Identifying the Quality Parameters for Virtualization in Cloud Computing

Adesh Kumar
Department of Computer Science
SLBSRSV, New Delhi India

Cloud computing technology provides scalable and flexible technical infrastructure capabilities as an on- demand service (Singh and Kumar, 2014). The users can access cloud computing resources from any place, anywhere and at any time via any mobile computing devices such as laptops, mobiles, tablets or smart phones. In cloud computing technology, machines with large data centers can be dynamically provisioned, configured, controlled and reconfigured to deliver services in a scalable manner. Cloud computing allows to efficiently manage upgrades and maintenance, backups, disaster recovery and failover functions. According to Vaquero et al., cloud guarantees are offered by the infrastructure provider by means of customized Service Level agreements (SLA). Cloud computing solves the potential problems of education, climate change, economics and terrorism. In this paper a study is carried out to identify the quality parameters for virtualization in cloud computing.

Keywords- Quality Parameter, Cloud Computing, virtualization.

I. INTRODUCTION

Cloud computing reduces the investment on purchasing the hardware, software and software licenses by providing services on rental basis (Pocatilu et al, 2010). It reduces the licensing cost and provides backups to keep multiple copies of data. Cloud computing technology provides scalable and flexible technical infrastructure capabilities as an on- demand service (Singh and Kumar, 2014). The users can access cloud computing resources from any place, anywhere and at anytime via any mobile computing devices such as laptops, mobiles, tablets or smart phones. In cloud computing technology, machines with large data centers can be dynamically provisioned, configured, controlled and reconfigured to deliver services in a scalable manner (Zissis and Lekkas, 2012). Cloud computing allows to efficiently manage upgrades and maintenance, backups, disaster recovery and failover functions (Zaharescu, and . Zaharescu, 2012) . According to Vaquero et al. (Vaquero et al., 2008), cloud guarantees are offered by the infrastructure provider by means of customised Service Level agreements (SLA). Cloud computing solves the potential problems of education, climate change, economics and terrorism (Kop and Carroll, 2011).

Cloud Computing Deployment Type-

Cloud computing has various deployment models that are accepted by cloud users today (Mell and Grance, 2009). Cloud computing can be classified as public, private, community or hybrid models as shown in figure 1.

Public cloud- Public cloud are available under pay-as you-go way consumption model to the general public. The cloud resources are hosted in the premises of the cloud service provider.

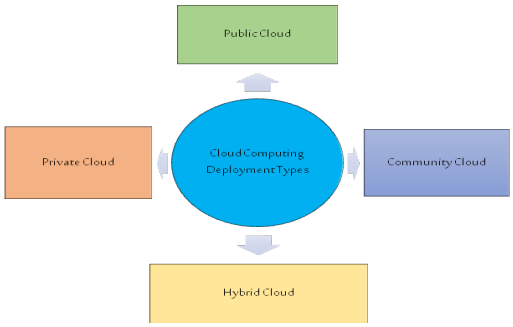

Figure 1. Cloud Computing Deployment Type

Private cloud- Private cloud are configured, operated and managed for internal use of any organizations that is not available for general public (Mell and Grance, 2009). Private cloud allows users to interact with local data center while experiencing the cloud interface and advantages of public cloud.

Community Cloud- Community clouds are shared by several organizations and support a specific community that has shared resources or concerns . These clouds are managed by third party or may exist on premise on or off premise.

Hybrid cloud- Hybrid clouds are made by combining public cloud and private cloud together with standardized technology that enables data and application portability (Mell and Grance, 2009). Hybrid model is adopted by most of the organization.

Cloud Computing Deployment Type-

There are various layers of services in cloud computing so first I discuss about the services used in cloud computing then I will find the quality parameter for each service.

Software as a service (Saas) - The Software as a service (Saas) reside on the top of the service models that can be accessed by end users. The users are increasingly shifting from locally installed computer applications to online software services. The use of Saas service model eliminates the burden of software maintenance on local system (Hayes, 2008).

