ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 10, October 2014

Implementation of Load Balancing Based on Partitioning in Cloud Computing

S. Adiseshu Gupta[1], K. V. Srinivasa Rao[2]

PG Student, Department of CSE, Prakasm Engineering College, Kandukur, AndhraPradesh, India

Associate Professor, Department of CSE, Prakasam Engineering College, Kandukur, Prakasam (Dt), AndhraPradesh, India

ABSTRACT: Load balancing in the cloud computing environment has an important impact on the performance. Good load balancing makes cloud computing more efficient and improves user satisfaction. This article introduces a better load balance model for the public cloud based on the cloud partitioning concept with a switch mechanism to choose different strategies for different situations. The algorithm applies the game theory to the load balancing strategy to improve the efficiency in the public cloud environment. Load Balancing Model Based on Cloud Partitioning for the Public Cloud environment has an important impact on the performance of network load. A cloud computing system which does not use load balancing has numerous drawbacks. Now-a-days the usage of internet and related resources has increased widely. Due to this there is tremendous increase in workload. So there is uneven distribution of this workload which results in server overloading and may crash. In such systems the resources are not optimally used. Due to this the performance degrades and efficiency reduces. Cloud computing efficient and improves user satisfaction. Thisarticle introduces a better load balance model for public cloud based on the cloud partitioning concept with a switch mechanism to choose different strategies for different situations. The algorithm applies the game theory for load balancing strategy to improve the efficiency in the public cloud environment.

KEY WORDS: load balancing model, public cloud, cloud partition, game theory.

I.INTRODUCTION

Cloud computing is an attracting technology in the field of computer science. In Gartner's report[1], it says that the cloud will bring changes to the IT industry. The cloud is changing our life by providing users with new types of services. Users get service from a cloud without paying attention to the details[2]. NIST gave a definition of cloud computing as a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction[3]. More and more people pay attention to cloud computing[4, 5]. Cloud computing is efficient and scalable but maintaining the stability of processing so many jobs in the cloud computing environment is a very complex problem with load balancing receiving much attention for researchers. Since the job arrival pattern is not predictable and the capacities of each node in the cloud differ, for load balancing problem, workload control is crucial to improve system performance and maintain stability. Load balancing schemes depending on whether the system dynamics are important can be either static and dynamic[6]. Static schemes do not use the system information and are less complex while dynamic schemes will bring additional costs for the system but can change as the system status changes. A dynamic scheme is used here for its flexibility. The model has a main controller and balancers to gather and analyze the information. Thus, the dynamic control has little influence on the other working nodes. The system status then provides a basis for choosing the right load balancing strategy.

The load balancing model given in this article is aimed at the public cloud which has numerous nodes with distributed computing resources in many different geographic locations. Thus, this model divides the public cloud into several cloud partitions. When the environment is very large and complex, these divisions simplify the load balancing. The cloud has a main controller that chooses the suitable partitions for arriving jobs while the balancer for each cloud partition chooses the best load balancing strategy.

10.15662/ijareeie.2014.0310055

Cloud computing is efficient and scalable but maintaining the stability of processing so many jobs in the cloud computing environment is a very complex problem with load balancing receiving much attention for researchers. Since the job arrival pattern is not predictable and the capacities of each node in the cloud differ, for load balancing problem, workload control is. crucial to improve system performance and maintain stability. Load balancing schemes depending on whether the system dynamics are important can be either static and dynamic . Static schemes do not use the system information and are less complex while dynamic schemes will bring additional costs for the system but can change as the system status changes. A dynamic scheme is used here for its flexibility. Disadvantages are Load balancing schemes depending on whether the system dynamics are important can be either static and dynamic . Static schemes do not use the system information and are less complex.

The basic designs of the system and algorithms to implement it are described in this paper Goals of Load Balancing

- To improve the performance substantially.
- To have a backup plan in case the system fails even partially.
- To maintain the system stability.
- To accommodate future modification in the system.

A. Literature Survey:

Cloud computing is an attracting technology in the field of computer science. In Gartner’s report[1], it says that the cloud will bring changes to the IT industry. The cloud is changing our life by providing users with new types of services. Users get service from a cloud without paying attention to the details[2]. NIST gave a definition of cloud computing as a model for enabling ubiquitous, convenient, on demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction[3].More and more people pay attention to cloud computing[4, 5]. Cloud computing is efficient and scalable but maintaining the stability of processing some any jobs in the cloud computing environment is a very complex problem with load balancing receiving much attention for researchers. Since the job arrival pattern is not predictable and the capacities of each node in the cloud differ, for load balancing problem, workload control is crucial to improve system performance and maintain stability.

