# An Approach for Server Consolidation in a Priority Based Cloud Architecture

**Suneel K S[1] and H S Guruprasad[2]**

*Submitted in March, 2015; Accepted in December, 2015*

***Abstract - Cloud computing is a new emerging technology in IT field. Cloud computing has capability to provide whole IT as a service to its users. There are many characteristics of cloud computing that makes it attractive in a variety of problems. Users of the cloud are free from the housekeeping activities related to the infrastructure. This is because in cloud computing, managing the hardware is cloud service provider's concern. When a user application is deployed in the cloud, depending on the QoS specified by the user, cloud service provider deploys servers for smooth running of the user application. The total number of servers deployed for a user application must be optimal, because underutilized servers are not economical for both cloud service provider and cloud user. Underutilized servers consume power when they are idle; hence deploying optimal number of servers is critical in the operation of the cloud. Server consolidation is the method of increasing the utilization levels of servers, such that more applications are accommodated in servers, this avoids unwanted deployment of servers. This paper provides an approach towards server consolidation in a priority based cloud architecture.***

***Index Terms - Cloud Computing, Server Consolidation, Priority based Cloud Architecture, Server Consolidation Algorithm, Live Task Migration***

## 1.0 INTRODUCTION

Cloud computing is mainly based on utility computing. Utility computing meters the amount of resources, which is provided for the customer, and bills the customer accordingly. Usually utility computing has limited set of resources to provide but cloud computing has huge computing infrastructure that can be provisioned to the customer. Similar to water, electricity and many other utilities, 21st generation is moving towards utility computing [1]. Utility computing is realized through cloud computing. A cloud has one or more datacenters and each datacenter may have one or more hosts on which the tasks submitted by the customers run.

There are many features that make cloud computing very successful in current computing generation. Among these, two features of concern are "pay-as-you-go" and shared environment. "pay-as-you-go" means that the customers of the cloud may not have to be bothered about the upfront financial investment before using the cloud and charges from the cloud, are based on the amount and time of resources that the customer has used.

If the demand for which the organization or company requires computing resources is periodic or volatile or the company is not able to accommodate the computing infrastructure, then cloud computing acts as the perfect solution. In cloud computing, the computing infrastructure and its related aspects are cloud service provider's concern. Second feature that is of concern is the shared environment. Multiple users share the same underlying cloud infrastructure. This increases the utilization of the resources and thus decreases the cost per utilization of the resources.

Cloud computing has improved the computing infrastructure efficiency by means of four key factors [2].

1. Dynamic provisioning: Typically, IT managers deploy more resources than the actual resources that are needed to run an application. The reason for this is to tackle the fluctuations in demand. But, this results in over-provisioning and underutilization of resources. In cloud computing, cloud service providers deploy dedicated resources to overview and predict the behavior of demand of the deployed application, so that they can automatically scale-up and down the resources depending on the demand of the application. This is called dynamic provisioning and it is visioned on the better utilization of IT resources.

2. Multi-tenancy: Multiple organizations can use the same cloud for their purposes. The cloud differentiates between each organization and provides resources accordingly. This leads to the shared environment which increases the utilization of resources and decreases the overall energy use.

3. Server Utilization: Servers in cloud can process requests at a greater speed than the on-premise servers. The reason for this is that cloud service providers tend to run the servers at a higher utilization levels. Servers at the cloud can handle multiple types of requests using virtualization. For each request, a virtual machine is created in cloud and all virtual machines are made to run on a single underlying platform.

4. Datacenter Utilization: Server utilization ultimately leads to the datacenter utilization. Cloud service providers carefully balance the energy consumption in datacenter and datacenter utilization.

The software at the cloud service provider side, such as cloud coordination software, is responsible for the management of idle time of hosts under a datacenter. The load balancing at the datacenter specifies that no host should be idle until all the user requests at the datacenter are executed. A static load balancing strategy is not sufficient to manage the idle time of the hosts. Hence, in this paper a runtime strategy, which uses

[1] PG Scholar, Dept. of CSE, BMSCE, Bangalore, kssunil.shastry@gmail.com,
[2] Professor and Head, Dept. of CSE, BMSCE, Bangalore, drhsguru@gmail.com [Corresponding Author]

collaboration between servers, is used in combination with the static strategy in the management of idle time of servers at the datacenter. Management of idle time of servers in datacenter finally leads to the server consolidation.

