An Effective C, C++, PHP, Perl, Ruby, Python Compiler using Cloud Computing

M.Pabitha
Dept of CSE
Dr.Mahalingam College of Engineering & Technology
Pollachi, Tamilnadu, India.

T.Selvakumar
Networking Dept
Aricent Group
Chennai, Tamilnadu India

S.Punitha Devi
Dept of CSE
P. A College of Engineering & Technology
Pollachi, Tamilnadu, India

ABSTRACT

Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and unconfined with minimal management effort. Our work mainly targets the learning community by providing an efficient cloud compiler as SaaS by coalescing the two major concepts called cloud computing and open source which helps to diminish the troubles of portability, compatibility, power and storage space by making use of the concept of cloud compiler. The basic underlying architecture to deploy a cloud compiler is the establishment of private cloud under linux environment, which provides hosted services to a limited number of people and the service is distributed in the heterogeneous manner. Private cloud makes the cloud infrastructure based on Ubuntu Enterprise Cloud (UEC) scalable as per the requirement. And our cloud compiler allows a programmer to pick up the fastest or the most convenient tool to compile the code and remove the errors. The validity of our approach is then verified with the experimental results. Also we have made a performance analysis and the experimental results shows that the performance of cloud compiler is more efficient compared to all other normal compiler. Hence our proposed cloud compiler is considered to be the best performers among the various compilers.

General Terms - Cloud computing, Cloud compiler, Open source

Keywords - Private cloud, SaaS, Cloud setup, Cloud controller, Ubuntu Enterprise Cloud (UEC)

1. INTRODUCTION

Cloud computing portends a major change in how we store information and run applications. Instead of running programs and data on an individual desktop computer, everything is hosted in the cloud, a nebulous assemblage of computers and servers accesses via the internet. Cloud computing lets to access all your applications and documents from anywhere in the world, freeing you from the confines of the desktop and making it easier for group members in different locations to collaborate. The emergence of cloud computing is the computing equivalent of the electricity revolution of a century ago. Before the advent of electrical utilities, every farm and business produced its own electricity from freestanding generators. After the electrical grid was created, farms and businesses shut down their generators and bought electricity from the utilities, at a much lower price and with much reliability than they could produce on their own. Look for the same type of revolution to occur as cloud computing takes hold. The desktop-centric notion of computing that we hold today is bound to fall by the wayside as we come to expect the universal access, 24/7 reliability, and ubiquitous collaboration promised by cloud computing. Cloud computing isn't network computing. With network computing applications/documents are hosted on a single company's server and accessed over the company's network. Cloud computing is a lot bigger than that. It encompasses multiple companies, multiple servers, and multiple networks. Plus unlike network computing, cloud services and storage are accessible from anywhere in the world over an internet connection. Cloud computing builds on established trends for driving the cost out of the delivery of services while increasing the speed and agility with which services are deployed. It shortens the time from sketching out application architecture to actual deployment. The concept of cloud services development encompasses several different types of development like Software as a Service (SaaS), Platform as a Service (PaaS), on-demand computing. Our cloud compiler focuses on providing as SaaS. With SaaS, a single application is delivered to thousands of users from the vendor's servers. Customers don't pay for owning the software; rather they pay for using it. The recent upswing in technology and increasing concern related to portability, compatibility, storage space, performance caused a boost in providing an efficient cloud compiler for languages like C, C++, PHP, Perl, Ruby, Python based on cloud computing.