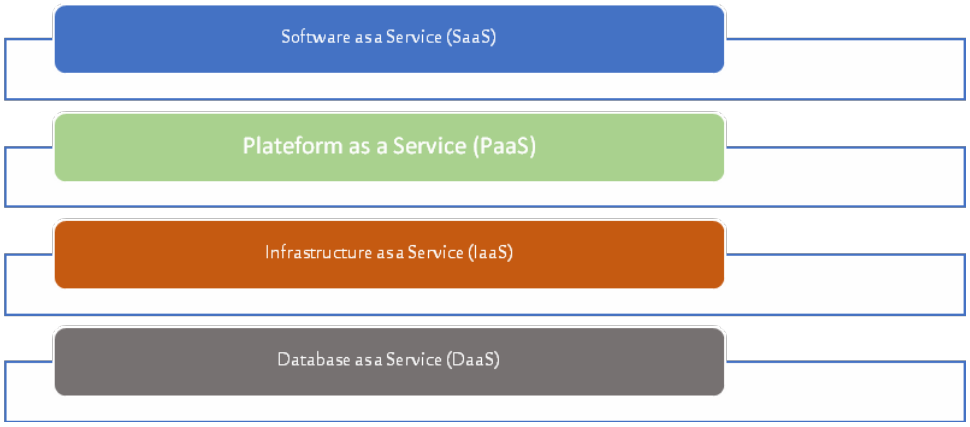

Figure 2. Cloud Computing Service Type

Platform as a service (Paas) -The second abstraction layer, platform as a service (Paas) is the programmable layer of service model used by programmers to develop and deploy codes in the clouds. Developers do not need to know the hardware configuration running in the cloud. The developers can access and program in various types of development environment. Paas platform provides various types of specialized services for data access and authentication (Buyya et al. 2010).

Infrastructure as a service (Iaas) -The Infrastructure as a service (Iaas) is a third layer of service model manages physical resources in the cloud. It manages the virtual machines and other virtualized resources running in the cloud computing environment on demand (Sotomayor,et al., 2009) . The infrastructure as a service runs different types of operating system with customized stack of softwares (Nurmi at al., 2009). This layer provides on- demand provisioning of server running to satisfy varied user requirements.

Database as a Service- In this type of service of cloud computing, the database is available as service to the user of particular cloud.

II.USE OF VIRTUALIZATION

With the increase in applications of virtualization concepts across a wide range of areas in computer science, the girth of the definition has been increasing even more. However, just for the discussions in this paper (Chiueh et al., 2005), we use the following relaxed definition:" Virtualization is a technology that combines or divides

computing resources to present one or many operating environments using methodologies like hardware and software partitioning or aggregation, partial or complete machine simulation, emulation, time-sharing, and many others". Although virtualization can, in general, mean both partitioning as well as aggregation, for the purposes of this paper, we shall concentrate on only partitioning problems (as these are much more prevalent). A virtualization layer, thus, provides infrastructural support using the lower-level resources to create multiple virtual machines that are independent of and isolated from each other. Sometimes, such a virtualization layer is also called Virtual Machine Monitor(VMM).Virtualization can, in general, mean both partitioning as well as aggregation, for the purposes of this paper, we shall concentrate on only partitioning problems (as these are much more prevalent). A virtualization layer, thus, provides infrastructural support using the lower-level resources to create multiple virtual machines that are independent of and isolated from each other. Sometimes, such a virtualization layer is also called Virtual Machine Monitor(VMM). Although traditionally VMM is used to mean a virtualization layer right on top of the hardware and below the operating system, we might use it to represent a generic layer in many cases. There can be in numerous reasons how virtualization can be useful in practical scenarios, a few of which are the following:

- Server Consolidation: To consolidate workloads of multiple under-utilized machines to fewer machines to save on hardware, management, and administration of the infrastructure

- Application consolidation: A legacy application might require newer hardware and/or operating systems. Fulfilment of the need of such legacy applications could be served well by virtualizing the newer hardware and providing its access to others.

- Sandboxing: Virtual machines are useful to provide secure, isolated environments (sandboxes) for running foreign or less-trusted applications. Virtualization technology can, thus, help build secure computing platforms.

- Multiple execution environments: Virtualization can be used to create multiple execution environments (in all possible ways) and can increase the QoS by guaranteeing specified amount of resources.

- Virtual hardware: It can provide the hardware one never had, e.g. Virtual SCSI drives, Virtual

- ethernet adapters, virtual ethernet switches and hubs, and so on.

- Multiple simultaneous OS: It can provide the facility of having multiple simultaneous operating systems that can run many different kind of applications.