## 2.0 LITERATURE SURVEY

### 2.1 Priority based cloud architecture:

A cloud in cloud computing is a set of one or more datacenters and each datacenter has one or more hosts that execute the user’s tasks. In a priority based cloud, hosts are grouped into different sets and each set is assigned a priority level.

Hosts in a priority level only execute incoming tasks which corresponds to same priority level. Depending on the levels of the priority of user’s tasks, the grouping of servers in the datacenter can be tuned accordingly.

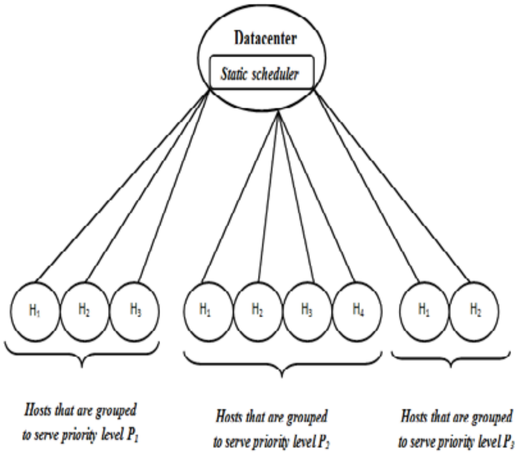
Datacenter
Static scheduler
$H_1$
$H_2$
$H_3$
$H_1$
$H_2$
$H_3$
$H_4$
$H_1$
$H_2$
Hosts that are grouped to serve priority level $P_1$
Hosts that are grouped to serve priority level $P_2$
Hosts that are grouped to serve priority level $P_3$

**Figure 1.1: Priority based cloud architecture**

In Fig 1.1, the datacenter can handle three priority levels. The Figure shows the static scheduler that is situated at the datacenter level. The main functionality of this static scheduler is to distribute the tasks onto the group of hosts according to their priority levels.

This architecture uses static approach for load balancing. The basic essence of load balancing is that the load on the hosts must be distributed such that no host should be idle until all the tasks in the datacenter are executed. This means that the hosts in the datacenter must be utilized almost equally to increase the speed of task execution. In this architecture, if the granularity of arrival of tasks at a priority level, say p, is less than the granularity of arrival of tasks at priority level p', then hosts that handle priority p is always idle between the two corresponding arrival of set of tasks and hosts that handle priority level p'is always overloaded.

This introduces the variation in the utilization levels of hosts. This variation affects the overall utilization of the datacenter. In order to overcome this, a combination of static and dynamic approach for load balancing is used in this paper which is adaptive to the events of load balancing. The approach proposed in this paper uses the collaboration between the hosts of a datacenter, so that they can manage their workloads. Since both static and dynamic approaches have been used, the proposed methodology is stable and effective [3].

### 2.2 Server Consolidation

Datacenters are the central elements in cloud computing. Datacenters are responsible for the storing, managing, networking and controlling of data. A Datacenter is made up of many numbers of servers. 30% of the servers in a datacenter are under-utilized [4]. Due to under-utilized servers the ratio of power consumption to server utilization is very high. This indicates that the amount of power consumed by a under-utilized server is same as the amount of power consumed by a moderately utilized server [4]. In cloud computing, cloud service provider must be able to rapidly increase the server numbers in peak demands. Increasing number of servers at peak time adds flexibility in the technology. But, technology like cloud computing which treats electric power also a utility, must optimize the number of servers required for the operation. This calls for the server consolidation techniques that have to be implemented at the cloud [5]. Server consolidation in layman’s term can be defined as the process of aggregating multiple tasks running on different servers to a reduced optimum number of servers.

[7 & 10] lists the challenges of the datacenter. Virtualization in datacenter along with server consolidation is sufficient enough to address all the challenges specified in [6]. Some of the advantages of server consolidation is mentioned in [7]. Server consolidation in datacenters leads towards the energy efficiency and resource utilization. But, there is calculated amount of risk in server consolidation. Discussable risk associated with the server consolidation is the performance. When more and more tasks are multiplexed onto a server, achieving performance isolation becomes difficult [8].