Cloud Compiler is a family of cloud compiler licensed programs for the open source operating environments. A cloud compiler is a program that functions equivalently to an actual compiler but does not require that the actual compiler be installed or licensed on the machine on which it runs. The cloud compiler utilizes web service to transmit the user's source code to another cloud which the actual compiler is installed, compile it there and generate the output in the cloud and send the output to the user via the web services if the source program is correctly given otherwise through errors. Most of the options and features of an actual compiler are supported. The Cloud Compilers are not difficult to install. The primary tasks are the typing the programs in the client machine and send to the cloud on which compiles will be submitted [2]. Some interesting results indicating the performance of the cloud compiler are also presented. The remainder of this paper is organized as follows. In Sect. 2 compare the cloud computing with other related technologies. In Sect. 3, we describe the system framework of the open source cloud compiler and present its system design principles. The cloud compiler deployment is detailed in Sect. 4. Section 5 summarizes the experimental results. Finally, in Sect 6, we conclude with our best findings and future plans.

2. PREVIOUS WORK

Research based on cloud compiler and open source is very much limited, related to the Ubuntu Enterprise Cloud (UEC). Some of the important works are enumerated below:

Authors of [4] depicted that Private clouds can be built and managed by a company's own IT organization or by a cloud provider. In this hosted private model, a company such as Sun can install, configure, and operate the infrastructure to support a private cloud within a company's enterprise data centre. This model gives companies a high level of control over the use of cloud resources while bringing in the expertise needed to establish and operate the environment.

Authors in [10] portrayed Cloud computing implies a service oriented architecture, reduced information technology overhead for the end-user, great flexibility, reduced total cost of ownership and on demand services among other advantages. The National Institute of Standards and Technology (NIST) define Cloud Computing as a model for enabling easy, on-demand network access to a shared pool of configurable computing resources.

Google File System (GFS) [12] is a proprietary distributed file system developed by Google and specially designed to provide efficient, reliable access to data using large clusters of commodity servers. Files are divided into chunks of 64 megabytes, and are usually appended to or read and only extremely rarely overwritten or shrunk. Compared with traditional file systems, GFS is designed and optimized to run on data centers to provide extremely high data throughputs, low latency and survive individual server failures

Shuai Zhang et al. [7] also introduces the application field the merit of cloud computing, such as, it do not need user‘s high level equipment, so it reduces the user‘s cost. It provides secure and dependable data storage center, so user needn‘t do the awful things such storing data and killing virus, this kind of task can be done by professionals.

Borjasotomayor et al. [8] uses Open Nebula as a middleware that deploy and manage VMs, either individually or in groups that must be Co scheduled on local resources or external public clouds. It automates VM setup (preparing disk images, setting up networking, and so on) regardless of the underlying virtualization layer (Xen, KVM, or VMware are currently supported) or external cloud (EC2 or Elastic Hosts are currently supported).

Authors in [14] depicted the characteristics of cloud computing are much more complex in which can be used to distinguish cluster, grid and cloud computing systems. Cluster‘s resources are located in single administrative domain with single entity. Resources of grid system are distributed and located in administrative domain with multi entity and management policies.

Authors in [7] viewed the Cloud computing two different aspects. One is about the cloud infrastructure which is the building block for the up layer cloud application. The other is of course the cloud infrastructure. By means of several technical methods, cloud computing has achieved two important goals for the distributed computing: high scalability and high availability. SaaS provides Internet application to the customer also provides the software the off-line operation and the local data storage, lets software and service which the user all may use it anytime and anywhere to order. These studies endure from the secure and dependable data storage by reducing technology overhead for the end-user. Some of the studies also achieved high scalability and high availability, also provides reliable access to data using large clusters. Some studies suffer from problems like usage of simplified assumptions with no investigation into the validity of those assumptions; the results of evaluation may hinge on the validity of assumptions. Thus, despite the progress that has been made in this field, there remains a clear need for large-scale empirical comparison of cloud infrastructure by utilizing the open source techniques. There is also a need for guidance on the implementation and selection of techniques.

3. SYSTEM DESIGN

The overall design process involved in the creation of the private cloud infrastructure by utilizing the open source Linux servers, open source eucalyptus packages and the design of the cloud compiler which is deployed over the private cloud infrastructure.

