IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 12, December 2015

Survey on Different Auto Scaling Techniques in Cloud Computing Environment

Pranali Gajjar[1], Brona Shah[2]

Silver Oak College of Engineering and Technology, Gujarat Technological University, S. G. Highway, Gota, Ahmedabad, India[1, 2]

Abstract: Cloud computing is a recent technology trending that help companies in providing their services in a scalable manner. Hence, used this service capabilities required many procedures in order to get better performance. Cloud computing environments allow customers to dynamically scale their applications. The key problem is how to lease the right amount of resources, on a pay-as-you-go basis. The objective of this paper was to present a comprehensive study about the auto-scaling mechanisms available today. Auto-scaling techniques are diverse, and involve various components at the infrastructure, platform and software levels. Many techniques have been proposed for auto-scaling. We propose a classification of these techniques into five main categories: static threshold-based rules, control theory, reinforcement learning, queuing theory and time series analysis.

Keywords: Cloud Computing, Auto scaling, Auto scaling techniques.

1. INTRODUCTION OF CLOUD COMPUTING

Cloud computing is in its infant form and numerous definitions have been proposed by many scientists. Some of the definitions are, Buyya et al. defines, "A Cloud is a type of parallel and distributed system consisting of a collection of inter-connected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resource(s) based on service-level agreements established through negotiation between the service provider and consumers" [1].

The National Institute of Standards and Technology (NIST) defines, "A model for enabling 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. This cloud model promotes availability and is composed of five essential characteristics, three service models, and four deployment models" [2].

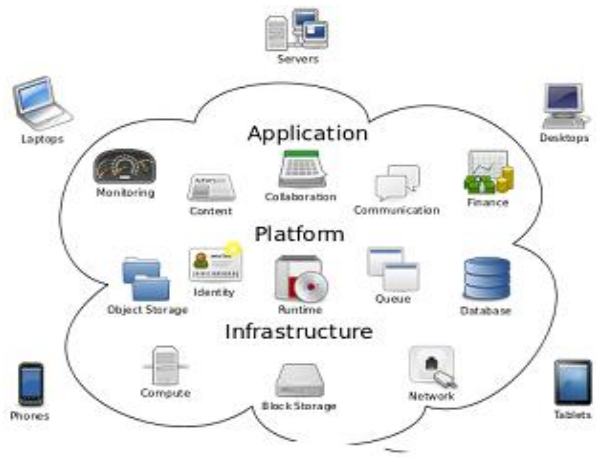

Figure 1: Cloud Computing

In brief cloud is essentially a bunch of commodity computers networked together in same or different geographical locations, operating together to serve a number of customers with different need and workload on demand basis with the help of virtualization. Cloud services are provided to the cloud users as utility services like water, electricity, telephone using pay-as-you-use business model. These utility services are generally described as XaaS (X as a Service) where X can be Software or Platform or Infrastructure etc. Cloud users use these services provided by the cloud providers and build their applications in the internet and thus deliver them to their end users. So the cloud users don't have to worry about installing, maintaining hardware and software needed. And they also can afford these services as they have to pay as much they use. So the cloud users can reduce their expenditure and effort in the field of IT using cloud services instead of establishing IT infrastructure themselves [3].

1.1 Characteristics of Cloud

- **On Demand Self Service:** On demand self-service refers to services requested by the customers to manage their own computing resources. These services are provided over the internet by a cloud provider to a customer who has requested for services and can manage their own computing resources.
- **Resource Pooling:** Cloud computing provides shared pool of resources that can be rapidly provisioned and can be released with minimal effort. Customers draw resources from remote data centers.
- **Broad Network Access:** To avail cloud computing services, internet works as a backbone of cloud computing. All services are available over the network and are also accessible through standard protocols using web enabled devices viz. computers, laptops, mobile phones etc.

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 12, December 2015

- **Rapid Elasticity:** As cloud computing provides services over the internet. These services can be managed or can be requested from cloud providers as per customer's requirement. Rapid elasticity refers to services which can be smaller or larger as per user requirement.
- **Measured Service:** These are services which are billed according to customer demand for definite services. As customers can request for services as per their own requirement, services are billed according to customer's demand.

1.2 Service Models of Cloud Computing

Cloud computing is a computational process in which services are delivered over a network using computing resources. The name 'cloud' symbolizes an abstraction for complex infrastructure it contains in system diagrams.