- Debugging: It can help debug complicated software such as an operating system or a device driver by letting the user execute them on an emulated PC with full software controls.

- Software Migration: Eases the migration of software and thus helps mobility.

- Appliances: Lets one package an application with the related operating environment as an appliance.

- Testing/QA: Helps produce arbitrary test scenarios that are hard to produce in reality and thus eases the testing of software.

Virtualization, next to its core, is the ability to emulate hardware using software. From beginning to end the system initialization processes, several form of operating system still requirements to be booted from the hardware. This may be a full blow OS such as Linux or Windows, or it may be a stripped down OS specifically designed to provide virtualization, such as VMware's ESXi (which is a stripped down Linux operating system). In either case an operating system is first booted and then an emulation software stack is loaded which is referred to as a hypervisor. The hypervisor is the component which is responsible for emulating specific hardware configurations to guest operating systems. When a guest is loaded into a virtual machine (VM), the hardware that gets detected is the simulated hardware via the hypervisor, not the actual hardware itself. The guest OS is abstracted from the true hardware, adding a component of versatility. The hypervisor is capable of creating multiple simulated environments, or multiple VMs, which permits us to run multiple operating systems that may have slightly different hardware requirements. The various servers like web servers, database servers

and messaging servers are consolidated from 100 servers to few servers which help in TCO, power bill and cooling equipment cost reduction (Janakiram, 2012).

III. Quality Parameters for Virtualization

Gupta (2013) have segregated that the quality attributes for can be categorised into the following five categories:

1. Functional QoS attributes
2. Non-functional QoS attributes
3. Service function-related QoS attributes
4. Data-related QoS attributes
5. Security-related QoS attributes.

Functional QoS Attributes-

In case of the functional attributes, the three criteria are Performance, Throughput and Maintenance behaviour.

Performance-Because SaaS is provided as a service over a network, performance is a key indicator of the QoS provided to the consumer. Performance guidelines are defined as service deliverables in the service level agreement (SLA). These guidelines must be met for customer satisfaction and retention. Quality attributes for a performance indicator are:

- Response time
- Throughput
- Availability

Throughput- The volume of data processed by the service or application is termed throughput. Throughput can be calculated at the:

(a) Network level – For example, the amount of data transmitted over the network per unit time.

(b) Application level – For example, the number of transactions processed by an application in a given amount of time.

Maintenance Behaviour-Constant enhancements and upgrades are a necessary aspect of SaaS - along with continuous testing to ensure smooth and minimal service downtime. The most important parameters of this behaviour include:

(a) Upgrades – These are releases or enhancements to the software being provided as a service.

(b) Service support – It is around-the-clock support to ensure that the software is available with no or very little downtime.

Non-functional QoS Attributes-

This is the category of attributes in which the most important non-functional QoS attributes fall . These include the availability, reliability, reusability, customisability, `Exception Handling, Configuration Correctness, Maintainability and monitoring.`

Availability− The percentage of time software is available as a service will be a key factor for determining the QoS being provided by the vendor. The quality metrics for availability would involve:

• Infrastructure level (server downtime)
• Database level (database uptime, database downtime)
• Application level (application uptime, application downtime)

Reliability− To ensure the success of a SaaS implementation, reliability is a key quality factor to be considered. Proper attention should be given to the condition of the infrastructure- network and to the architecture, prevalent practices, and the monitoring tools available for the corresponding service reliability. Reliability is classified into three categories:

o Service reliability – Uninterrupted service must be available to the customer when desired.

o Application reliability – Reduced frequency of application failure makes it more reliable.

o Backup reliability – The backup solution provided for SaaS must be made reliable by following redundancy strategies or using other available methods.

Reusability− This feature of SaaS is defined by how common the software being used is in the software industry and how often it would meet the requirements of the customers. Because vendors would have to buy and maintain the software at their end, measuring this feature becomes all the more important. Reusability strongly influences the return on investments (ROI) being made by the customer. Quality attributes that would come into play for reusability are:

• Maintenance
• Complexity
• Infrastructural cost (licensing fees) • Reuse frequency of the software

Customizability- Customizability of software is the programmable attribute of software that determines the flexibility of the software for changes that fulfil customers' needs and requirements. In providing software as a service, customizability of software is a very important aspect for the vendors to take into account. In the absence of customizability, use of SaaS would be very limited and SaaS would not be able to meet the desired configurations and requirements of consumers.

