each web portal constructed for individual e-Science application contains PaaS features including certificate upload and proxy delegation, data preparation and format conversion, analysis powered by the distributed grids and clouds, results visualization, and data management. In ASGC, common tools, library, application programming interface (API), and supporting packages are organized into a software framework – Grid Application Platform (GAP). GAP is designed to be a sharable e-Science application development framework which reduces cost and time. The latest version 2, GAP2, supports the pull mechanism based on lightweight pilot job framework. In the future, the data abstraction interface will be included on the Distributed Cloud platform to enhance customized services from complex analysis workflow to individual software tools.

With GAP, users or developers can focus their work on the application logic and required user interface by combining available software components and tools. GAP is now integrated with gLite Service Grid, Cloud and Desktop Grid. The complicated underlying DCI could be completely hidden from users. Changes of DCI technology could also be independently implemented so that the impact on the e-Science applications would be minimized.

In addition to the High Energy Physics data analysis for ATLAS, CMS, and AMS experiments, the Grid Virtual Screening Services (GVSS), BLAST and HMMER bioinformatics tools, Weather Research and Forecast (WRF) model had also been integrated into the current Distributed Cloud infrastructure.

GVSS is developed for large-scale drug screening. It vastly reduces the cost and time of the screening by utilizing the dynamic on-demand resources and services on distributed Grid and Cloud infrastructure. From data preparation, docking simulation, results analysis to visualization, all the services in the virtual screening process are supported by the GVSS web portal. The docking engine used in GVSS is Autodock, the most commonly used docking simulation application. Both Audock version 3 and version 4 are supported by GVSS now. GVSS also provides 700 proteins from Protein Data Bank and around 10 million drug compounds from ZINC database as the default data source. Users could choose the target protein file and ligand compounds directly from the embedded database with 3D visualization tools, or upload their own files because the data format conversion facility is also supported. Massive molecular docking could be processed faster and much more easily with GVSS. All the docking results are presented interactively with 2D or 3D visualization effects and can be downloaded for further analysis. Other analytical tools in GVSS include a docking energy ranking table, a histogram of (docking energy) result distribution, and 2D/ 3D principal component analysis. The workflow, screenshots and primary features are showed at Figure 5.

