Volume 6, Issue 4, April 2016 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

A SLA Based Autonomic Resource Management Model for Cloud Computing Architecture

G. Sobers Smiles David
M.C.A.,M.Phil., PG Dept.,of C.Sc.,
Bishop Heber College(Autonomous),
Trichirappalli, Tamilnadu, India

Dr. R. Anbselvi
M.Sc.,MPhil.,Ph.D, PG Dept., of C.Sc.,
Bishop Heber College(Autonomous),
Trichirappalli, Tamilnadu, India

Abstract—Cloud Computing has revolutionized the way data is stored and accessed. It also promises the capability to offer quality computing services which could be subscribed. With the number of users on the high, there is a need to cater to different types of consumers and their requirements. The data centers offering services are yet to embrace SLA based resource provisioning. So, the issues such as autonomic resource management, customer driven service management, and computational risk management are yet to be addressed. This paper presents the issues, challenges, and an architecture for SLA based resource provisioning. The proposed architecture combines Virtualization technology and the provisioning policies to maximize the provisioning of resources with minimal effort.

Keywords—Cloud Computing, Service Level Agreement, Atonomic Resource Provisioning, University Clouds

I. INTRODUCTION

Cloud Computing offers huge benefits to users in terms of cost, speed and efficiency. Gone are the days where users had to rely on investment to start a project. By providing on demand access to a shared pool of resources in a self service that is scaled dynamically and metered manner, cloud computing offers compelling advantages in cost, speed and efficiency[1]. Some of the other advantages include

- Just-in-time infrastructure
- More efficient resource utilization
- Usage based costing
- Reduced time to market

Technically, cloud offers Automation, auto-scaling, more efficient development cycle, improved testability, managing the overflow traffic, disaster recovery, and business continuity. Cloud computing could be termed as the new paradigm for provides dynamic creation of next generation data centers by pooling services of networked virtual machines. To realize the true potential of cloud computing the leading cloud vendors are deploying cloud centers across the globe to effectively counter failure. Data centers have become the backbone of companies to sustain their business operations.

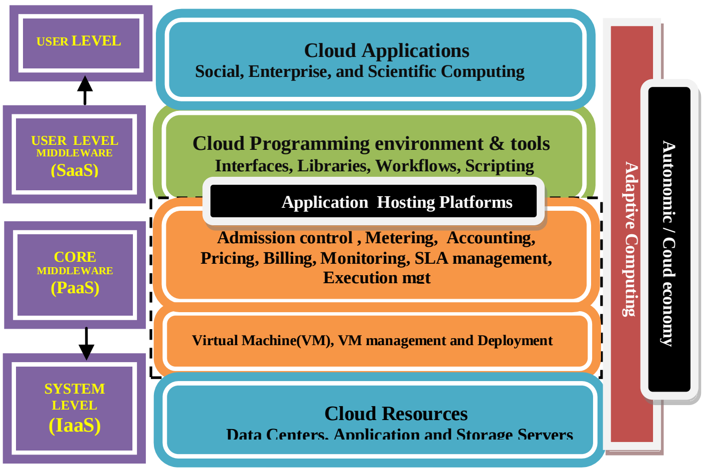

Fig. 1 Layered Cloud Computing Architecture

Also, global companies need faster response time and save time by distributing the workload across various data centers at a time[2]. Resource Management Systems (RMS) play a vital role in ensuring that the Service Level Agreements with clear QoS (Quality of Service) parameters are enforced. The RMS balances the service requests from the users and the expected service performance from the provider. System oriented approaches in resource allocation maximize system usage and job performance but fail to on-demand-service computing. Before cloud computing, ICT administrator's job was easy, as the only objective of resource provisioning was the performance. The complexity involved in resource provisioning has grown exponentially manifold due to increased complexity [4].

This paper is organized as follows. In section A, the issues in current Cloud Computing Architectures is discussed. In section B, Resource allocation strategy is presented. Section C gives an overview of the challenges in SLA based Resource provisioning. Chapter II presents the Resource Allocation Architecture and its entities. Chapter III presents the Autonomic Resource Provisioning Architecture and the proposed SLA based Resource provisioning Algorithm and discusses the experimental setup and analyses the results. Chapter IV presents the Conclusion and Future directions.