### 2.3 Utilization of hosts in datacenter and overall performance of datacenter:

The above graph shows the percentage of utilization of three hosts in a particular datacenter. It is evident from the figure that the host 3 is utilized at most and host 1 is least utilized. This imbalance in the utilization levels costs the performance of the datacenter because the host 3 can become a performance bottleneck since it is overloaded all the time. The imbalance in the utilization level of hosts indicates that the load is not exactly balanced onto the hosts or the ineffectiveness of the load balancer. In these conditions usual tendency of the cloud coordination software is to deploy more hosts to offset the performance overhead that has occurred. This phenomenon is called automatic scale-up. Although, scaling up does increase the performance, it also increases the energy consumption of the datacenter. Increase in energy consumption raises many concerns in the area of power consumption and environmental impacts.

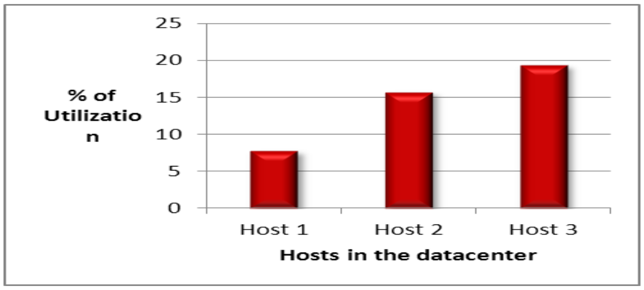

**Figure 1.2: Utilization of Servers in a datacenter**

Solution to the above problem is to look at the host utilization. [9] Provides proof that server utilization is the area of improvement to decrease the power consumption in the datacenters. With respect to the above graph, the solution is to balance the utilization levels of hosts under the datacenter either by using improved load balancing techniques or by using special techniques that target the area where load balancer cannot reach. This paper provides one such mechanism that uses the host collaboration to synchronize the workloads between the hosts so that the utilization levels are almost close. The proposed approach also aims at server consolidation.

**2.4 Task migration in cloud**

A task in cloud represents the user's work that is to be executed in the cloud. In order to understand the task migration in cloud following concepts are required.

**2.4.1 Virtualization in cloud:**

Cloud mainly relies on server virtualization. Virtualization is used to address the volatile requirements of the computing environments of cloud user. Each server in a cloud is virtualized. Virtualization is the process of abstracting physical hardware and to provide logical or virtual hardware for use of the applications and operating systems. These virtual hardware are called virtual machines (VM). Each virtual machine contains an operating system called as guest-OS that runs on the virtual machine and applications that run on these guest-OS [10].

Some of the benefits of virtualization are [11]

1. Cost reduction: Several virtual machines run on a single underlying hardware reduces the cost of using different hardware for different applications.

2. Decoupling: Traditionally, applications that needed to run on a machine should be present on the same machine. But virtualization decouples this binding and typically allows the applications to reside on the virtual machine.

3. Flexibility: Since a virtual machine represents the environment that an application requires to run, flexibility in using the infrastructure by creating a virtual machine for each of the application.

4. Sustainability: Virtualized environments are soft on resources. They use fewer resources and this leads to the efficient utilization of resources.

A software that runs on the top of physical hardware of the server and manages the virtual machines in the server is called as the virtual machine monitor (VMM). VMM can be visualized as the operating system of servers. VMM are also called as hypervisors [10].

Since every task in cloud runs on its VM, it is easy to migrate the task from one host to another. It is as simple as moving VM from source host to the destination host.

**2.5 Lifetime of a task in cloud:**

Lifetime of a task in a cloud includes the following stages [12]

1. Upload: Once a new task is generated by the user, its source code and the data required for the creation of VM is sent to the datacenter of the cloud.

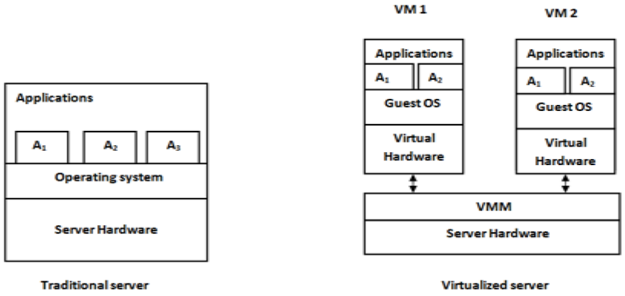

**Figure 1.3: Traditional and virtualized server architectures**

2. Task assignment: Once the data required for the execution of the task is uploaded to the datacenter, a local dispatcher assigns the task to a server for the execution.