Exception Handling- Exception handling for various special scenarios and error or failure conditions should be considered and accounted for. Robust software is critical for successful SaaS.

Configuration Correctness-While providing SaaS, it is very important to comply with the requirements of the customer. Consistency needs to be maintained with respect to infrastructure requirements, operational information and the user interface specified in the user agreement or the business requirement document.

Maintainability-Maintainability can be defined as the ease with which a service or a product can be maintained with respect to enhancements, changing requirements, and functionalities. Maintainability is performed at the following levels:

• Application level

• Service level

• Function level

Monitoring− Monitoring in cloud computing is very important because resources are being provided as service. Customers would want to keep track of all outages and track the status of the service being provided. In SaaS, monitoring would be important at various levels:

• Tenant level – this involves monitoring performance of the application at user level.

• Application level – this involves monitoring performance of the overall application. It has been termed "application performance monitoring".

• Network level – this monitoring at network level may involve maintaining log files.

Service Function QoS Attributes-

Multi−tenancy−Multi-tenancy is the operation of multiple instances of an application(s) in a shared environment. Implementation of multi-tenancy can take place at various levels.

• Application level – a SaaS application should allow multiple tenants while being efficient, secure, and robust.

• SaaS underlying infrastructure level – the infrastructure hosting the SaaS application should also support multi-tenancy.

• User interface level – multi-tenancy should be supported at the user interface level allowing the configuration of multiple tenants successfully and efficiently.

• Database level – this involves configuring a shared database while accounting for security concerns by using techniques such as data isolation or instance sharing.

Business-Based Attributes-

1. Support services – These are the level of support provided for the services provided by SaaS including execution and maintaining ACID (atomicity, consistency, isolation and durability) properties at transaction level.

2. Penalty rate – In the case of failure to carry out the agreed-upon services, the penalty rates to be paid by the vendor need to be predetermined and quoted in the service level agreement.

3. Compensation costs – Similarly, failure to deliver the software or software updates would result in providing the customer with compensation costs as per the service level agreement. These costs could be a certain fraction of the original executing costs or an amount previously decided upon and based on market standards.

Service−FunctionScalability． Scalability is the major challenge in providing and making good use of SaaS effectively and efficiently. The SaaS infrastructure provided by the vendor should be able to scale up or scale down, as required, with relative ease. The quality metrics that decide the QoS of a SaaS with respect to scalability are:

• Infrastructure level – The SaaS underlying infrastructure should be scalable with the changing requirements and needs of the customer.

• Platform level – The SaaS underlying platform should also support scalability with the changing requirements and enhancements desired by the customer.

• SaaS level – The service should also be scalable at SaaS level like number of users, number of servers.

Service-Function Connectivity−How well a service function is connected at the security level and its performance levels with respect to connectivity must be analyzed. The information being shared should be over secure connections and should be carried out efficiently without impairing performance levels. The QoS for service-function connectivity should be handled at two levels:

• Security level
• Performance level

Application Programming Interface (API) Transaction Behaviour- The implementation of APIs involving inputs, outputs, function calls is subject to various quality concerns and needs to be monitored for QoS at the following levels:

• Security level
• Performance level • Load/Stress level

Workflow Behaviour-The workflows in the application should function as specified in the business requirements. The decision flow and data flow should be consistent with the requirements. In addition, all requirements specified in the business-requirements document should be covered by the workflows while catering to following quality metrics:

• Function correctness
• Flow control
• Requirement coverage/mapping

Function Efficiency- The logic of functions should be developed, implemented, and executed with speed, accuracy, and efficiency as primary concerns. Function efficiency needs to be monitored at the following levels:

• Logic level
• Implementation level • Execution level

Service Recovery- This is the recovery from outages with minimum or no losses with respect to information and data. Recovery mechanisms should be defined in detail comprising of detailed instructions in case of an outage and regular mock drills should be conducted.

Data-Related QoS Attributes-

Data Integrity- The infrastructure (servers, applications, databases) providing SaaS is in a remote location and is accessed via a network. Therefore, data encryption, accuracy, and data backup options are sought-after quality attributes and are necessary for data integrity.