Load balancing schemes depending on whether the system dynamics are important can be either static and dynamic[6]. Static schemes do not use the system information and are less complex while dynamic schemes will bring additional costs for the system but can change as the system status changes. A dynamic scheme is used here for its flexibility. The model has a main controller and balancers to gather and analyse the information. Thus, the dynamic control has little influence on the other working nodes. The system status then provides a basis for choosing the right load balancing strategy. The load balancing model given in this article is aimed at the public cloud which has numerous nodes with distributed computing resources in many different geographic locations. Thus, this model divide 1sthe public cloud into several cloud partitions. When the environment is very large and complex, these divisions simplify the load balancing. The cloud has a main controller that chooses the suitable partitions for arriving jobs while the balancer for each cloud partition chooses the best load balancing strategy.

II.PROPOSED ISSUES

(1) Cloud division rules: Cloud division is not a simple problem. Thus, the framework needs a detailed cloud division methodology. For example, nodes in a cluster may be far from other nodes or there will be some clusters in the same geographic area that are still far apart. The division rule should simply be based on the geographic location.

(2) How to set the refresh period for data statistics analysis, the main controller and the cloud partition balancers need to refresh the information at a fixed period. If the period is too short, the high frequency will influence the system performance. If the period is too much long, the information will be too old to make good decision. Thus, tests and statistical tools are needed to set reasonable refresh periods.

(3) A load status evaluation: A good algorithm is needed to set Load degree high and Load degree low, and the evaluation mechanism needs to be more comprehensive.

10.15662/ijareeie.2014.0310055

(4) Find out other load balance strategy: and other load balance strategies may provide better results, so tests are needed to compare different strategies. Many tests are needed to guarantee system availability and efficiency

Load balancing schemes depending on whether the system dynamics are important can be either static and dynamic . Static schemes do not use the system information and are less complex while dynamic schemes will bring additional costs for the system but can change as the system status changes. A dynamic scheme is used here for its flexibility. The model has a main controller and balancers to gather and analyze the information. Thus, the dynamic control has little influence on the other working nodes. The system status then provides a basis for choosing the right load balancing strategy.

The load balancing model given in this article is aimed at the public cloud which has numerous nodes with distributed computing resources in many different geographic locations. Thus, this model divides the public cloud into several cloud partitions. When the environment is very large and complex, these divisions simplify the load balancing. The cloud has a main controller that chooses the suitable partitions for arriving jobs while the balancer for each cloud partition chooses the best load balancing strategy.

There have been many studies of load balancing for the cloud environment. Load balancing in cloud computing was described in a white paper written by Adler[7] who introduced the tools and techniques commonly used for oad balancing in the cloud. However, load balancingin the cloud is still a new problem that needs new architectures to adapt to many changes. Chaczko etal.[8] described the role that load balancing plays in improving the performance and maintaining stability. There are many load balancing algorithms, such as Round Robin, Equally Spread Current Execution Algorithm, and Ant Colony algorithm. Nishant et al.[9] used the ant colony optimization method in nodes load balancing. Randles et al.[10] gave a compared analysis of some algorithms in cloud computing by checking the performance time and cost. They concluded that the ESCE algorithm and throttled algorithm are better than the Round Robin algorithm. Some of the classical load balancing methods are similar to the allocation method in the operating system, for example, the Round Robin algorithm and the First Come First Served (FCFS) rules. The Round Robin algorithm is used here because it is fairly simple.

III.PROPOSED WORK

A Load Balancing Model for the Public Cloud which is sed on cloud partitioning proposed efficient technique for balancing load in cloud. There is public cloud that has various nodes with distributed computing resources in number of different geographic locations. Thus, public cloud can be divided into several cloud partitions.

Whenever the environment is complex and huge, the load balancing can be simplified by these divisions. Then the he suitable partitions can be chosen by a main controller for arriving jobs. However, the balancer of each cloud partition chooses the best suitable load balancing strategy.