Figure 2: Service Model

- There are three main types of service models:
- Software as a Service (SaaS)
- Platform as a Service (PaaS)
- Infrastructure as a Service (IaaS)

1.2.1 Software-as-a-Service (SaaS):

In this multitenant service model, the consumers use application running on a cloud infrastructure. The cloud infrastructure including (servers, OS, Network or application etc.) is managed and controlled by the service provider with the user not having any control over the infrastructure. Some of the popular examples are SalesForce.com, NetSuite, IBM, Microsoft and Oracle etc.
Example for SaaS Company:

- Athena health
- Concur Technologies
- E2open

1.2.2 Platform-as-a-Service (PaaS):

With this model, the provider delivers to user a platform including all the systems and environments comprising software development life cycle viz. testing, deploying, required tools and applications. The user does not have any control over network, servers, operating system and storage but it can manage and control the deployed application and hosting environments configurations. Some popular PaaS providers are GAE, Microsoft's Azure etc.
Example for PaaS Company:

- Apprenda
- IBM
- Open shift

1.2.3 Infrastructure-as-a-Service (IaaS):

In this service model, the provider delivers to user the infrastructure over the internet. With this model, the user is able to deploy and run various software's including system or application softwares. The user has the ability to provision computing power, storage, networks. The consumers have control over operating systems, deployed applications, storage and partial control over network. The consumer has no control over underlying infrastructure. Some important IaaS providers are GoGrid, Flexiscale, Joyent, Rackspace etc.
Example for IaaS Company:

- Amazon web service
- At & t
- Ca technologies

1.3 Deployment models

Cloud systems can be deployed in four forms viz. private, public, community and hybrid cloud as per the access allowed to the users and are classified as follows:

1.3.1 Private cloud:

This deployment model is implemented solely for an organization and is exclusively used by their employees at organizational level and is managed and controlled by the organization or third party. The cloud infrastructure in this model is installed on premise or off premise. In this deployment model, management and maintenance are easier, security is very high and organization has more control over the infrastructure and accessibility.
Examples of Private Cloud:

- Eucalyptus
- Ubuntu Enterprise Cloud - UEC (powered by Eucalyptus)
- Amazon VPC (Virtual Private Cloud)
- VMware Cloud Infrastructure Suite
- Microsoft ECI data center.

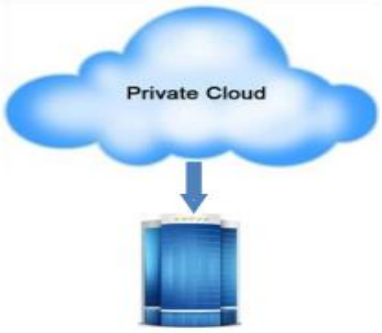

Figure 3: Private Cloud

1.3.2 Public cloud:

This deployment model is implemented for general users. It is managed and controlled by an organization selling cloud services. The users can be charged for the time duration they use the services. Public clouds are more vulnerable to security threats than other cloud models

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 12, December 2015

because all the application and data remains publicly available to all users making it more prone to malicious attacks. The services on public cloud are provided by proper authentication.
Examples of Public Cloud:

- Google App Engine
- Microsoft Windows Azure
- IBM Smart Cloud
- Amazon EC2

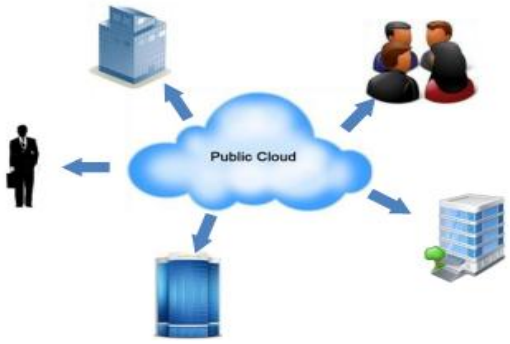

Figure 4: Public Cloud

1.3.3 Community cloud:

This cloud model is implemented jointly by many organizations with shared concerns viz. security requirements, mission, and policy considerations. This cloud is managed by one or more involved organizations and can be managed by third party. The infrastructure may exist on premise to one of the involved organization or it may exist off premise to all organizations.
Examples of Community Cloud:

- Google Apps for Government
- Microsoft Government Community Cloud

Figure 5: Community Cloud

1.3.4 Hybrid cloud:

This deployment model is an amalgamation of two or more clouds (private, community, public or hybrid). The participating clouds are bound together by some standard protocols. It enables the involved organization to serve its needs in their own private cloud and if some critical needs (cloud bursting for load-balancing) occur they can avail public cloud services.
Examples of Hybrid Cloud:

- Windows Azure (capable of Hybrid Cloud)
- VMware vCloud (Hybrid Cloud Services)

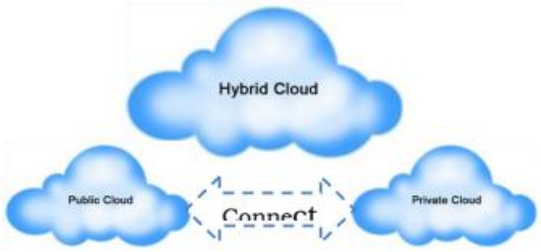

Figure 6: Hybrid Cloud

1.4 Advantages of Cloud Computing

Cloud computing offers many benefits and flexibility to its users. User can operate from anywhere at any time in a secure way. With the increasing number of web-enabled devices used now-a-days (e.g. tablets, smart phones etc.), access to one's information and data must be quick and easier. Some of these relevant benefits in respect to the usage of a cloud can be as follows:

- Reduces up-front investment, Total Cost of Ownership (TCO), Total Operational Cost (TOC) and minimizes business risks.
- Provides a dynamic infrastructure that provides reduced cost and improved services with less development and maintenance cost.
- Provides on-demand, flexible, scalable, improved and adaptable services on pay-as-you go model.
- Provides consistent availability and performance with automatically provisioned peak loads.
- Can recover rapidly and has improved restore capabilities for improved business resiliency.
- Provides unlimited processing, storage, networking etc. in an elastic way.
- Offers automatic software updates, Improved Document Format Compatibility and improved compatibility between different operating systems.
- Offers easy group collaboration i.e. flexibility to its users on global scale to work on the same project.
- Offers increased return on investment of existing assets, freeing capital to deploy strategically.
- Provides environment friendly computing as it only uses the server space required by the application which in turn reduces the carbon footprints.

1.5 Disadvantages of Cloud Computing

Every coin has two faces. That's not to say, of course, cloud computing is without disadvantages. Some of the disadvantages while using a cloud can be summarized as:

- Requires high speed network and connectivity constantly.
- Privacy and security is not good. The data and application on a public cloud might not be very secure.
- Disastrous situation are unavoidable and recovery is not possible always. If the cloud loses one's data, the user and the service provider both gets into serious problems.
- Users have external dependency for mission critical applications.
- Requires constantly monitoring and enforcement of service level agreements (SLAs).

DOI 10.17148/IJARCCE.2015.41298

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 12, December 2015

2. DEFINING SCALABILITY

André B. Bondi of AT&T Labs defined scalability as "The concept connotes the ability of a system to accommodate an increasing number of elements or objects, to process growing volumes of work gracefully, and/or to be susceptible to enlargement"[4]. He further proceeds to give an initial taxonomy consisting of four types of scalability:

1. Load scalability: It describes a system that is capable of operating graceful different loads while making good use of available resources. Some of the factors that may hamper load scalability is scheduling of a shared resource, scheduling of a class of resources in a manner that increases its own usage, and inadequate exploitation of parallelism.

2. Space scalability: It refers to the growth of memory usage compared to the scale of the system. Many different approaches like space efficient algorithms and compression can help with space scalability, but the effects (like added CPU time of compression) might reduce other types of scalability like load scalability.

3. Space-time scalability: It regards the ability of a system functions gracefully when the number of items it handles increase by an order of magnitude. Space-time scalability may be related to both load scalability and space scalability in that the amount of items might stem from an increased load, and the presence of these objects may use more memory and affect data structures.

4. Structural scalability: It refers to the implementation or standards of the system and how they limit the number of item the system may handle. The prime example of structural scalability concerns the addressing of the items, for instance will a fixed addressing space put a limit on the systems scalability.