3. Execution: The actual execution of the task is done at this stage by the server.

4. Migration: A task along with its VM may be transferred from one server to another so that the execution is handled by the destination server. This can happen several times in a task's lifetime.

5. Download: At this stage, the user retrieves the results of the execution from the server of the datacenter.

**2.6 Challenges for migrating tasks in cloud**

The major challenges in migration of a task from one host to another in cloud are defined by two important questions:

**2.6.1 When to migrate a task from source host to the destination host in a cloud?**

Whenever a task is submitted to the cloud, how to decide when the task has to be migrated from source host to the destination host. There are three models that decide the timing of task migration.

*Centralized migration policies:* In centralized migration policies, there will be a central component that decides when to migrate a task from source to destination host. This component considers various parameters for the decision such as load balancing, resource acquisition and so on [13]

*Server-initiated migration:* In centralized migration policies, the complexity of decision is high since it is a cloud-wide approach. Server-initiated migration reduces this complexity. In this approach, server is responsible to take the decision of migration of tasks.

Eg: whenever the load on the server increases, it initiates a migration of tasks to a destination server that is idle.

*Task-autonomic migration:* This gives task the capability to decide about the migration.

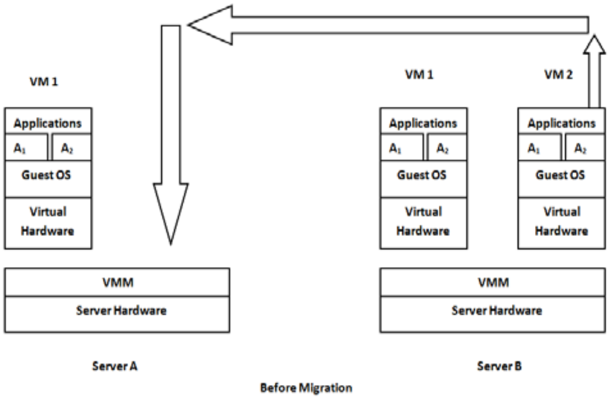

**Figure 1.4: Migration of task between servers in a datacenter**

### 2.6.2 How the migration of the tasks from source to destination host happens?

The task migration can be classified into two categories.

*Live migration:* In this type of migration, the task will be still working during its migration from one host to another.

*Non-live migration:* In this type of migration, the VM is stopped at the source host and it is transferred to the destination host and then the VM is started. This type of migration leads to the black-out during the migration of the task.

There are many algorithms that can be used to migrate tasks from source host to destination host. Some of them are:

*Pre-copy:* In this algorithm, the processor on which the task is running is not stopped during its migration to the destination host.

*Post-copy:* In this algorithm, the modes of the VM and least information that is required for the starting up of the VM at the destination host is sent to the destination host. Then, VM is started at the destination host, after this the source host initiates the page transmission to the destination host. This algorithm eliminates the overhead of resending pages that exist in pre-copy algorithm. [14, 15].

*Three-phase migration (TPM*): This algorithm is similar to the pre-copy and has least suspension time for migration of VM with all its mode along with virtual disk [16].

## 3.0 PROPOSED METHODOLOGY

The following functions are required for defining the algorithm.

*Granularity of arrival of tasks at priority levels (G(P)):* This function, denoted by G(P) where p is a priority level, determines the granularity of arrival of tasks at priority level p Granularity of arrival of tasks at a priority level 'p' is the amount of time elapsed after the completion of current set of tasks and before the arrival of new set of tasks both at a priority level 'p'.

*Priority of tasks and servers:* The priority of the tasks and servers are calculated by the following functions.

PLT(t) - determines the priority level of task 't' which are dynamic.

PLS(s) - determines the priority level of server 's' which are preset.

*Execution time of set of tasks (E(T)):* This function determines the execution time of set of tasks T.

E(T) - execution time of set of tasks T where T={t1,t2,…..,tn}

Algorithm:

Let S= < S1,,S2,…….,Sn> be the ordered set of servers that can handle tasks with priority levels 1,2,………,n and PLS(S1 ) > PLS(S2).