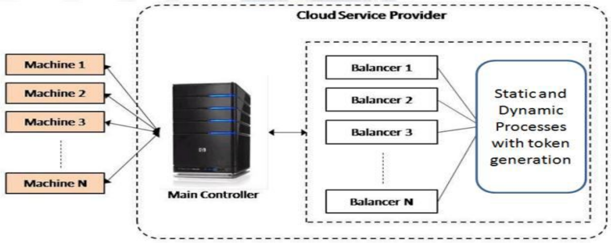


Fig. 1: System Architecture

A large public cloud has many nodes and those nodes can also be in different geographical locations. Cloud partitioning is one of the solutions to manage this large cloud. The public cloud has a subarea as a cloud partition with divisions based on the different geographic locations.

10.15662/ijareeie.2014.0310055

When the creation the cloud partitions is carried out, a load balancing starts at the job arrival, with the main controller deciding which cloud partition should receive the individual job. The partition of load balancer decides how the jobs should be assigned to the nodes. The partitioning will be able to accomplish locally, when the load status of a cloud partition is normal. The job should be transferred to another different partition, if the cloud partition load status is not normal.

A. Application Servers:

- Cloud architecture may contain one to many number of application servers according to its scope and utility.
- Each application server has its number of dedicated resources

B. Master Servers:

- Master server is the first component which interacts with client and accept its request
- It divide task into number of activities

A Load Balancing Model Based on Cloud Partitioning for the Public Cloud proposed efficient technique for balancing load in cloud.

Define a load parameter set: F = {F1; F2;…; Fm} with each Fi(1≤ i ≤ m; Fi € [0, 1]) parameter being either static or dynamic. m represents the total number of the parameters.

Then Compute the load degree as:

$$\text{Load\_degree(N)} = \sum_{i=1}^{m} \text{aiFi}$$

$$\text{Load degree}_{\text{avg}} = \frac{\sum_{i=1}^{m} Load\_Degree(Ni)}{n}$$

Step1:

Get size of all servers S1……………Sn

$$S = \sum_{i=1}^{s} aiSi$$

Step 2:

When, if S ∈ Server load(sL) is overflowed (limit exceeds) then File fu uploads on S ∈ server next to sL.

Where 1) Load is Idle When Load degree.(N) = 0;

2) Load is Normal when 0 < Load degree (N) ≤ Load degreehigh

3) Load is Overload when Load_degree high ≤ Load_degree(N)

Token Generation

1. User U1 sends file to server for file uploading

2. Check balancers load b1 and b2 While (DataReader.read()) If (B1= DataReader[S])

Begin

Get load From server S[]={s1,s2,s3,…,sn}

End

Else

While(DataReader.read()) If(B2= DataReader[S])

Begin

Get load From server S[]={s1,s2,s3,…,sn}

End

Else

3. Generation of token.

4. After generating token we will implement token generation algorithm.

5. And using this algorithm we will upload the file and balance the server.

6. If File Size Data **FS** of user **U1** > available space of Server

Steps given below from a to c.

Begin

a) Prevent overflow of server S1….Sn that exceed a limit of size.

b) Perform load evaluation technique.

c) Upload file to the next server.

End

7. If File Size Data **FS** of user **U1** < available space of Server

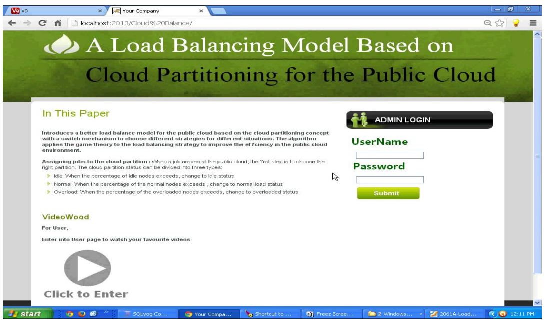
Steps given below from a to b.

8. Begin

a) Perform load evaluation technique.

b) Upload file to this server. End

IV. SIMULATION RESULT



localhost:2013/Cloud%20Balance/
A Load Balancing Model Based on
Cloud Partitioning for the Public Cloud
In This Paper
ADMIN LOGIN
UserName
Password
Submit
VideoWood
Click to Enter

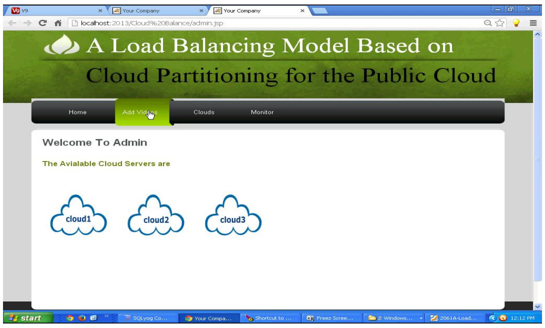
Introduces a better load balance model for the public cloud based on the cloud partitioning concept with a switch mechanism to choose different strategies for different situations. The algorithm applies the game theory to the load balancing strategy to improve the ef?ciency in the public cloud environment.