• Control Access – It enables user access to information based on rights and privileges.

• Accuracy – how accurate is the data

• Data backup – It is the storing or backing up data in case of outages

The ACID Attributes-Transactional aspects of SaaS need to comply with the requirements of distributed transactions that involve the ACID properties.

• Atomicity − Either all actions associated with a transaction are carried out successfully or none of them are performed.

• Consistency − Updates performed should be consistent throughout the system for all users.

• Isolation − Effect of transactions performed concurrently should take place once the transaction completes.

• Durability - If changes incurred by a transaction are performed successfully, they should be reflected in the system after being completed.

Database Scalability- This is the ability to increase database size and architecture in case of enhancements or changed requirements from the customer.

Data Recovery- This is the ability to recover data in case of a failure or outage scenario.

Security Related Attributes-

Service Security-Because it is remotely provided by means of a network, service security is very important in the SaaS industry and is usually detailed in the SLA. The various attributes involved in ensuring service security are:

• Control access (based on permissions and privileges)

• User security (like password protection)

• Web security (like email security)

Data Security- This refers to securing data by means of providing physical security, encryption methods, and data backups in remote geographic locations.

• Database security

• Off-site data backup

Service Confidentiality- Remote access is a major issue among SaaS consumers and so service confidentiality is a major concern. The service level agreement addresses this concern and provides a concrete guideline to be followed by vendors.

• Message level confidentiality

• Data confidentiality

• Data encryption

IV.CONCLUSION

In this paper a study has been carried out to identify the quality parameters for virtualization in cloud computing. The various quality parameter has been listed with reference to various layers of services available in cloud computing.Virtualization describes a technology in which an application, guest operating system or data storage is abstracted away from the true underlying hardware or software. A key use of virtualization technology is server virtualization, which uses a software layer called a hypervisor to emulate the underlying hardware. This often includes the CPU's memory, I/O and network traffic. The guest operating system, normally interacting with true hardware, is now doing so with a software emulation of that hardware, and often the guest operating system has no idea it's on virtualized hardware. While the performance of this virtual system is not equal to the performance of the operating system running on true hardware, the concept of virtualization works because most guest operating systems and applications don't need the full use of the underlying hardware (Pek et al., 2013). This allows for greater flexibility, control and isolation by removing the dependency on a given hardware platform. While initially meant for server virtualization, the concept of virtualization has spread to applications, networks, data and desktops.

REFERENCES

[1] Bouley, D., & Brey, T. (2009), “Fundamentals of data centre power and cooling efficiency zones”, The Green Grid White Paper, 21, pp. 1–17.
[2] Brooks, D., &Martonosi, M. (2001), “Dynamic Thermal Management for High- Performance Microprocessors”, In High-Performance Computer Architecture, 2001, HPCA, the Seventh International Symposium on, IEEE, pp. 171-182.
[3] Brown, D. J., & Reams, C. (2010), “Toward Energy-Efficient Computing”, Communications of the ACM, Vol. 53, No.3, pp. 50-58.
[4] Kop, R., & Carroll, F. (2012). Cloud computing and creativity: Learning on a massive open online course. European Journal of Open, Distance and E-learning, 15(2).
[5] Mell, P., &Grance, T. (2011). The NIST definition of cloud computing.
[6] Hayes, B. (2008). Cloud computing. Communications of the ACM, 51(7), 9-11
[7] Buyya, R., Broberg, J., &Goscinski, A. M. (Eds.). (2010). Cloud computing: Principles and paradigms (Vol. 87). John Wiley & Sons.
[8] Sotomayor, B., Montero, R. S., Llorente, I. M., & Foster, I. (2009). Virtual infrastructure management in private and hybrid clouds.IEEE Internet computing, 13(5), 14-22.

[9] Buyya, R., Yeo, C. S., Venugopal, S., Broberg, J., &Brandic, I. (2009). Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. Future Generation computer systems, 25(6), 599-616.

[10] Sbihi, B., Eddine, K., &Aknin, N. (2013). Towards a collaborative learning process based on the hybrid cloud computing and Web 2.0 tools. International Journal of Engineering and Technology, 1(2).