1. Let T be the set of tasks arrived at the datacenter from the user. For each task ti ∈ T, calculate PLT(ti) and assign them to the respective servers.
2. Servers assigned with the tasks start executing the tasks. Whenever a server Si ∈ S finishes execution of all the tasks assigned to it, Si sends finish message to an immediate server Sj such that PLS(Si ) > PLS(Sj).
3. Whenever a server Sj gets a finish notification from Si, it performs following steps.
   3a. Calculates the granularity of tasks at priority level i by using G(PLS(Si)).
   3b. Selects a set of tasks 'M' such that M={m1,m2,……,mk} where mi is a task and 1<=i<=k. And calculates E(M).
   3c. If G(PLS(Si))<E(M), then migrate set of tasks M from Sj to Si for it execution at Si and exit.
   Else
   Reduce the number of tasks in M an recalculate the E(M) and goto step 3.c

## 4.0 RESULT AND DISCUSSIONS

The diagrammatic explanation of the working of the algorithm is shown below.

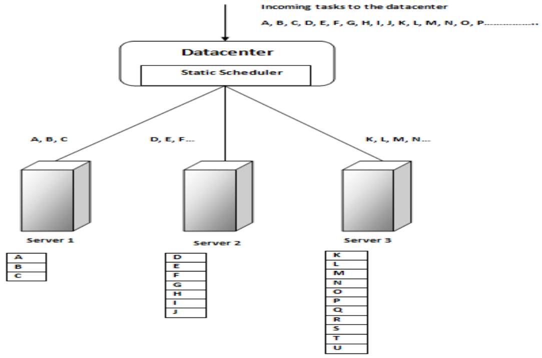

**Figure 1.5: Phase-1 in the working of proposed algorithm**

Let's consider a datacenter with 3 servers, which serve different priority levels. Let A, B, C… be the set of tasks arriving at the datacenter.

The simulation of the proposed methodology is done using Cloudsim Simulator**.** The graph shown in Fig 1.7 clearly explains the utilization of servers in the datacenter. Server utilization curve of the servers without consolidation is almost equivalent to the linear curve. This asserts that the server 3 is over utilized and server 1 is underutilized and the server 2 is in moderate utilization. This imbalance is due to the static scheduler which schedules the incoming tasks to the cloud in a pre-defined way and is not able to balance the utilization levels of the servers. The curve that represents the server utilization levels in a datacenter with consolidation is higher than its counterpart. Efficient balance of the load onto servers normalizes the utilization level of servers which is justified by the corresponding curve.

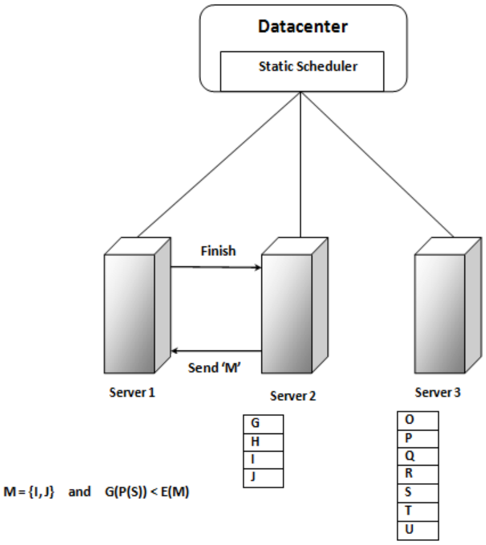

**Figure 1.6: Phase-2 in the working of proposed algorithm**

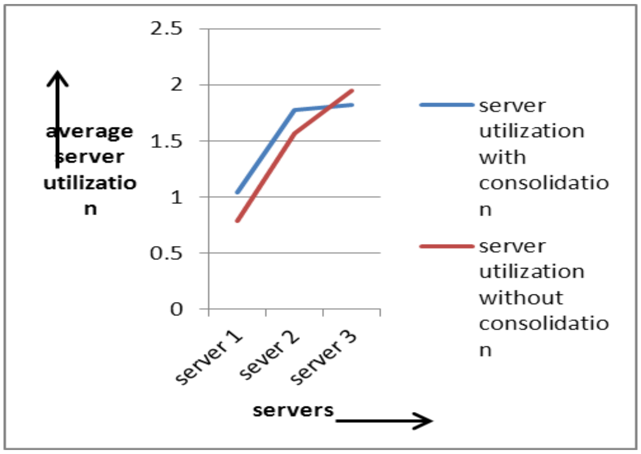

**Figure 1.7: Utilization of servers in the datacenter**

Fig 1.8 and Fig 1.9, shows the waiting time of tasks at server 2 and server 3. Primary analysis of these graphs specifies the reduction in the waiting time of the tasks after a period of time. Reduction in the waiting time of the tasks at server 2 starts after the tasks at server 1 has finished and server 1 has requested server 2 to send some of its tasks for execution at server 1. The reduced waiting time can be calculated from the graph. In this setup reduction in the waiting time of tasks is due to the effective migration of the tasks that are waiting in one server to another. The overall approach for server consolidation is justified by the graph that is obtained from the simulation of an instance of the problem.