A. Issues with current Cloud Computing Architectures

Cloud computing is one of the research areas that continues to evolve each day with rapidly changing demands and requirements. Because of this scenario, there is a compelling lack of standard methodologies, tools and applications to tackle this demand[21]. In higher education, Cloud computing solutions have to host a variety of applications to cater to the ever growing demand of the student community and the quest for excellence by the higher educational institutions. The cost involved in developing, and maintaining such a facility needs to be brought down, in order to allow access to quality content to the lower strata of society. It is high time, the governments bring down the bandwidth cost to allow the higher educational institutions to offer university cloud computing solutions at a reasonable cost.

The traditional resource management model is not capable of processing the task of resource assignment and allocating resources dynamically [16]. As Cloud offers the capability to access information anytime, anywhere and anyhow, it is difficult for a cloud service provider to dynamically allocate resources efficiently. The need of the hour is a customer centric resource management system that is market oriented and is capable meeting the needs of demand.

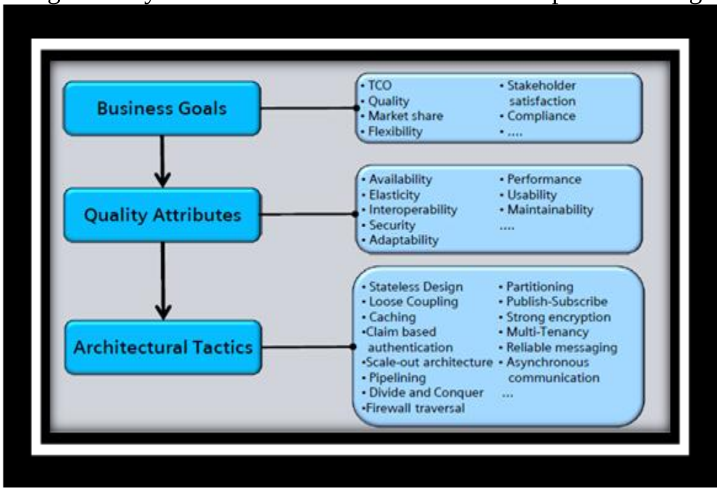

Fig. 2 Cloud Computing Architectural tactics

The emerging service market's success is directly dependant on customer satisfaction [6]. The factors involved in service quality directly impact customer satisfaction. So, cloud service providers are under immense pressure to meet the customer's needs. Service Level Agreement based resource management methodologies are needed in order to satisfy consumers, in this case, the students. These proposed methodologies must have the capability to manage the service paradigms such as service request, response, customer feedback, pricing model and effectively manage the ever growing customer demand with the limited resources autonomically[9].

B. Resource Allocation strategy-Definition

Resource Allocation is the process of procuring resources and then managing these resources by allocating them to applications that need them.

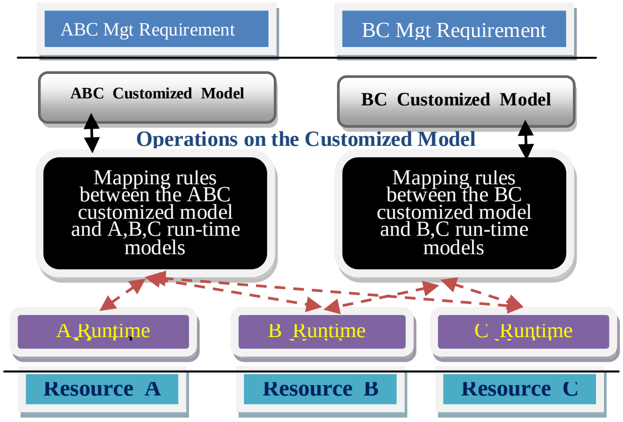

Fig. 3 Architectural Model for managing resources

a).Challenges and Benifits of Architecture approach

The huge challenge in managing cloud resources emanates from the diverse nature of cloud applications and their resources. Cloud resources include the web servers, memory, storage, network, CPU, application servers, and virtual machines. Virtualization offers scope for solutions to manage resources but increases complexity [9].

