A Survery of Open-Source Cloud Infrastructure using FutureGrid Testbed

Tak-Lon Wu, Shakeela Noohir Baasha, Sonali Surendra Karwa
School of Informatics and Computing
Indiana University, Bloomington, IN
{taklwu, shaknooh, skarwa@indiana.edu}

Abstract

Cloud Infrastructure Management is now an emerging problem in Cloud Computing. Open Source Projects such as Nimbus, Eucalyptus and OpenNebula provide the great opportunity for industries, researchers and scientists to understand the modern Cloud Infrastructure model. Here, our project aims to survey among these three open source Cloud Infrastructure Management tools and provides detail taxonomy among them.

1. Introduction

In the past few years, data-center supercomputer model has become very popular on the fly with the growing, emerging computing phenomenon, Cloud Computing. Within this new computing generation, several new opportunities can be provided as "X as a Service" (XaaS), where X are Application, Platform, or Infrastructure. Users can utilize more and own more on-demand compute resources, which they do not need to suffer from the expensive cost of supercomputer hardware, time limitation of using a big cluster sharing with other users, unexpected task termination due to higher priority task interrupted to the job queue, and other uncontrollable situations.

Among those XaaS model, IaaS, which provides dynamic resource provisioning, is the fundamental level and is now an emerging problem in Cloud Computing [1]. Commercial companies such as Amazon Cloud [2], Rackspace [3], and Salesforce.com [4] has successfully utilized this model and built up their commercial clouds to provide on-demand XaaS to public users. On the other hand, academic organizations also deploy the open source version of those cloud infrastructure technologies to construct their private Clouds for academics use. However, to build such private cloud, it is not obvious to decide a specific technology for their projects without a deep understanding or a clear comparison among those management tools. Here, this student project aims to focus on the three modern Cloud Infrastructures such as Nimbus [5], Eucalyptus [6], OpenNebula [7] and presents a detail implementation report, survey and taxonomy (initial result shown in Figure 2). Besides, we run real MapReduce applications such as HadoopWordCount HadoopBlast, and TwisterKmeans on FutureGrid Testbed [8], which has those technologies installed, to analyze their workflows and behaviors.

2. FuturdGrid Testbed

FutureGrid [8] is an experimental, high performance Grid and Cloud Testbed founded by NSF. It aims to provide researchers a real environment to understand the features, functionalities, and utilities of different Grid or Cloud technologies; also, it allows other scientists to test their applications on it. Ongoing topics such as virtualization, cloud infrastructure management, scheduling, cybersecurity, bioinformatics, and other emerging research issues are investigated by various groups of academic organizations.

Here, our project focuses on making a survey among three open-source cloud infrastructure management tools, Nimbus, Eucalyptus, and OpenNebula which are currently deployed on FutureGrid.

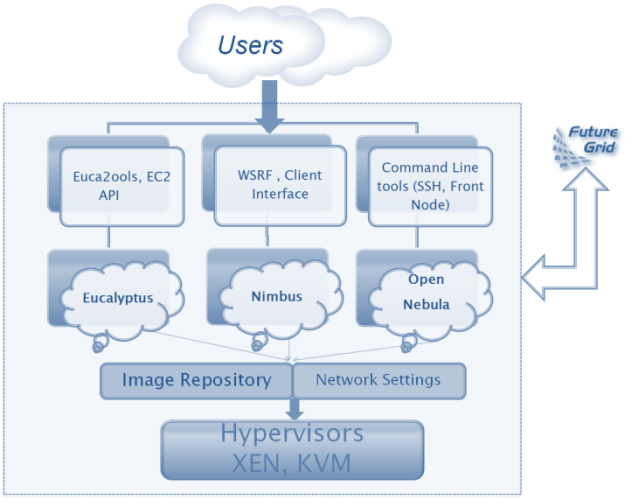

Fig. 1. System Architecture of FutureGrid Testbed

2.1 FutureGrid Hardware

According to the FutureGrid official website, there are six service sites across the nations. For our project, we utilize IU-India (India) and UC-Hotel (Hotel). Both of systems are built up with IBM iDataPlex machine. Each compute node of the systems has two Intel Xeon 5570

quad-core CPUs running at 2.93 GHz, 24 GB of DDR2 Ram, and dual data rate Infiniband 20Gbps. India has a total of 1024 cores, 3072 GB ram, and 335 TB of secondary storage in, on the other side, Hotel has a total of 672 cores, 2016 GB ram, and 120 TB of secondary storage.