3. TYPES OF SCALING

In cloud three way of scaling is done horizontal scaling, vertical scaling, auto scaling. Auto scaling is the ability to scale up and scale down the application server's capacity automatically according to customer defines. To maintain the performance when demand is huge it increases the number of instance and decrease automatically when demand reduces to minimize cost [5].

3.1 Horizontal Scaling:

Horizontal cloud scalability is the ability to connect multiple hardware or software entities, such as servers, so that they work as a single logical unit. It means adding more individual units of resource doing the same job. In horizontal scaling allocate (scaling out) or release (scaling in) IT resources of the same type.

Figure 7: Horizontal Scaling

In the case of servers, you could increase the speed or availability of the logical unit by adding more servers. Instead of one server, one can have two, ten, or more of the same server doing the same work. This is the most common way of scaling and also the cheapest.

3.2 Vertical Scaling:

Vertical scaling is the ability to increase the capacity of existing hardware or software by adding more resources. In vertical scaling replace the current IT resource by another one with higher capacity (scaling up) or with lower capacity (scaling down). This type of scaling is less common and more expensive. It is also slower than horizontal scaling because of the downtime required during the replacement of the resource. For example, adding processing power to a server to make it faster. It can be achieved through the addition of extra hardware such as hard drives, servers, CPU"s, etc.

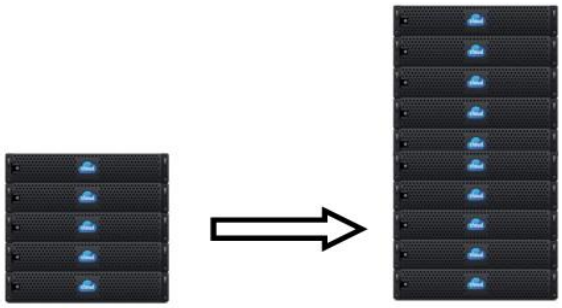

Figure 8: Vertical Scaling

3.3 Auto-Scaling:

Today, cloud computing is totally revolutionizing the way computer resources are allocated, making it possible to build a fully scalable server setup on the Cloud. If your application needs more computing power, you now have the ability to launch additional compute resources on-demand and use them for as long as you want, and then terminate them when they are no longer needed.In cloud computing applications with a dynamic workload demand need access to a flexible infrastructure to meet performance guarantees and minimize resource costs. While cloud computing provides the elasticity to scale the infrastructure on demand, cloud service providers lack control and visibility of user space applications, making it difficult to accurately scale the underlying infrastructure. Thus, the burden of scaling falls on the user. With cloud computing, the end user usually pays only for the resource they use and so avoids the inefficiencies and expense of any unused capacity. Many Internet applications can benefit from an automatic scaling property where their resource usage can be scaled up and down automatically by the cloud service provider.

"Auto-scaling automates the expansion or contraction of system capacity that is available for applications and is a commonly desired feature in cloud IaaS and PaaS offerings. When feasible, technology buyers should use it to match provisioned capacity to application demand and save costs." In Amazon Web Service (AWS), auto-scaling is defined as a cloud computing service feature that allows AWS users to automatically launch or terminate virtual

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 12, December 2015

instances based on defined policies, health status checks, and schedules. Meanwhile, In RightScale, auto-scaling is defined as "a way to automatically scale up or down the number of compute resources that are being allocated to your application based on its needs at any given time." From an academic point of view, auto-scaling is the capability in cloud computing infrastructures that allows dynamic provisioning of virtualized resources. Resources used by cloud based applications can be automatically increased or decreased, thereby adapting resource usage to the applications' requirements [6].

Auto Scaling helps you ensure that you have the correct number of EC2 instances available to handle the load for your application. You create collections of EC2 instances, called Auto Scaling groups. You can specify the minimum number of instances in each Auto Scaling group, and Auto Scaling ensures that your group never goes below this size. You can specify the maximum number of instances in each Auto Scaling group, and Auto Scaling ensures that your group never goes above this size. If you specify the desired capacity, either when you create the group or at any time thereafter, Auto Scaling ensures that your group has this many instances. If you specify scaling policies, then Auto Scaling can launch or terminate instances as demand on your application increases or decreases [6].

For example, the following Auto Scaling group has a minimum size of 1 instance, a desired capacity of 2 instances, and a maximum size of 4 instances. The scaling policies that you define adjust the number of instances, within your minimum and maximum number of instances, based on the criteria that you specify [6].