One of the important challenges in managing resources lies with the administrators for they have to be familiar with the interfaces (API's) and then write programs on them. In this model, the runtime model of the resource is constructed and then a synchronization model is constructed to avoid ambiguity. Mapping rules are needed to specify management requirement and to ensure transformation of models.

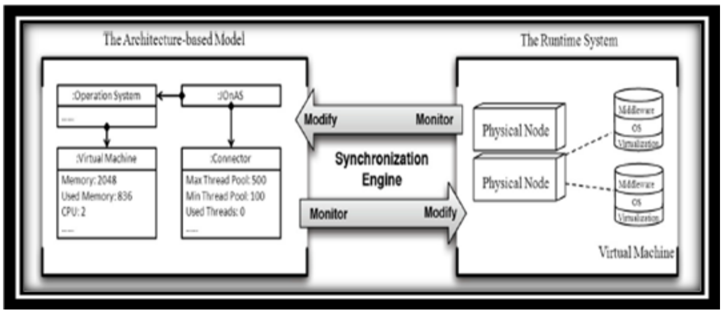

Fig. 4 Synchronization of models

C. Challenges of SLA based resource provisioning

The SLA based resource provisioning's foremost challenge is differentiating and satisfying the service request of users. Secondly, ensuring customer satisfaction has become a crucial factor. Customer satisfaction includes, provisions for feedback, realizing the specific needs of the customer, security against risks, etc.,

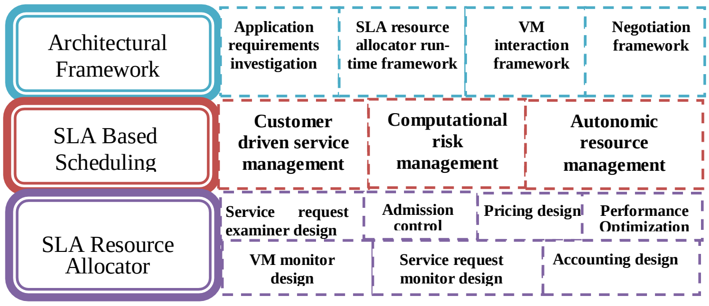

Fig. 5 Challenges in SLA based resource provisioning

The service requirement of users constantly changes with time. So, the autonomic resource provisioning mechanism, must perform the following

- ❖ continuous monitoring of current service requests and self manage the resources whenever there is a changes
- ❖ handling incoming service requests and amending them
- ❖ adjusting allocation, schedules and prices in case of amendment
- ❖ automatic configuring for new requests

II. RESOURCE MANAGEMENT SYSTEM

A. Definition and Significance:

There are 4 entities in the proposed system.

- ❖ Cloud Consumer:
  Users who submit service request to the service provider. The provider then acts on the request.
- ❖ SLA based Resource Manager:
  The resource manager is responsible for resource discovery, resource provisioning, resource allocation, Monitoring, and gathering feedback from the consumer. These tasks are accomplished by the following sub-modules.
- ❖ Admission controller:
  When a request is received, this mechanism determines whether to accept or reject the request based on QoS parameters and requirements. It also considers previous usage and can accord priority.
- ❖ Finance controller:
  Depending on the timing of the request, i.e peak time, and off-peak time, this mechanism decides on the price to be charged. Also, the availability and demand for a resource, is also taken into consideration for varying the price. i.e fixed or varying price. This mechanism also keeps track of the allocated resources, so as to charge the consumer based on their usage.
- ❖ Machine controller:
  This mechanism keeps track of the availability of virtual machines.

- Dispatch controller:
  This mechanism is used to start the execution of service requests that are admitted on allocated virtual machines.
- Service controller:
  This mechanism keeps track of the progress of all the service requests that are in execution.