[11] Conde-González, M. Á., García-Peñalvo, F. J., Zangrando, V., García-Holgado, A., Seoane-Pardo, A. M., Alier, M., ... &Brouns, F. (2013). Enhancing informal learning recognition through TRAILER project. In Proceedings of the Workshop on Solutions that Enhance Informal Learning Recognition (WEILER 2013) (Vol. 1039, pp. 21-30). CEUR-WS.

[12] Pocatilu, P., Alecu, F., &Vetrici, M. (2010). Measuring the efficiency of cloud computing for e-learning systems. Wseas transactions on computers, 9(1), 42-51.

[13] Zaharescu, E., &Zaharescu, G. A. (2012). Enhanced virtual e-learning environments using cloud computing architectures. International Journal of Computer Science Research and Application, 2(1), 31-41.

[14] Zissis, D., &Lekkas, D. (2012). Addressing cloud computing security issues. Future Generation computer systems, 28(3), 583-592.

[15] Vaquero, L. M., Rodero-Merino, L., Caceres, J., & Lindner, M. (2008). A break in the clouds: towards a cloud definition. ACM SIGCOMM Computer Communication Review, 39(1), 50-55.

[16] Buyya, R; Pandey, S; Vecchiola, C (2009), "Cloudbus Toolkit for Market-Oriented Cloud Computing, Cloud Computing", Proceedings, 2009, 5931, No. 21, pp. 24 – 44

[17] Cai C, Wang L, Khan SU, Jie T (2011), "Energy-Aware High Performance

[18] Addis B., Ardagna D., Panicucci B., Squillante M., Zhang L. (2013), "A Hierarchical Approach for the Resource Management of Very Large Cloud Platforms", IEEE Transactions on Dependable and Secure Computing, Vol. 10, No. 5, pp. 253-272, 2013

[19] Ali, Q., I., Alnawar, J., Mohammed (2013), "Optimization of Power Consumption in Cloud Data Centres Using Green Networking Techniques", Received: 2-10-2013 Accepted: 9- 6-2013, Al-Rafidain Engineering, Vol.22, No. 2, March 2014, pp. 13-27.

[20] Andersen, D. G., Franklin, J., Kaminsky, M., Phanishayee, A., Tan, L., &Vasudevan, V. (2009, October),"FAWN: A Fast Array Of Wimpy Nodes", In Proceedings of the ACM SIGOPS 22nd symposium on Operating systems principles, ACM, pp. 1-14.

[21] Ardagna D., Trubian M., Zhang L. (2007), "SLA Based Resource Allocation Policies in Autonomic Environments", Journal of Parallel and Distributed Computing, Vol. 67, No. 3, pp. 259-270, 2007

[22] Ardagna, D., Panicucci, B., Trubian, M., & Zhang, L. (2012), "Energy-Aware Autonomic Resource Allocation in Multi-Tier Virtualized Environments", Services Computing, IEEE Transactions on, Vol. 5, No. 1, pp. 2-19.

[23] Armbrust, M., Fox, A., Griffith, R., Joseph, A., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I. &Zaharia, M.(2009), "Above the Clouds: A Berkeley View of Cloud Computing", Electrical Engineering and Computer Sciences University of California at Berkley, USA, Technical Report UCB/EECS-2009-28, February 10, 2009, pp. 1-23,

[24] Auer, G., Giannini, V., Desset, C., Godor, I., Skillermark, P., Olsson, M., ...&Fehske, A. (2011). How Much Energy Is Needed To Run A Wireless Network?" IEEE Wireless Communications, Vol. 18, No. 5, pp.40-49.

[25] Baliga, J., Ayre, R. W., Hinton, K., & Tucker, R. S. (2011), "Green Cloud Computing: Balancing Energy in Processing, Storage, and Transport", Proceedings of the IEEE, Vol. 99, No. 1, pp.149-167.

[26] Baliga, J., Ayre, R., Hinton, K., & Tucker, R. S. (2011), "Energy Consumption in Wired and Wireless Access Networks", IEEE Communications Magazine, Vol. 49, No.6, pp.70-77.

[27] Baliga, J., Ayre, R., Hinton, K., Sorin, W. V., & Tucker, R. S. (2009), "Energy Consumption in Optical IP Networks", Journal of Lightwave Technology, Vol. 27, No. 13, pp. 2391-2403

[28] Barroso, L. A. (2005), "The Price of Performance", Queue, Vol. 3, No. 7, pp. 48–53, 2005.