Assigning jobs to the cloud partition : When a job arrives at the public cloud, the ?rst step is to choose the right partition. The cloud partition status can be divided into three types:

- Idle: When the percentage of idle nodes exceeds, change to idle status
- Normal: When the percentage of the normal nodes exceeds , change to normal load status
- Overload: When the percentage of the overloaded nodes exceeds, change to overloaded status

For User,

Enter into User page to watch your favourite videos

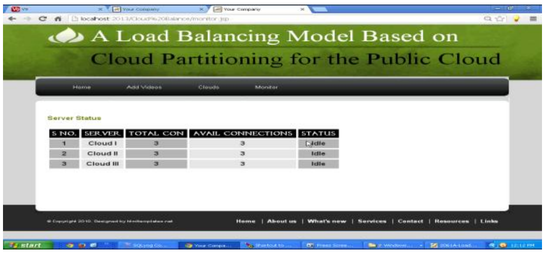


localhost:2013/Cloud%20Balance/admin.jsp
A Load Balancing Model Based on
Cloud Partitioning for the Public Cloud
Home
Add Videos
Clouds
Monitor
cloud1
cloud2
cloud3

Welcome To Admin

The Avialable Cloud Servers are

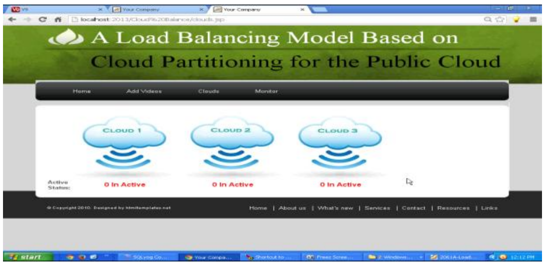
Admin:



localhost:2013/Cloud%20Balance/monitor.jsp
A Load Balancing Model Based on
Cloud Partitioning for the Public Cloud
Home
Add Videos
Clouds
Monitor
Home | About us | What's new | Services | Contact | Resources | Links

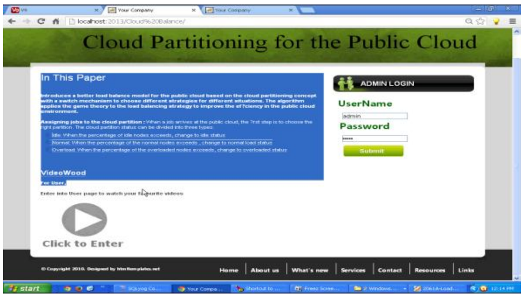
Server Status

S NO.	SERVER	TOTAL CON	AVAIL CONNECTIONS	STATUS
1	Cloud I	3	3	Idle
2	Cloud II	3	3	Idle
3	Cloud III	3	3	Idle

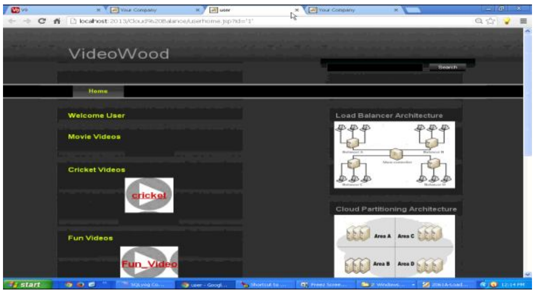


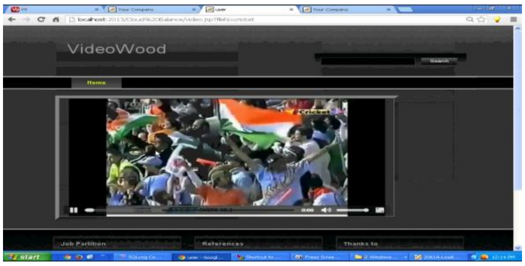
localhost:2013/Cloud%20Balance/clouds.jsp
A Load Balancing Model Based on
Cloud Partitioning for the Public Cloud
Home
Add Videos
Clouds
Monitor
CLOUD 1
CLOUD 2
CLOUD 3
Active Status:
0 In Active
0 In Active
0 In Active
Home | About us | What's new | Services | Contact | Resources | Links