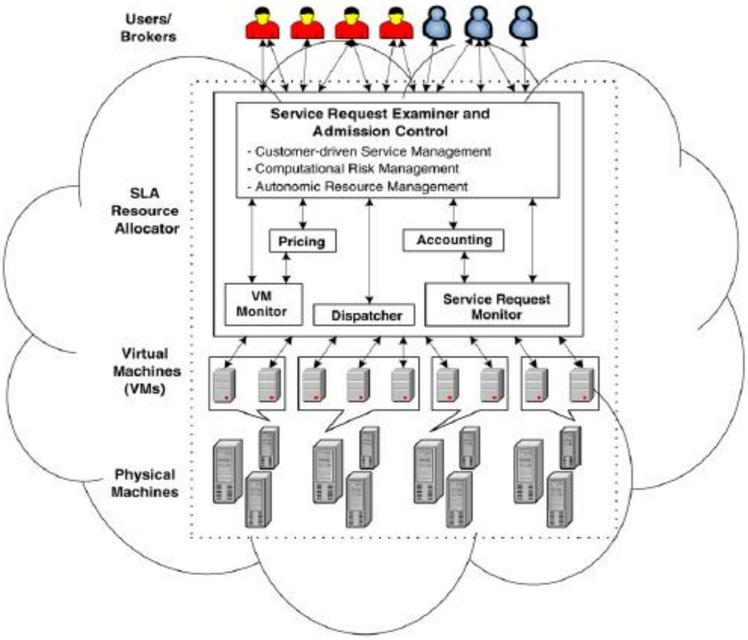

Fig. 6 Resource Management System Architecture

Fig.6 shows the Architecture of resource management system in cloud computing.

- Physical Machines:
  This mechanism provides storage, compute facility and manages the data centers.
- Virtual Machines:
  With a single physical machine, multiple virtual machines can be instantiated using this mechanism. Virtual machines have the capability to concurrently run on different operating system environments.

B. Virtualization

In a virtualized environment, a large pool of logical network instances, called virtual networks exist along with the infrastructure of physical network. A virtual network comprises of virtual nodes that connect via virtual paths, constructing a virtual topology [8]. Each virtual node is hosted on a physical node called host, and each virtual path is established via a physical path. The virtual networks are logically separated from each other, and are managed by a separate entity. From an architectural point of view, virtualization satisfies the following design goals.

- Concurrent existence of multiple virtual networks in the same environment
- Options to create new virtual networks on top of existing virtual networks
- Flexibility to execute ad-hoc network topology and customized control protocols
- Options to have complete administrative control over a virtual network
- Logical isolation from each virtual networks
- Heterogeneousness of physical infrastructure

For realizing the resource discovery framework, a distributed peer to peer architecture is proposed. The proposed framework is service oriented and designed to provide network as a service to the consumers. The core of this framework is the Information Module that keeps track of physical and virtual resources and handles the mapping between them. The objective of this framework is efficient resource discovery and seamless management of physical and virtual resources.

The process of embedding of virtual networks consists of six phases namely, Resource Description, Publication, Selection, Negotiation, Allocation, and Management. These phases perform unique functionalities and collaborate between them vis-à-vis different roles.

- Resource Description:
  This phase provides information about functional attributes such as node, link, operating systems and non-functional attributes such as bandwidth, and capacity.
- Resource Publication:
  Once the information is described in the previous phase, they are published in repositories for efficient information retrieval.
- Resource Selection:
  In this phase, depending on the needs of the request, the best resource is selected by multi objective based selection.

Fig.7 shows the different phases in virtual network embedding process.