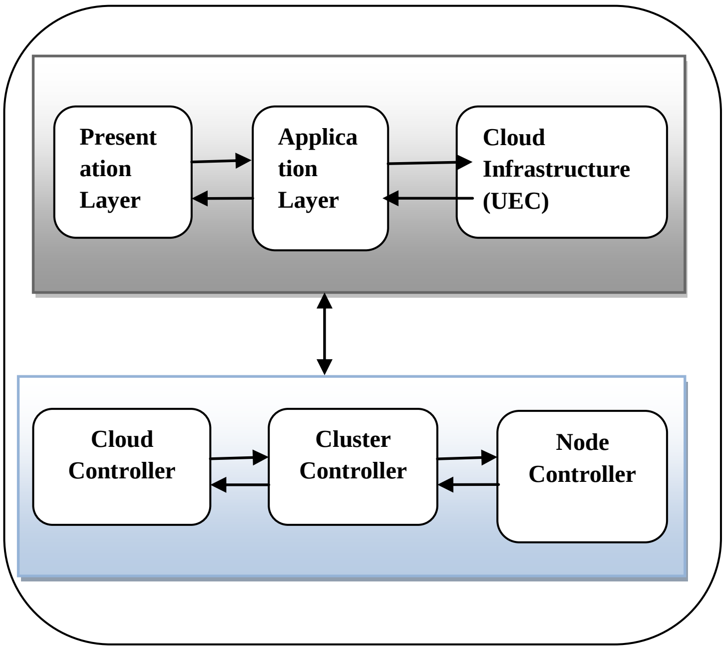


Fig 1 System framework of the Open source Cloud compiler

The contributions of our research work are summarized following: The Cloud Controller (CC) is the front end to the entire cloud infrastructure. CC monitors the availability of resources on various components of the cloud infrastructure, including hypervisor nodes that are used to actually provision the instances, also provides an EC2/S3 compliant web services interface to the client tools on one side and interacts with the rest of the components of the Eucalyptus infrastructure on the other side. The installation steps in the front end machine are as follows:

Step 1: Use the Ubuntu server 10.04

Step 2: When boot, select Install Ubuntu Enterprise Cloud.

The Cluster Controller (CLC) is the middle end to the entire cloud infrastructure. It communicates with Cloud Controller (CC) on one side and NCs on the other side. CLC handles one or more Node Controllers and deploys/manages instances on them, and also to control the virtual network under the modes of Eucalyptus. A UEC node is a VT enabled server capable of running KVM as the hypervisor. UEC automatically installs KVM when the user chooses to install the UEC node. The VMs running on the hypervisor and controlled by UEC are called instances. Eucalyptus supports other hypervisors like Xen apart from KVM, but Canonical has chosen KVM as the preferred hypervisor for UEC. Node Controller runs on each node and controls the life cycle of instances running on the node.

The NC interacts with the Operating System and the hypervisor running on the node on one side and the Cluster Controller (CLC) on the other side. It also helps to collect the data related to the resource availability and utilization on the node and reporting the data to CLC. The installation steps in the node machine are as follows:

Step 1: Connecting to the network on which the cloud/cluster controller is already running.

(i) Boot from the same ISO on the node(s).

(ii)Select Install Ubuntu Enterprise Cloud

(iii) Detection of the Clusters presents in it and preselect the node

(iv) Confirmation of the partitioning scheme

Step 2: Registration of the Node(s). As of Ubuntu 10.04 LTS, all component registration should be automatic, assuming:

(i) Exchange of Public SSH keys have been exchanged properly

(ii)Configuration of services

(iii)The services are publishing their existence

(iv)Running the appropriate uec-component-listener

(v)Verify Registration.

Step 3: Attaining the credentials

(i) Installing and booting the Cloud Controller, users of the cloud will need to retrieve their credentials. This can be done either through a web browser, or at the command line

4. CLOUD COMPILER DEPLOYMEMT IN THE CLOUD MACHINE

The cloud compiler is hosted on the web server in the Linux cloud instances then accessed via web browser.