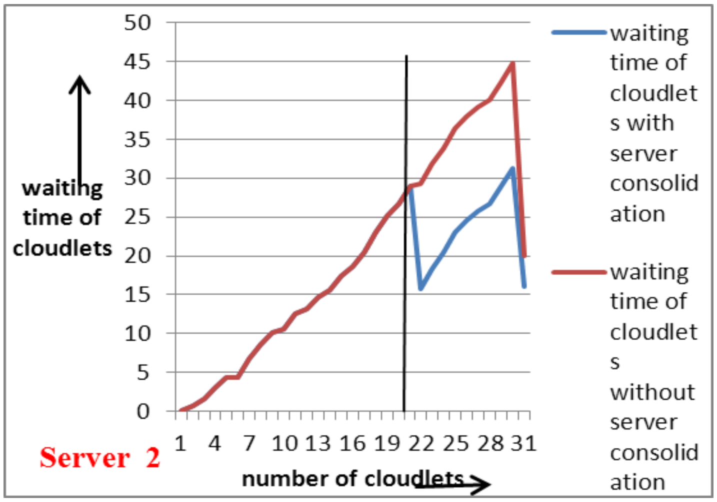

**Figure 1.8: Waiting time of tasks at server 2**

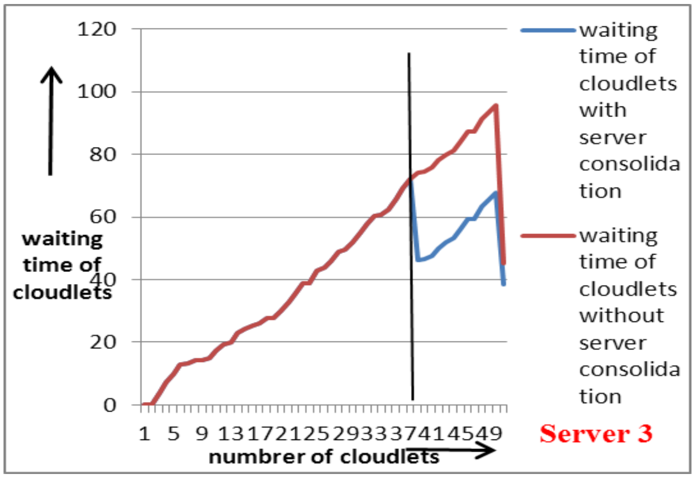

**Figure 1.9: Waiting time of tasks at server 3**

## 5.0 CONCLUSION

The simulated result of the proposed algorithm for server consolidation works positively and the same is justified by the graphs that are plotted using the simulation results. From the graphs of the simulation, it can also be concluded that the proposed algorithm is efficient increasing the utilization time of the servers by effectively adjusting the waiting time of the tasks at different servers. The proposed algorithm can be further extended to many cloud computing architecture.

## 6.0 ACKNOWLEDGEMENT

The work reported in this paper is supported by the College [BMSCE] through the TECHNICAL EDUCATION QUALITY IMPROVEMENT PROGRAMME [TEQIP-II] of the MHRD, Government of India.

## REFERENCES

[1] Rajkumar Buyya, Chee Shin Yeo, Srikumar Venugopal, “Market-Oriented Cloud Computing: Vision, Hype, and Reality of Delivering Computing as the 5th Utility”, CCGRID '09 Proceedings of the 2009 9th IEEE/ACM International Symposium on Cluster Computing and the Grid, DOI: 10.1109/CCGRID.2009.97.

[2] “Cloud Computing and Sustainability: The Environmental Benefits of Moving to the Cloud”, a white paper by Accenture corp LTD in collaboration with WSP.

[3] Niranjan G. Shivaratri, Phillip Krueger, Mukesh Singhal, “Load Distributing for Locally Distributed Systems”, Computer journal, Volume 25 Issue 12, December 1992, DOI: 10.1109/2.179115.