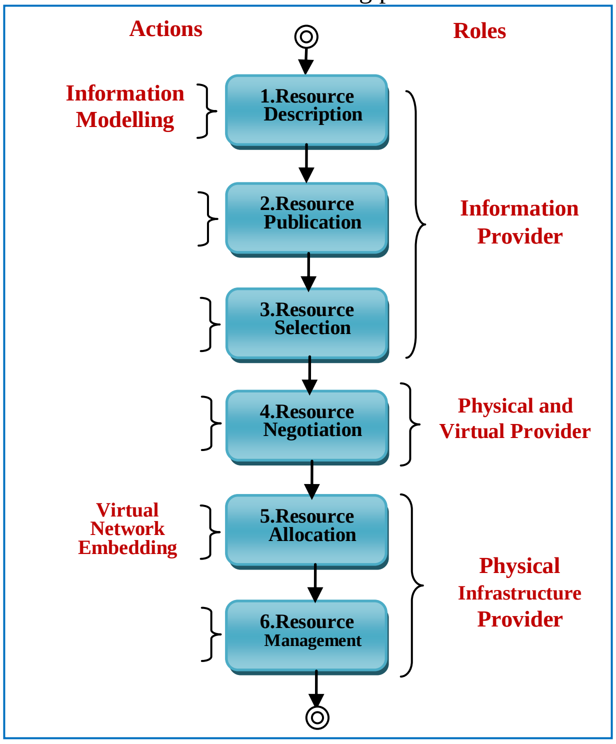

Fig. 7. Distributed Peer to Peer Resource Provisioning Architecture

- Resource Negotiation:
  In the negotiation phase, the issues of cost and quality for SLA (Service Level Agreements) are negotiated..
- Resource Allocation:
  In the allocation phase, the selected virtual resources are mapped onto their physical counterparts.
- Resource Management:
  In this phase, the allocated resources are managed for dynamic variations in the virtual networks and may be subjected to various resource management strategies such as dynamic topologies adoption and virtual nodes migration.

III. AUTONOMIC RESOURCE PROVISIONING MODEL

Autonomic management [2], [6] is the most desired feature in any distributed computing environment such as clouds. Autonomic systems are self managing systems with abilities such as self-healing, self-improving, self-regulating, and self-protecting. Preliminary investigation in the development of autonomic resource provisioning model is already made by Academia and Industry. Parashar and Harini [5], provided an overview of early autonomic systems in storage management(OceanStore[7], Storage Tank[8]), Computing resources(Oceano[18]), and DBMS (Smart DB2[10]). One of the main beneficiary of Autonomic systems is Computing grids, which has benefited the maximum]12]. CometCloud[11] provided an infrastructure model for automatic management of workflow applications in Cloud. Other works [13],[14],[15] explored in depth, the resource provisioning in cloud applications.

However, these works do not offer an integrated environment for cost effective autonomic resource provisioning with efficient security features. Amazon's Elastic MapReduce provides options for its customers to add or remove nodes depending upon the changing needs of their job flow. But, this service does not provide autonomic provisioning.

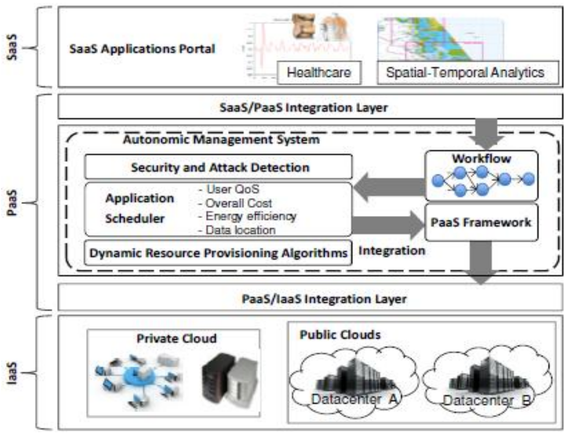

Fig. 8. Autonomic Resource Provisioning Architecture

Fig.8 shows the architecture for autonomic resource provisioning. This architecture comprises of the following architectural components.

- SaaS Applications Portal:
  Using this service, users can submit their request and avail the requested service. This portal acts as the SaaS layer.
- Autonomic Management System:
  Using this layer, the resources are managed autonomically. This layer also includes security features. This layer acts as the PaaS layer.
- IaaS Layer:
  This layer is used to provide resources based on user requirements whenever needed by the consumers, in this case the students.