Step 1: The cloud instances is started in the node machine

Step 2: The compiler for the various languages such as C, C++, PHP, PERL, PYTHON, and RUBY is developed as the SAAS web application and implemented on the web server in the cloud machine.

Step 3: The compiler application will be available as SaaS and can accessible within the Private Infrastructure

The proposed execution worked on a real networked environment with 2 systems which are acted as the front end and node machine. This setup used to make the private cloud setup. The cloud controller and cluster controller is running on the front end machine and the node controller is running on the second node machine. Then the compiler application is deployed over the private cloud to serve the users. Finally the performance is calculated among the cloud compiler and the various compilers to compare the efficiency.

Table 1. Various Compilers Input and Output

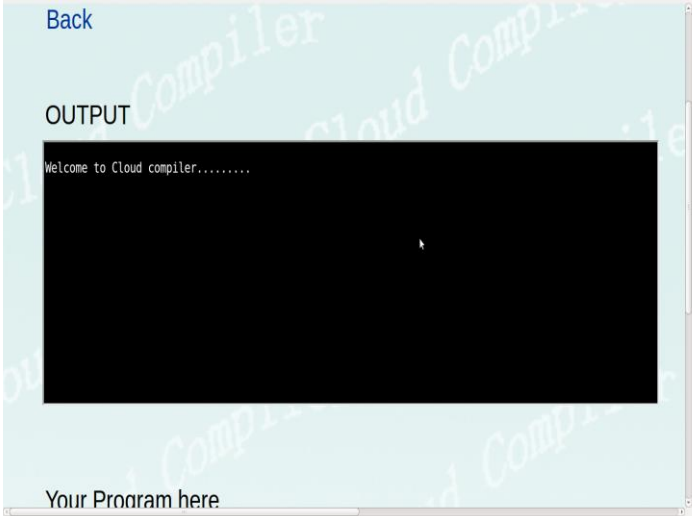
S. No	COMPILER	INPUT	OUT PUT

1	C Compiler	# include <stdio.h>
		Void main ()
		{
		printf(“ Welcome to Cloud Compiler “);
		}
			Welcome to Cloud Compiler

2	CPP Compiler	# include <iostream.h>
		using namespace std;
		int main ()
		{
		cout<<“ Welcome World”<<”\n”;
		cout<<“ I am C++”:
		return(0);
		}
			Welcome World
			I am C++

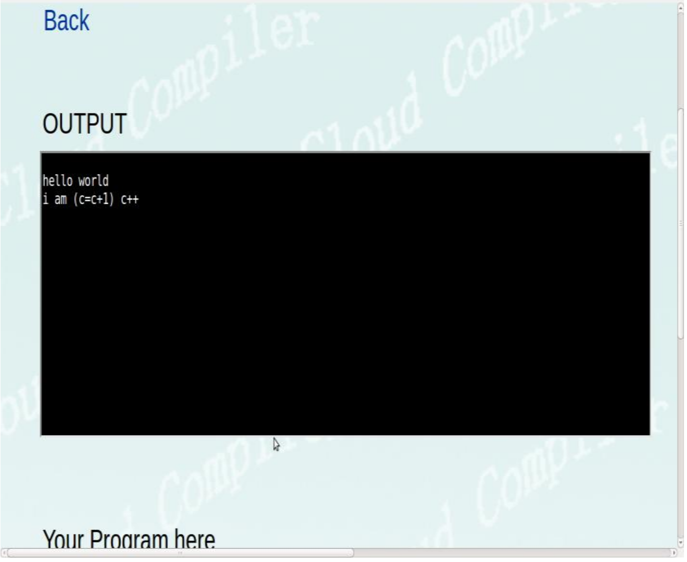
3	PHP Compiler	<?php echo ‘Hello World’; echo “\n”; echo ‘ I am PHP’; ?>	Hello World I am PHP
4	PYTHON Compiler	print “ Welcome World” print “ I am PYTHON”	Welcome World I am PYTHON
5	PERL Compiler	print “ Welcome World” print “ I am PERL”	Welcome World I am PERL
6	RUBY Compiler	puts “ Welcome World” puts “ I am RUBY”	Welcome World I am RUBY

The cloud compiler is hosted on the web server in the linux cloud instances then accessed via web browser. The following windows evidence the results of the proposed system.