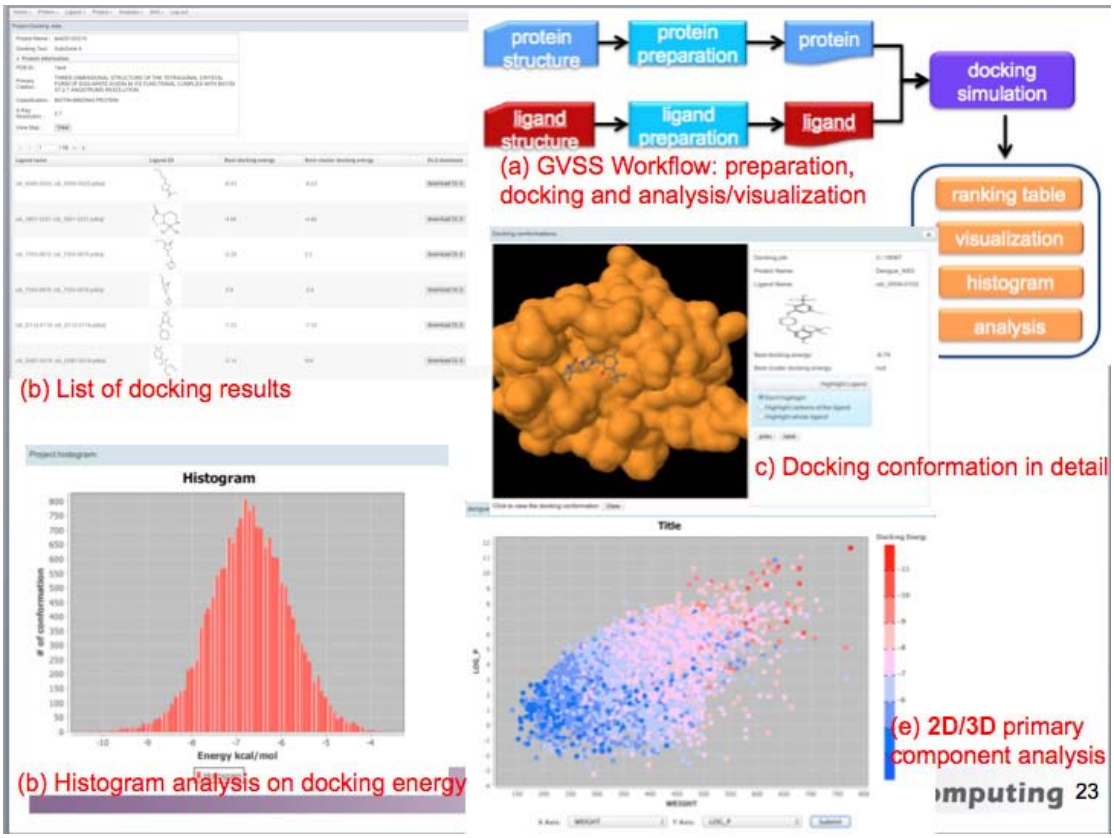

Figure 5. Features of Grid Virtual Screening Services

Other operating services include the BLAST, HMMER and Ensembl applications on gene sequencing analysis.

IV. Results and Analysis

In our current configuration, hundreds of blades (thousands of cores, 2GB RAM for each core in average) are allocated as dynamic resources for on-demand provisioning to VOs. Two basic types of performance evaluation were conducted in the ASGC Distributed Cloud environment – the virtual machine creation and the e-Science application running on dynamic resources.

Various scales of virtual machine creation test were conducted—from 32, 256, 575, 1000 to 1900 VMs at the same time. Default configuration is to have the Kernel-based Virtual Machine (KVM) hypervisor installed in each blade. OpenNebula was used as the VM manager. 3GB VM image in QEMU format was used as the default VM image. For each VM deployment process, OpenNebula scheduled the process, enabling the startup of image on hypervisor. In this evaluation, the priority is reliability rather than the maximal number of concurrent VM creation in order to minimize the failure rate. The VM creation process included the QEMU image instance generation, the kernel boot up and the operating system initialization. For 1900 VMs request, the current system was able to finish the provision in 577 seconds. For 1000 VMs, it took 330 seconds to startup all VMs. It took 117 seconds to do the same for 256 VMs and 60 seconds for 32 VMs. All test results could be found in Figure 6.

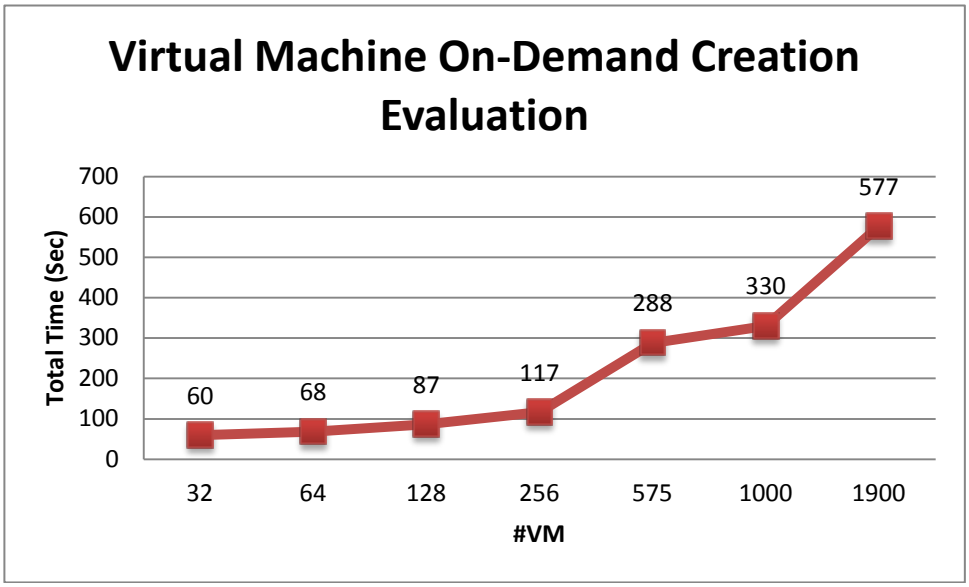

Figure 6. Results of Virtual Machine Creation Evaluation

The key to the VM creation performance is the capability of hypervisor to startup guest OS environment at a time. The VM creation failure rate would increase if too many requests are handled simultaneously. Hence it is crucial to identify the most stable configuration of the hypervisor as well as effective production of the virtual infrastructure. With a good scheduler mechanism, high-throughput large-scale VM creation could be achieved. We are confident to support on-demand resource provisioning with thousands of VMs in a reasonable amount of time. Detailed profiling of the performance bottleneck and the way to bridge the gap would be defined through more tests aforementioned.

In combination with real application, the single docking by Autodock version 3 and BLAST were tested on VM. In the test for Autodock version 3, a single docking could be done by 242 seconds, while the same process takes 238 seconds in a physical machine. It is known that compared to physical machines, there is overhead in virtual machines. Therefore, it is quite impressive that the test results indicate that our KVM performance has reached 98% of the physical machine performance.