A. Autonomic Resource Provisioning Algorithm

The algorithm is prediction based and uses previous history for effective provisioning of resources. The cost of a resource provisioned depends on the time duration for which it is used and the type of the resource. In a higher educational institution's cloud, students who request for services can be prioritized into two levels. i.e Basic and Advanced. Basic users are those who avail services for a shorter duration of time and Advanced users are those who avail services for a longer duration of time. The price of basic services is planned to be low compared to the advanced. Higher priority can be accorded to the advanced users based on supply and demand. During peak times, users can be priced slightly higher irrespective of the priority. The algorithm uses prediction to manage the supply and demand.

B. Algorithm

Step 1: Initiate the cloud resources

Step 2: Admit user requests and check user credentials. If valid user, then process the request, else reject the request.

Step 3: Check for previous history of request, if new, then assign priority based on the time duration of the service requested, else use the previous history to accord priority.

Step 4: Check for resource status and peak time. If peak time, then assign resource accordingly based on priority, else, assign resource irrespective of priority on a First come First served basis.

Step 5: Check the SLA parameter and monitor the allocated resources and availability of resources. If supply > demand, maintain status quo, else make changes in resource allotment by provisioning more virtual machines.

Step 6: Check the status of allocated resources. If usage time > threshold value & demand > threshold value, terminate the service. Calculate the price, by subtracting the penalty charge for abrupt termination. Update the resource availability database.

Step 7: Release resources which have completed the task and check whether SLA parameters are met. Calculate usage time and based on priority, charge the user. Update resource availability database.

Step 8: Check demand and previous history of requests for request patterns. If request pattern low and demand low, terminate virtual machines based on existing requests and patterns.

Fig. 9. Autonomic Resource Provisioning Algorithm

C. Advantages of the prediction based Algorithms

- Since the algorithm is prediction based, it is possible to optimize the resource provisioning and bring down the cost involved.
- Since the provisioning model is constantly updated based on the request patterns, the resources are scaled up or down according to the user requirements, thereby ensuring SLA.
- Truly a pay-per-use model, as pricing includes penalty for incomplete requests. Penalty is a SLA component.
- QoS parameters are ably met, as customer satisfaction is expected to be high. Customer Satisfaction is also a SLA component.
- Although provisioning of resources is a NP hard optimization problem, the proposed algorithm in the model efficiently handles the device management.
- The strategy followed in the model is prediction based, i.e., using previous request patterns for decision making in allocation of resources, the problem of resource management is smoothly done.

D. Predicting Future Resource Needs:

The resource needs of a higher educational institution needs to be constantly reviewed in order to handle resource management efficiently. This can be done by parsing the list of pending requests and studying the behaviour of past requests. The prediction can be done by using the following formula.

$E(t)= \alpha \times [E(t-1)] + [(1-\alpha)] \times O(t), \ 0 <= \alpha <= 1$
Where, E(t) – Estimated load at time t
O(t) – Observed load at time t
α- Tradeoff between stability and responsiveness

E. Results

The evaluation of the proposed algorithm was carried out in CloudSim tool. Deadline defined by the user is set as the SLA parameter. The experimental setup consisted of 5 static virtual machines, with the following configuration. 32 bit platform, 16 GB memory, 160 GB of instance storage and Rs.50 per instance per hour. For the experiment a CPU intensive application is used. Each job consisted of 100 tasks and the execution time of each task was set as 3 minutes. Initially, the job was executed sans SLA parameter and then the SLA parameter was included. The deadline was changed for each iteration. The deadlines used in the experiment are 15 mts, 30 mts and 45 mts. Table 1 depicts the results of the experiment.

Table 1- Experimental Results

	Initial static machines	Machines provisioned as per the needs	Job execution time	Extra Price
Without QoS	**5**	**0**	**1:05:52**	**Rs.0**
45 mts	**5**	**2**	**0:40:04**	**Rs.8**
30 mts	**5**	**6**	**0:27:16**	**Rs.38**
15 mts	**5**	**20**	**0:14:11**	**Rs.93**

The results show that, the QoS requirements are efficiently met by the algorithm. To meet the user requirements, the resource management model was able to provision more number of virtual machines and finish the execution within the stipulated time. Also, as the user requirement goes up, so too the machines provisioned, thereby meeting the SLA.