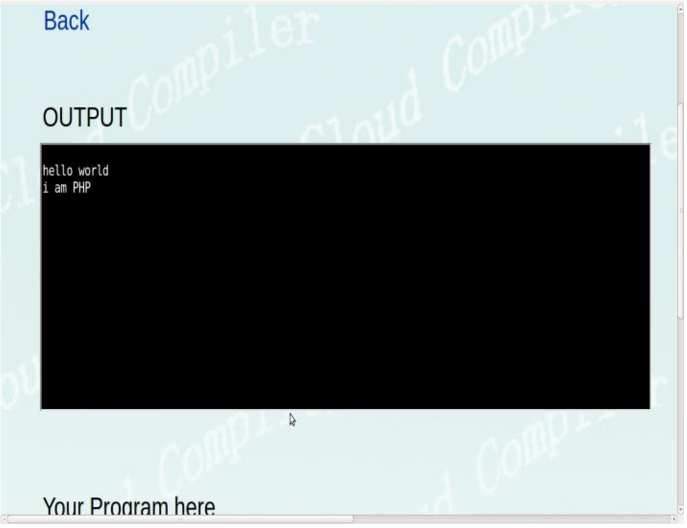
```
Welcome to Cloud compiler.........
```

Fig 2: C compiler output window



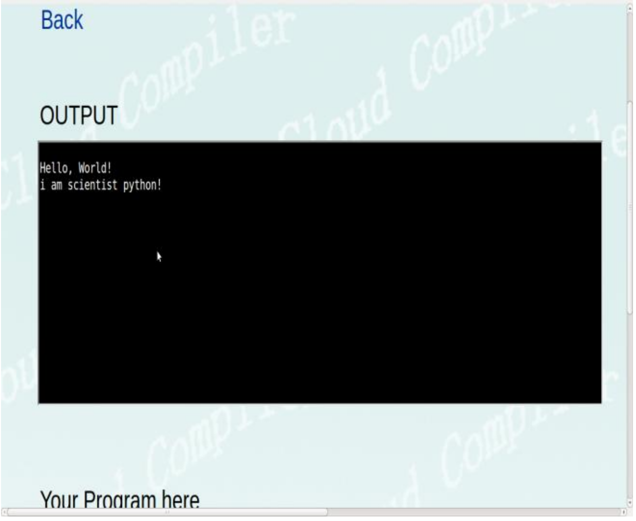
```
hello world
i am (c=c+1) c++
```

Fig 3:C++ Compiler Output window



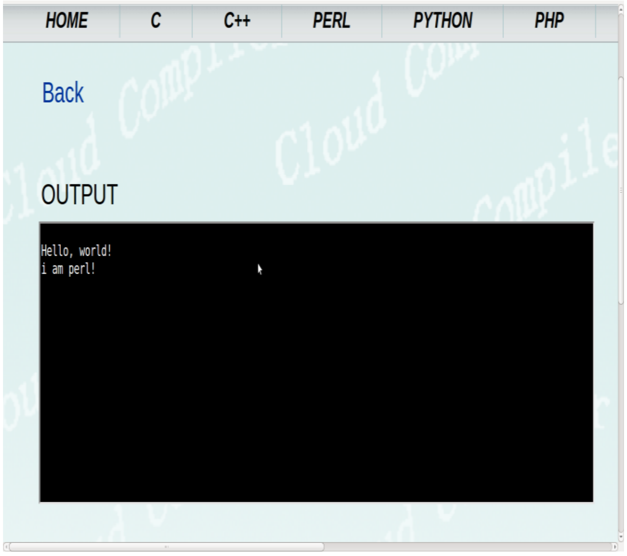
```
hello world
i am PHP
```

Fig 4: PHP Compiler Output Window



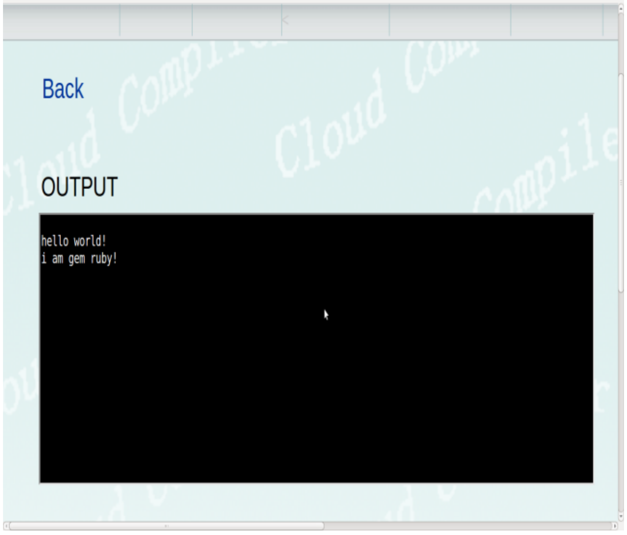
```
Hello, World!
i am scientist python!
```

Fig 5: Python Compiler Output Window



```
Hello, world!
i am perl!
```

Fig 6: PERL Compiler Output Window



```
hello world!
i am gem ruby!
```

Fig 7: Ruby compiler Output Window

The portability, compatibility, storage space, performance caused a boost in providing an efficient compiler for languages like C, C++, PHP, Perl, Ruby, Python based on cloud computing. Some interesting results indicating the performance of the cloud compiler are also presented. This cloud compiler is used to execute the Programming languages like C, C++, PHP, Perl, Ruby, Python via web interface it is get the service from the gcc compiler installed in cloud. It can be able to execute all types of program similar to the gnu c compiler

5. EXPERIMENTAL RESULTS AND DISCUSSIONS