[29] Barroso, L. A., &Hölzle, U. (2007), "The Case for Energy-Proportional Computing", Computer, pp. 33–37.

[30] Barroso, L. A., Clidaras, J., and Hölzle, U. (2013), "The Datacenter as A Computer: An Introduction to the Design of Warehouse-Scale Machines", Synthesis lectures on computer architecture, Vol. 8, No. 3, pp. 1-154.

[31] Beaty, D., & Schmidt, R. (2004), "Back to the Future: Liquid Cooling: Data Center Considerations", Ashrae Journal, Vol. 46, No. 12, pp 1-42.

[32] Beloglazov A., Abawajy J., Buyya R. (2012), "Energy-Aware Resource Allocation Heuristics for Efficient Management of Data Centers for Cloud Computing", Future Generation Computer Systems, Vol. 28, No. 5, pp.755–768.

[33] Beloglazov, A., Buyya, R., Lee, Y. C., &Zomaya, A. (2011), "A Taxonomy and Survey of Energy-Efficient Data Centers and Cloud Computing Systems", Advances in computers, Vol. 82, No. 2, pp. 47-111.

[34] Benini, L., Bogliolo, A., & De Micheli, G. (2000), "A Survey of Design Techniques for System-Level Dynamic Power Management", IEEE transactions on very large scale integration (VLSI) systems, Vol. 8, No. 3, pp. 299-316.

[35] Bernsmed, K., Jaatun, M. G., Meland, P. H., &Undheim, A. (2011, August), "Security SLAs for Federated Cloud Services", In Availability, Reliability and Security (ARES), 2011 Sixth International Conference on, IEEE, pp. 202-209.

[36] Bianchini, R., &Rajamony, R. (2004), "Power and energy management for server systems", Computer, Vol. 37, No. 11, pp. 68-76

[37] Bianco P.,Lewis G. A., Merson P. (2008), "Service Level Agreements in Service- Oriented Architecture Environments", Software Architecture Technology Initiative, Integration of Software-Intensive Systems Initiative, Software Engineering Institute, CMU/SEI-2008-TN-021,September 2008, pp. 1-49, http://www.sei.cmu.edu

[38] Bianzino, A. P., Chaudet, C., Rossi, D., &Rougier, J. L. (2012). A Survey of Green Networking Research", IEEE Communications Surveys & Tutorials, Vol. 14, No. 1, pp. 3-20.

[39] Blough, B., Bean, J., Jones, R., Patterson, M., Jones, R., & Salvatore, R. (2011), "Qualitative Analysis of Cooling Architectures for Data Centers", The Green Grid White Paper, pp. 1-30, www.thegreengrid.org.

[40] Bolla, R., Bruschi, R., Davoli, F., &Cucchietti, F. (2011), "Energy Efficiency in the Future Internet: A Survey of Existing Approaches and Trends in Energy-Aware Fixed Network Infrastructures. IEEE Communications Surveys & Tutorials, Vol. 13, No. 2, pp. 223-244.

[41] He, Z., & Liang, G. (2012, October), "Research and Evaluation Of Network Virtualization In Cloud Computing Environment", In Networking and Distributed Computing (ICNDC), 2012 Third International Conference on, pp. 40-44, IEEE.

[42] Hähnel, M., Döbel, B., Völp, M., and Härtig, H. (2013, January), "eBond: Energy Saving In Heterogeneous RAIN", In Proceedings of the fourth international conference on Future energy systems, ACM. Affect, pp. 193-202.

[43] White Paper (2013, April), "The Power of Wireless Cloud", Centre for Energy- Efficient Telecommunications (CEET), the University of Melbourne, pp. 1-22.

[44] Ghribi, C., Hadji, M., &Zeghlache, D. (2013, May),"Energy Efficient VM Scheduling for Cloud Data Centers: Exact Allocation And Migration Algorithms", In Cluster, Cloud and Grid Computing (CCGrid), 2013 13th IEEE/ACM International Symposium on, IEEE, pp. 671-678.

[45] Huang, W., Li, X., & Qian, Z. (2013, July), " An Energy Efficient Virtual Machine Placement Algorithm with Balanced Resource Utilization", 2013 Seventh International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, pp. 313-319.