IV. CONCLUSION AND FUTURE DIRECTIONS

The growing trend of cloud adoption in higher educational institutions warrants new management models for this highly challenging computing environment. As the need for cloud computing solutions continue to grow exponentially, so too the threats. The proposed autonomic computing model is a humble first step in addressing the above problems. The dynamic nature of autonomic provisioning is found to be able to satisfy the QoS (Quality of Service) requirements of the consumer, resulting in improved efficiency in managing resource provisioning and optimized consumption of energy. Also, the proposed algorithm employs previous history of requests and is prediction based. This helps the model to identify malicious requests and thereby prevents DDoS attacks. Also, it does away with wastage of budget and energy consumption due to these malicious requests. The SLA obligations are thoroughly met, thereby resulting in increased customer satisfaction. In future more QoS aware algorithms could be employed resulting in more energy consumption and security aware architecture could be realized.

REFERENCES

[1] Rajkumar Buyya, Rodrigo N. Calheiros, and Xiaorong Li, "Autonomic Cloud Computing: Open Challenges and Architectural Elements", *IEEE Digital Library,* EAIT, 2012.

[2] J.O. Kephart and D. M. Chess, The Vision of Autonomic Computing, *Computer*, 36(1):41-50, IEEE, Jan. 2003.

[3] K. Appleby, S. Fakhouri, L. Fong, G. Goldszmidt, Kalantar, S. Krishnakumar, D. P. Pazel, J. Pershing, and Rochwerger. "Oceano-SLA based management of a computing utility". *Proceedings of the 2001 IEEE/IFIP International Symposium on Integrated Network Management (INM 2001), Seattle, WA, USA*, 2001.

[4] R. Buyya, C. S. Yeo, and S. Venugopal,".Market oriented Cloud Computing: Vision, Hype and Reality for delivering IT services as computing utilities", *Proceedings of 10th International Conference on High Performance Computing*, 2008.

[5] H. Liu, V. Bhat, M. Parashar and S. Klasky," An Autonomic Service Architecture for Self-Managing Grid Applications", *Proceedings of the 6th IEEE/ACM International workshop on Grid Computing,*USA,2005

[6] R. Ranjan, R. Buyya, and M. Parashar, "Autonomic Cloud Computing: Technologies, Services, and Applications, Concurrency and Computation: Practice and Experience", 24(9):935-937, Wiley Press, New York, USA, June 2012.

[7] J. Kubiatowicz, D. Bindel, Y. Chen, S. Czerwinski, P.Eaton, D. Geels, R. Gummadi, S. Rhea, H. Weatherspoon, W. Weimer, C. Wells, and B. Zhao. OceanStore: an architecture for global-scale persistent storage, *ACM SIGPLAN Notices*, 35(11):190-201, ACM, Nov. 2000.

[8] J. Menon, D. A. Pease, R. Rees, L. Duyanovich, and B. Hillsberg. IBM Storage Tank—A heterogeneous scalable SAN file system, *IBM Systems Journal*, 42(2):250-267, IBM, Apr. 2003

[9] Mayanka Katyal and Atul Mishra, Application of Selective Algorithm for Effective Resource Provisioning In Cloud Computing Environment, *International Journal on Cloud Computing: Services and Architecture (IJCCSA) ,Vol. 4, No. 1,* February 2014

[10] G. M. Lohman and S. S. Lightstone, SMART: making DB2(more) autonomic. *Proceedings of the 28th International Conference on Very Large Data Bases (VLDB 2002)*, Hong Kong, China, 2002

[11] H. Kim, Y. el-Khamra, I. Rodero, S. Jha, and M. Parashar, Autonomic Management of Application Workflows on Hybrid Computing Infrastructure, *Scientific Programming* 19(2):75-89, IOS, June 2011.