This private cloud setup and cloud compiler experiment shows that the performance of the cloud compiler is relatively high compare with some other compilers. The result of the performance graph is explained in this chapter. The private cloud setup experiment and cloud compiler experiment have successfully implemented and the performance is calculated for the cloud compiler from various machines that shows that the cloud infrastructure uses the available resource effectively and serves the learning platform in the learning environment. As well as the performance graph shows that cloud compiler performance is superior. The execution time for various compiler by executing the Fibonacci series using recursion in the various compilers.

Table 2. Performance Calculation

N value	TurboC(in sec)	Cloud compiler(in sec)
43	35.8242	27.65
44	57.5275	44.82
45	91.7582	72.7522
46	146.044	117.76117
47	232.967	190.8208
48	373.956	305.81
49	606.099	502.7516
50	989.3945	810.901031

Where N is the number of series generated in the Fibonacci series (numeric value). After calculating the runtime of the cloud compiler and other TURBO C compiler the graph is plotted.

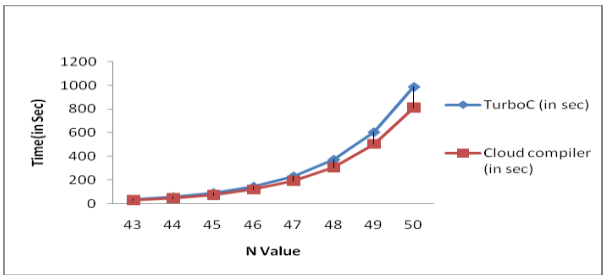


Fig 8 Comparison of Performance obtained for different compilers

The following graph shows that the efficiency of the cloud compiler is high. As it can be seen from the results, the cloud compiler gives the best results in terms of execution time. This compiler overrides all the other compilers and gives the best performance.