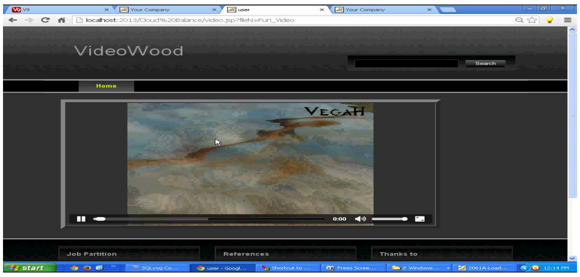
User:



localhost:2013/Cloud%20Balance/
Cloud Partitioning for the Public Cloud
In This Paper
ADMIN LOGIN
UserName
admin
Password
Submit
VideoWood
Click to Enter
Home | About us | What's new | Services | Contact | Resources | Links







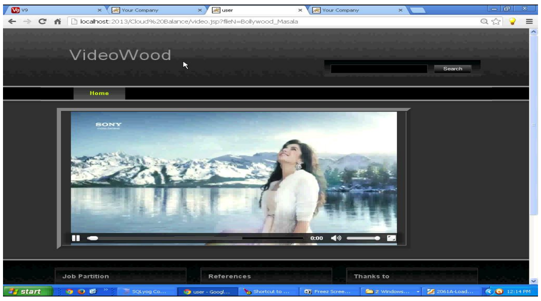
10.15662/ijareeie.2014.0310055

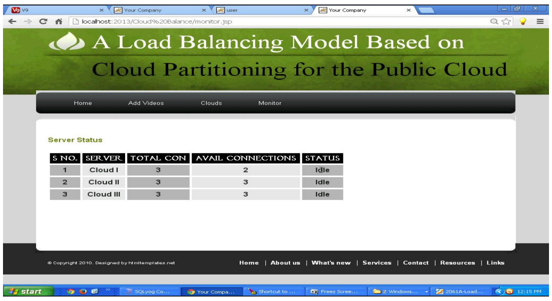
ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 10, October 2014




A Load Balancing Model Based on Cloud Partitioning for the Public Cloud

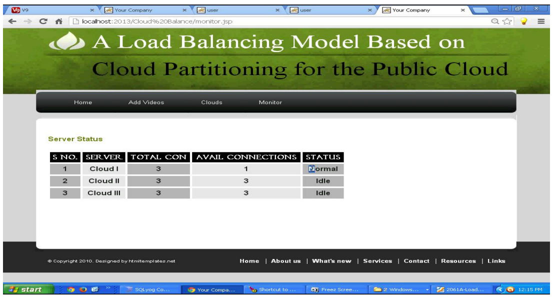
Home
Add Videos
Clouds
Monitor

Server Status

S NO.	SERVER	TOTAL CON	AVAIL CONNECTIONS	STATUS
1	Cloud I	3	2	Idle
2	Cloud II	3	3	Idle
3	Cloud III	3	3	Idle

© Copyright 2010. Designed by htmltemplates.net

Home | About us | What's new | Services | Contact | Resources | Links


A Load Balancing Model Based on Cloud Partitioning for the Public Cloud

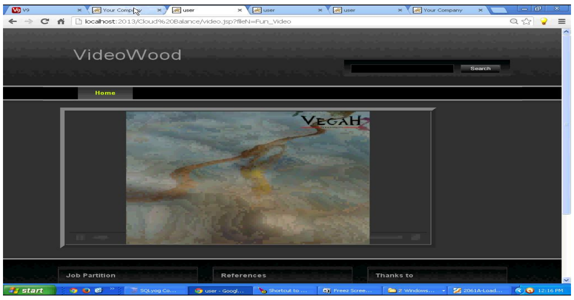
Home
Add Videos
Clouds
Monitor

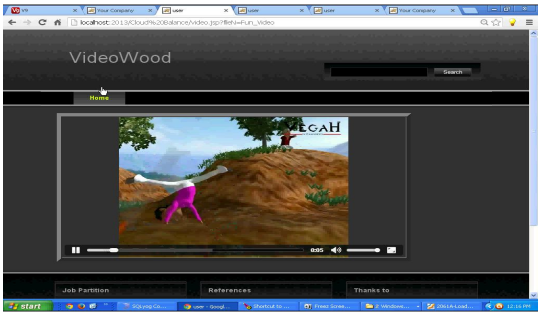
Server Status

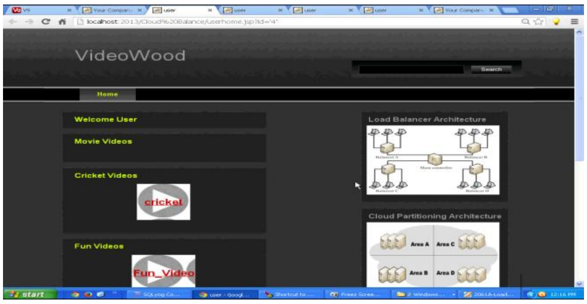
S NO.	SERVER	TOTAL CON	AVAIL CONNECTIONS	STATUS
1	Cloud I	3	1	Normal
2	Cloud II	3	3	Idle
3	Cloud III	3	3	Idle