2.2 FutureGrid Infrastructure Deployment

Eucalyptus and OpenNebula has been setup in India, and Nimbus is serving at Hotel. Noted that OpenNebula is still working in progress, and regarding to the FutureGrid team, this service become available to public on February 2011. In India, currently, they provide 39.1% (400 cores) capacity of the system to run Eucalyptus, meanwhile, in Hotel, 65.6% (336 cores) of the system can be invoked as Nimbus instances.

3. Implementation Environment

In spite of experimenting on the existing FutureGrid deployment, we also implement each of these tools on a small test environment in order to understand the environmental requirements. Two Ubuntu LTS 10.04 machines have been deployed; one of them has installed the UEC packages as required by Eucalyptus. Since OpenNebula is not officially released, most of the features of OpenNebula are captured within the test environment.

4. Poster

In the submitted poster, it will include a system architecture of Nimbus, Eucalyptus, OpenNebula deployed on FutureGrid as shown in Figure 1, a detail taxonomy table as shown in Figure 2, and finally a performance graph of three Hadoop/Twister applications running on FutureGrid as shown in Figure 3.

Features	OpenNebula	Eucalyptus	Nimbus
Hypervisor Support	KVM, Xen, VMware	Xen hypervisor, KVM, VMware	KVM, Xen
Provision and utilization	On-demand, Dynamic	Dynamic Provisioning, On-demand	Dynamic Provisioning Environment using Workspace Service
Guest OS Support	Linux	Linux	Linux
Image repository	Shared File System	Walrus (Storage Controller)	Cumulus repository
Network configuration	DHCP, fixed, Variable	HTTP, SOAP. Virtual private network per cloud allocation	External :Public IPs or Private IPs. Internal: private network via a local clustern/w
Security	User logs into head node, (unless open port in Frond-End node)	WS Security for authentication, Cloud Controller produce the public/private key	GSI Authentication and Authorization: PKI ,works with grid proxies VOMS, shibboleth (via GridShib) custom PDPs) Secure access to VMs: EC2 key generation or accessed from .ssh
Fault tolerance	Persistent database backend to store hosts, networks and virtual machines information	Correlated failures can be reduced by separate clusters	Checking worker Nodes periodically and recovery
Public/private and hybrid	all	all	Private and Shared clouds
GUI	Unix-like command line interface	Amazon AWS interface.	Nimbus Web is the evolving web interface for Nimbus. Its aim is to provide administrative and user functions in a friendly interface
Unique Features	VM migration support	User Management Web interface	Nimbus Context Broker

Fig. 2. Initial Taxonomy of Nimbus, Eucalyptus, and OpenNebula

Acknowledgement

Information about FutureGrid Testbed was obtained originally from FutureGrid website and technical team. Any findings, opinions and conclusions expressed in this material are those of the author(s) and do not reflect the views of the National Science Foundation (NSF) under Grant No. 0910812 to Indiana University for ”FutureGrid: An Experimental, High-Performance Grid Test-bed.”

We, here, would like to express our special thanks to Professor Judy Qiu, Dr. Gregor von Laszewski, Greg Pike, Andrew J. Younge, and the FutureGrid technical team for their advices and support.

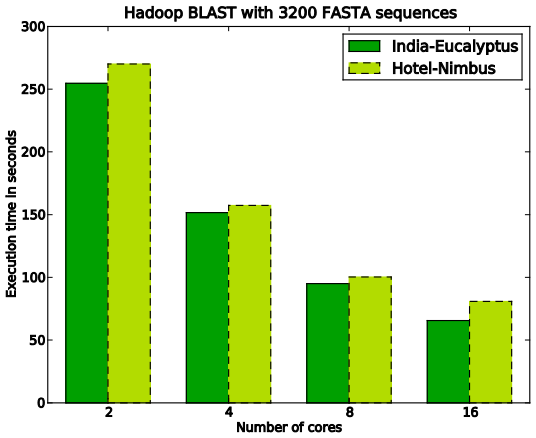

Fig. 3. Performance of Real MapReduce Applications (HadoopBlast)

References

[1] Borja Sotomayor, Ruben S. Montero, Ignacio M. Llorente, Ian Foster, "Virtual Infrastructure Management in Private and Hybrid Clouds," IEEE Internet Computing, pp. 14-22, September/October, 2009

[2] Amazon AWS Cloud, http://aws.amazon.com/

[3] Rackspace.com, http://www.rackspace.com/

[4] Salesforce.com, http://www.salesforce.com/

[5] Nimbus Project, http://www.nimbusproject.org/

[6] OpenNebula Project, http://opennebula.org/

[7] Eucalyputs Project, http://open.eucalyptus.com/

[8] FutureGrid: An Experimental, High-Performance Grid Test-bed, http://www.futuregrid.org/