6. CONCLUSION

The proposed system has implemented an effective cloud compiler for languages like C, C++, Perl, Python, Php and Ruby. This paper depicts a clear view of how a private cloud based on UEC under linux environment can be setup by deploying a cloud compiler, and also examines the effect of different compilers on the performance of runtime. We have demonstrated that, our proposed cloud compiler reduces the run time when compared with all other compilers. And also measured to the current scenarios, where each machine needs to install compilers separately. But our cloud compiler will eliminate the need to install the compilers separately, thus it helps for a programmer to pick up the fastest or the most convenient tool to compile the code and remove the errors at the centralized server. To the best of our knowledge, it is the first work that applies cloud compiler to enhance the performance and we compare the performance of our cloud compiler with the various compilers. Our performance graph shows that our cloud compiler gives better performance and based on some comparative results that we also obtained, we concluded our proposed cloud compiler is considered to be the best performers among the various compilers.

7. REFERENCES

[1] Abramson D., Buyya R., and Giddy J. “A computational economy for grid computing and its implementation” in the Nimrod-G resource broker.

[2] F. Bellard. QEMU, a Fast and Portable Dynamic Translator. Proceedings of the USENIX Annual Technical Conference, FREENIX Track.

[3] Bo Dong, Qinghua Zheng, Jie Yang, Haifei Li, Mu Qiao,MOE KLINNS Lab and SKLMS Lab, Xi'an Jiaotong University, 710049, China, An E-learning Ecosystem Based on Cloud Computing Infrastructure 2009 Ninth IEEE International Conference on Advanced Learning Technologies.

[4] B ECHTOLSHEIM , Cloud Computing and Cloud Networking talk at UC Berkeley, December 2008.

[5] Eucalyptus Beginner’s Guide – UEC, Edition (Ubuntu Server 10.04 - Lucid Lynx), v1.0, 25 May 2010.

[6] Daniel Nurmi, Rich Wolski, Chris Grzegorczyk, Graziano Obertelli, Sunil Soman, Lamia Youseff, Dmitrii Zagorodnov, The Eucalyptus Open-source Cloud-computing System.

[7] Shuai Zhang Shufen Zhang Xuebin Chen Xiuzhen Huo, Cloud Computing Research and development Trend, Future Networks, 2010. ICFN '10. Second International Conference.

[8] Borja Sotomayor, Ruben S.Montero, Ian Foster Virtual Infrastructure Management In Private And Hybrid Cloud published by the IEEE computer security, (2010) by, pp.14-22.

[9] GoGrid, http://www.gogrid.com.

[10] Aamir Nizam Ansari, Siddharth Patil,Arundhati Navada, Aditya Peshave, Venkatesh Borole , Online C/C++

Compiler using Cloud Computing, Multimedia Technology (ICMT), July 2011 International Conference, pp. 3591-3594.

[11] EMC Managed Storage Service, http://www.emc.com/.

[12] Ghemawat S, Gobioff H, Leung S-T (2003), The Google file system. In: Proc of SOSP, October 2003

[13] Amazon Inc. Amazon Web Services EC2 site. http://aws.amazon.com/ec2, 2008

[14] Chunye Gong Jie Liu Qiang Zhang Haitao Chen Zhenghu Gong, The Characteristics of Cloud Computing, Parallel Processing Workshops (ICPPW), 2010 39th International Conference.

[15] Eucalyptus home page,http://eucalyptus.cs.ucsb.edu/.

[16] I. Foster and C. Kesselman, editors. The Grid Blueprint for a New Computing Infrastructure. Morgan Kaufmann, 1998

[17] Amazon S3 Availability, http://status.aws.amazon.com /s3-20080720.html.

[18] GOOGLE Apps, http:// www.google.com/apps/ business/ index.html.