© Copyright 2010. Designed by htmltemplates.net

Home | About us | What's new | Services | Contact | Resources | Links







V.CONCLUSIONS AND FUTURE SCOPE

The system will get the load of all cloud systems to perform following functions as the client while Controller will get the load of all balancers and send it to the user by establishing connection to balancers. After this, by making connection to servers, balancer gets the load of all servers and sends it to the Controller.

Client will upload the files on the cloud by using load balancing algorithm to controller who uploads the file to the balancer who has a minimum load. Next, balancer uploads the file to the server that has a minimum load. Client can Search and Download the file and Deployments from the controller who will search and download the file from balancers. Next, cloud will Search and Download the particular file from sever.

The application will check the available space and actual size of the server. It may vary from each other and also the load of the server it contains so. After uploading the file it will check the size of file and the file will be uploaded to the other server if its size is greater than the available size. In this way, one can prevent beyond the limit and also runs out of available bandwidth..

REFFRENCES

1.Gaochao Xu, Junjie Pang, and Xiaodong Fu (2013), A Load Balancing Model Based on Cloud Partitioning for the Public Cloud Proc. 14th European Conf. Research in Computer Security (ESORICS '09 IEEE transactions on cloud computing year

10.15662/ijareeie.2014.0310055

2.K. Nishant, P. Sharma, V. Krishna, C. Gupta, K. P. Singh, N. Nitin, and R. Rastogi (Mar. 2012), Load balancing of nodes in cloud using ant colony optimization, in Proc. 14th International Conference on Computer Modelling and Simulation (UKSim), Cambridgeshire, United Kingdom,, pp. 28-30.

3.Z. Chaczko, V. Mahadevan, S. Aslanzadeh, and C. Mcdermid (2011), Availability and load balancing in cloud computing,; presented at the 2011 International Conference on Computer and Software Modeling, Singapore.

Stavros Papadopoulos, Spiridon Bakiras, and Dimitris Papadias, pCloud: A Distributed System for Practical PIR , supported by grant HKUST 618108 from Hong Kong RGC, and by the NSF Career Award IIS- 0845262.

4 .B. Adler, Load balancing in the cloud: Tools, tips and techniques, http://www.rightscale. com/info center/white- papers/Load-Balancing-in-the-Cloud.pdf, 2012.

5. R. Hunter, The why of cloud, http://www.gartner.com/DisplayDocument?doc cd=226469&ref= g noreg, 2012.

6. M. D. Dikaiakos, D. Katsaros, P. Mehra, G. Pallis,and A. Vakali, Cloud computing: Distributed internetcomputing for IT and scientific research, InternetComputing, vol.13, no.5, pp.10-13, Sept.-Oct. 2009.

7. P. Mell and T. Grance, The NIST definition of cloudcomputing,http://csrc.nist.gov/ ublications/nistpubs/800-145/SP800-145.pdf, 2012.8. Microsoft Academic Research, Cloud computing, http://libra.msra.cn/Keyword/6051/cloud-computing?query=cloud%20computing, 2012.

9. Google Trends, Cloud computing, http://www.google.com/trends/explore#q=cloud%20computing, 2012.

10. N. G. Shivaratri, P. Krueger, and M. Singhal, Loaddistributing for locally distributed systems, Computer,vol. 25, no. 12, pp. 33-44, Dec. 1992.

BIOGRAPHY

MR.S. Adiseshu Gupta received B. Tech. degree in Computer Science and Information technology from JNTUK University, in 2011. Currently he is doing M. Tech. in Prakasam Engineering College, from JNTUKUniversity, Kakinada, India.

K. V. Srinivasa Rao received M.TECH degree in ComputerScience and Engineering from JNTUA University, and currently heis working as an Associate Professor, Department of CSE inPrakasam Engineering College, Kandukur, India