[12] J. Kaufman, T. Lehman, G. Deen, J. Thomas, OptimalGrid -autonomic computing on the Grid, IBM, 2003.

[13] J. Chen, C. Wang, B. Zhou, L. Sun, Y. Lee, A. Zomaya,Tradeoffs Between Profit and Customer Satisfaction for Service Provisioning in the Cloud, *Proceedings of the 20th* ACM International Symposium on High Performance Distributed Computing (HPDC 2011), San Jose, USA, 2011

[14] A. Goscinski and M. Brock, Toward dynamic and attributebased publication, discovery and selection for Cloud computing, *Future Generation Computer Systems*, 26(7): 947-970, Elsevier, Jul. 2010.

[15] B. Bethwaite, D. Abramson, F. Bohnert, S. Garic, C.Enticott, T. Peachey, Mixing the Grid and Clouds: Highthroughput Science using the Nimrod Tool Family, In: *Cloud Computing: Principles, Systems and Applications*,Antonopoulos and Gillam (eds), Springer, 2010..

[16] Sleiman Rabah, May El Barachi, Nadjia Kara, Rachida Dssouli and Joey Paquet A service oriented broker-based approach for dynamic resource discovery in virtual networks, *Rabah et al. Journal of Cloud Computing: Advances, Systems and Applications*, (2015) 4:3

[17] Lei Ying,Siva Theja Maguluri and R. Srikant,Heavy Traffic Optimal Resource Allocation Algorithms for Cloud Computing Clusters, *dl.acm.org*, 2012

[18] K. Appleby, S. Fakhouri, L. Fong, G. Goldszmidt, M. Kalantar, S. Krishnakumar, D. P. Pazel, J. Pershing, and B.Rochwerger. Oceano-SLA based management of a computing utility. Proceedings of the 2001 IEEE/IFIP International Symposium on Integrated Network *Management (INM 2001)*, Seattle, WA, USA, 2001.

[19] Swarupa Irugurala, Dr.K.Shahu Chatrapati, Various Scheduling Algorithms for Resource Allocation In Cloud Computing, *The International Journal Of Engineering And Science (IJES), Volume 2, Issue 5,Pages 16-24,2013*

[20] Zhen Xiao, Weijia Song, and Qi Chen, Dynamic Resource Allocation using Virtual Machines for Cloud Computing Environment, *IEEE TRANSACTION ON PARALLEL AND DISTRIBUTED SYSTEMS. VOL. 24 , NO. 6* YEAR 2013,

[21] Hadi Goudarzi and Massoud Pedram, Multi-dimensional SLA- based Resource Allocation for Multi-tier Cloud\Computing Systems, *ACM digital Library*, 2012

[22] Mansoor Alicherry, Network Aware Resource Allocation in Distributed Clouds, *Proceedings IEEE INFOCOM*,2012.

[23] Brett Snyder1, Jordan Ringenberg, Robert Green, Vijay Devabhaktuni and Mansoor Alam1Evaluation and design of highly reliable and highly utilized cloud computing systems, Snyder *et al. Journal of Cloud Computing Advances, Systems and Applications* (2015) 4:11

AUTHOR PROFILE

Prof. G. Sobers Smiles David, is working as Asst., Professor and Head, PG Department of C.S, Bishop Heber College (Autonomous), Trichirappalli. He has got 15 years of U.G and P.G teaching experience. He has got his M.C.A from Bishop Heber College in 2001. He has got his M.Phil in C.S from Bharathidasan University in 2004. Currently he is pursuing Ph.D in Bharathidasan University. His areas of interest include Cloud Computing, Computer Networks, Cryptography and Network Security, Pervasive Computing, and distributed computing.

Dr. R. Anbuselvi is working as Asst., Professor, PG Department of C.S, Bishop Heber College (Autonomous), Trichirappalli. She has got 20 years of UG and PG teaching experience. She has been awarded Ph.D in Computer Science through Mother Teresa Women's University, Kodaikkanal. Her research interest includes Artificial Intelligence, Data mining and Warehousing, Distributed Systems and Cloud Computing.