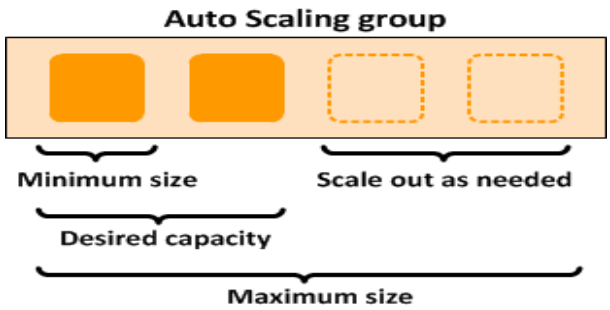

Figure 9: Auto scaling

Based on these definitions, the key features of auto-scaling are:

- The ability to scale out (i.e., the automatic addition of extra resources during increased demand) and scale in (i.e., the automatic termination of extra unused resources when demand decreases, in order to minimize cost).
- The capability of setting rules for scaling out and in.
- The facility to automatically detect and replace unhealthy or unreachable instances.

3.3.1 Benefits of Auto Scaling:

Adding Auto Scaling to your application architecture is one way to maximize the benefits of the AWS cloud. When you use Auto Scaling, your applications gain the following benefits:

- Better fault tolerance. Auto Scaling can detect when an instance is unhealthy, terminate it, and launch an instance to replace it.
- Better availability. You can configure Auto Scaling to use multiple Availability Zones. If one Availability Zone becomes unavailable, Auto Scaling can launch instances in another one to compensate.
- Better cost management. Auto Scaling can dynamically increase and decrease capacity as needed. Because you pay for the EC2 instances you use, you save money by launching instances when they are actually needed and terminating them when they aren't needed.

4. CLASSIFICATION OF AUTO SCALING TECHNIQUES

It is difficult to work out a proper classification of auto-scaling techniques, due to the wide diversity of approaches found in the literature that are sometimes hybridizations of two or more methods. Considering the anticipation capacity as the main criteria, techniques could be divided into two main classes: reactive (the system reacts to changes but does not anticipate them) or predictive (the system tries to predict future resource requirements in order to ensure sufficient resource are available ahead of time) [7].

4.1 Reactive Scaling:

Reactive scaling is also implemented. When a company begins running applications in cloud resources, some unexpected changes in the workload may occur. A reactive scaling strategy can meet this demand by adding or removing scaling up or down resources. Periodic acquisition of performance data is important both to the cloud provider and to the cloud agencies for maintaining QoS. In addition, reactive scaling enables a provider to react quickly to unexpected demand. In other words, when CPU or RAM or another resource reaches a certain level of utilization, the provider adds more of that resource to the environment.

4.2 Proactive Scaling:

Proactive scaling is usually done in a cloud by scaling at predictable, fixed intervals or when big surges of traffic requests are expected. Proactive scaling is also known as predictive scaling. A well-designed proactive scaling system enables providers to schedule capacity changes that match the expected changes in application demand. To perform proactive scaling, they should first understand expected traffic flow. This simply means that they should understand (roughly) how much normal traffic deviates from agency expectations. The most efficient use of resources is just below maximum agency capacity, but scheduling things that way can create problems when expectations are wrong.

4.3 Auto Scaling Techniques:

Auto-scaling techniques grouped these categories [8, 9]:

1. Static Threshold Based Rules
2. Reinforcement Learning
3. Queuing Theory
4. Control Theory
5. Time-series Analysis

DOI 10.17148/IJARCCE.2015.41298

4.3.1 Static Threshold-based Rules

Threshold based techniques are widely used in the commercial auto-scaling systems. The popularity of these approaches is due to their simplicity and intuitive nature. To implement a threshold based auto-scaling environment, the first step is to monitor one or more performance metrics. In this approach, number of VMs varies based on the measures of performance metrics and thresholds which are set by the operator. Typically there are two rules, once for scaling up and another for scaling down. These rules usually have a time variable e.g. bring up an instance if the %processor time > 85 over a 15 minute period. There is usually a cooling down period associated with a rule (post invocation) where a node is not shut down for a defined period. This is a reactive strategy; an instance is added only when a flag (threshold reached) is raised. The most significant drawback of the threshold based techniques is the difficulty of setting suitable threshold values.