BLAST is a memory intensive application. For BLAST, a small test case of 34 sequences with a 6GB reference database takes 19.95 seconds in a physical machine with 24 GB memory. In a VM with 1GB memory, the same application costs 367.4 seconds. When we increase the VM memory to 2GB, the execution time reduced to 72.6 seconds. The performance could be much better to be 20.55 seconds by 4 GB RAM, and even better in 20.01 seconds by 6 GB RAM.

V. Discussion

Providing on-demand resource is one of the primary purposes of Distributed Cloud. Based on the WLCG, cross-site infrastructure level services are enabled by establishing integrated site level virtual infrastructure. The trust framework, along with the virtual organization model, achieves the maximal scalability of distributed resource federation. Platform level services could be the desired services for complex data analysis, increasing data management efficiency on the cloud virtual infrastructure. From the successful e-Science applications in the past, the Distributed Computing Infrastructure proves to be the suitable infrastructure for processing a large quantity of data. Cloud technology is expected to provide the on-demand and intelligent services by swiftly reconfiguring shared tools, data, knowledge, infrastructure, and any kind of resources.

Distributed Cloud relies on the underlying Grid to access federated resources dynamically based on data availability, priority, and resource criteria. The WLCG information system has to reflect site elasticity on resources. A finer granular description on job requirements from the users is needed, for it is the criterion used in choosing the suitable site or creating the suitable execution environment. The system information schema has to include the dynamicity and virtualization of not just compute and storage resources but also the network (I/O quantitative metrics). The complete site information is of great help when automating the endpoint authorization process through OpenNebula.

ASGC has built the Distributed Cloud over WLCG and had it integrated with OpenNebula, VM Image and virtual appliance repository, and web portal for e-Science. On-demand and dynamic resources provided by thousands of VMs are available to high-energy physics, life science, weather simulation, and generic cluster computing applications. Efforts have been invested in identifing the performance metrics and bottleneck of the e-Science applications so that the findings, key performance parameters, can be used to establish the intelligent reconfiguration.

A common issue concerns Cloud Computing is the Message Passing Interface (MPI) applications on distributed cloud. In a virtual cluster established independently on physical machines without other applications, the situation is much more straightforward—just like a typical computing cluster. The application performance varies, depending on the loading and characteristics of the coexistent VMs on the same hypervisor. Distributed Cloud provides smooth job migration and checkpoint support through virtualization technology. However, for the dynamic provisioning in the MPI application environment, it requires much more sophisticated definition of the resource configuration in order to achieve satisfying performances. Definitively identifying the software environment, process constraint (e.g. number of threads in a node), I/O arrangement from interconnection bandwidth, file system and data locality are only the first steps to the efficient execution of this kind of jobs. ASGC continues to work on providing the dynamic work nodes for ATLAS/CMS, the on-demand resource provisioning for AMS, and the enhancement of MPI environment configuration in Distributed Cloud. More efforts are required in working on the MPI performance issue and further intelligent adaptive mechanism.

With the e-Science infrastructure, Distributed Cloud demonstrated the ability of accelerating the science discovery and innovation in multiple disciplines. In the Big Data era, ASGC will continue to enhance the Distributed Cloud by shortening the time-consuming data management and data computation process and supporting data sustainability.

VI. Acknowledgement

We thank Felix Lee and Cloud team members of the Academia Sinica Grid Computing Centre in building the distributed cloud prototype, without which this work would not have been possible.

References

1. The Worldwide LHC Computing Grid, http://wlcg.web.cern.ch
2. Mathieu, G. and Richards, A., et. al, GOCDB, a Topology Repository for a Worldwide Grid Infrastructure, Journal of Physics: Conference Series 219 (2010).
3. OpenNebula, http://opennebula.org
4. gLite – Lightweight Middleware for Grid Computing, http://www.glite.org
5. Synge, O., HEPiX Virtualualization Working Group: 2011-04-01.
6. Buncic, P., and Sanchez, C. et. al., CernVM – a Virtual Software Appliance for LHC Applications, Journal of Physics: Conference Series 219 (2010).
7. vNode – Virtual Node On Demand, http://vnode.web.cern.ch
8. Nilsson, P. and Caballero, J. et. al., The ATLAS PanDA Pilot in Operation, Journal of Physics: Conference Series 331 (2011).
9. HY. Chen, M. Hsiung, HC. Lee, E. Yen, S.C. Lin, and YT. Wu (2010), GVSS: A High Throughput Drug Discovery Service of Avian Flu and Dengue Fever for EGEE and EUAsiaGrid, Journal of Grid Computing, 2010.