[4] Mueen Uddin, Azizah Abdul Rahman, “Server Consolidation: An Approach to Make Data Centers Energy Efficient & Green”, International Journal of Scientific & Engineering Research, Volume 1, Issue 1, October-2010.

[5] Moreno Marzolla, Ozalp Babaoglu, Fabio Panzieri, “Server Consolidation in Clouds through Gossiping”, WOWMOM '11 Proceedings of the 2011 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, DOI: 10.1109/WoWMoM.2011.5986483.

[6] “Server Consolidation Using Cisco Unified Computing System and EMC CLARiiON Storage”, White Paper June 2010, Revision 1.0.

[7] Vernon Turner, Matthew Eastwood, “Server Consolidation Benefits, Considerations, and Dell's Approach” An IDC White Paper.

[8] Hui Lv, Yaozu Dong, Jiangang Duan, Kevin Tian, “Virtualization Challenges: A View from Server Consolidation Perspective”, VEE '12 Proceedings of the 8th ACM SIGPLAN/SIGOPS conference on Virtual Execution Environments, Pages 15-26, DOI: 10.1145/2151024.2151030.

[9] Marjan Gusev, Sasko Ristov, Monika Simjanoska, Goran Velkoski, “CPU Utilization while Scaling Resources in the Cloud”, Cloud Computing 2013, Proceedings of 4th Int. Conference on Cloud Computing, GRIDS, and Virtualization, Valencia, Spain, IEEE Conference proceedings, ISBN 978-1-61208-216-5, IARIA, 2013.

[10] “Virtualization Overview”, A white paper by VMware Inc.

[11] “Virtualization: Benefits and Challenges”, A white paper by Information Systems Audit and Control Association (ISACA).

[12] Lazaros Gkatzikis, Iordanis Koutsopoulos, “Migrate or Not? Exploiting Dynamic TaskMigration in Mobile Cloud Computing Systems”, Wireless Communications, IEEE Volume: 20, Issue: 3, DOI: 10.1109/MWC.2013.6549280.

[13] Christopher Clark, Keir Fraser, Steven Hand, Jacob Gorm Hansenf, Eric Julf, Christian Limpach, Ian Pratt, Andrew Warfield, “Live Migration of Virtual Machines”, NSDI'05 Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation - Volume 2, Pages 273-286.

[14] Julian Fejzaj, Igli Tafa, Elinda Kajo, “The improvement of Live Migration in Datacenter in different virtual environment” IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 6, No 2, November 2012.

[15] Michael Richmond, Michael Hitchens, “A new process migration algorithm”, ACM SIGOPS Operating Systems Review Homepage table of contents archive,Volume 31 Issue 1, Pages 31-42, DOI: 10.1145/254784.254790.

[16] Yingwei Luo, Binbin Zhang , Xiaolin Wang, Zhenlin Wang , Yifeng Sun, Haogang Chen, “Live and Incremental Whole-System Migration of Virtual Machines Using Block-Bitmap”, in Proceedings of Cluster 2008: IEEE International Conference on Cluster Computing. IEEE Computer Society, 2008, pp. 99-106.

[17] Rodrigo N Calheiros, Rajiv Ranjan, Anton Beloglazov, Cesar A F De Rose, Rajkumar Buyya, ”CloudSim: A Toolkit for the Modeling and Simulation of Cloud Resource Management and Application Provisioning Techniques”, Journal of software Practice and Experience, Vol 41, Issue 1, Jan 2011, pp 23-50, DOI: 10.1002/spe.995.

[18] Rajkumar Buyya, Rajiv Ranjan, Rodrigo N Calheiros,”Modeling and Simulation of Scalable Cloud Computing Environments and the CloudSim Toolkit: Challenges and Opportunities”, 7th High Performance Computing and Simulation Conference, Leipzig, Germany, June 21-24, 2009, ISBN: 978-1-4244-4907-1.

[19] Bhavani B H, H S Guruprasad, “Resource Provisioning Techniques in Cloud Computing Environment: A Survey”, International Journal of Research in Computer and Communication Technology, Volume: 3, Issue: 3, March 2014, pp 395-401, ISSN (Online) 2278-5841, ISSN (Print) 2320-5156.

[20] Bhavani B H, H S Guruprasad, “Efficient Resource Scheduling under Varying workloads in Cloud Data Center”, International Journal of Emerging Engineering Science and Technology, Volume 1, Issue 1, pp 40-45, Feb 2015.