4.3.2 Reinforcement Learning

Auto-scaling based on reinforcement learning is a predictive approach to auto-scaling. VM instantiation is predicted via learned behavior. It makes decisions based on interaction between the auto-scaling agent and the scalable application. In the cloud provisioning problem domain, the auto-scaling component is the agent that interacts with the scalable application environment and decides whether to add or remove resources to gain the maximum award (i.e., minimize response time). The main drawbacks of these approaches are bad initial performance, long training time and the problem to handle sudden bursts in input workload.

4.3.3 Queuing Theory

Queuing theory can be utilized to add capacity by analyzing and making decisions based on a queue i.e. requests queued at the load balancer. Classical queuing theory has been extensively used to model Internet applications and traditional servers, in order to estimate performance metrics such as the queue length or the average waiting time for requests.Approaches based on queuing theory, monitor system parameters and apply specific performance laws (i.e., Little's law and utilization law) to estimate system's performance metrics. Since queuing theory only provides an estimation of performance metrics, most of the authors, have combined it with another approaches (i.e., threshold based policies, control theory, and reinforcement learning) to deal with auto-scaling problem. There are two important obstacles to using queuing theory approaches in auto-scaling systems. First, they impose non-realistic assumptions which are not valid in real scenarios; and second, they are not efficient for complex systems.

4.3.4 Control Theory

Control systems use a feedback loop by modifying the controller input to influence the normative output. Control systems are mainly reactive, but there are also some proactive approximations such as Model Predictive Control, or even combining a control system with a predictive model. Control theory has been applied to automate management of resources in various engineering fields, such as storage systems, data centers and cloud computing platforms. The main objective of a controller is to maintain the output of the target system (e.g., performance of a cloud environment) to a desired level by adjusting the control input (e.g. number of VMs). Similar to queuing theory approaches, control theory based techniques mostly use other provisioning approaches (such as threshold based approach) to perform decision making.

4.3.5 Time-series Analysis

Time-series analysis uses historical data to predict future usage. Time series are used in many domains including finance, engineering, economics and bioinformatics, generally to represent the change of a measurement over time. A time-series is a sequence of data points, measured typically at successive time instants spaced at uniform time intervals. An example is the number of requests that reaches an application, taken at one-minute intervals. The time-series analysis could be used to find repeating patterns in the input workload or to try to forecast future values.

5. CONCLUSION

One of the conclusions that can be extracted from this survey is that auto scaling is the ability to scale up or down the capacity automatically according to conditions of the user defines. With Auto Scaling ensure that the number of instances is increasing seamlessly during demand to maintain performance, and decreases automatically during demand reduce to minimize costs. The system should be able to adapt to the customer request so as to increase resources or decrease the resources, so as to maintain the balance between performance and cost effectiveness. We discussed various types of scaling. Here we have discussed the auto scaling techniques. Improving the performance and utilization of the cloud systems are gained by the auto-scaling of the applications; this is because of the fact that, some approaches have been proposed for auto scaling.

REFERENCES

[1] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg and I. Brandic, "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility", Future generation computer systems, vol. 25, no. 6, pp. 599–616, June 2009.

[2] L. Badger, T Grance, R. P. Comer and J. Voas, DRAFT cloud computing synopsis and recommendations, Recommendations of National Institute of Standards and Technology (NIST), May-2012.

[3] Cloud Computing: Issues & Challenges, International Conference on Cloud, Big Data and Trust 2013, Nov 13-15, RGPV, By Mohammad Sajid, Zahid Raza

[4]André B. Bondi, Characteristics of scalability and their impact on performance, In Proceedings of the 2Nd International Workshop on Software and Performance, WOSP '00, pages 195–203, New York, NY, USA, 2000. ACM. ISBN 1-58113- 195-X. doi: 10.1145/350391.350432. URL http://doi.acm.org/10.1145/350391.350432.

[5]M.Kriushanth, L. Arockiam, and G. JustyMirobi, "Auto Scaling in Cloud Computing: An Overview", July 2013

[6] "Amazon Auto Scaling in Cloud Computing",http://aws.amazon.com/autoscaling/30.05.2012

[7]Laura R. Moore, Kathryn Bean and Tariq Ellahi, "A Coordinated Reactive and Predictive Approach to Cloud Elasticity"

[8]Brian C. Carroll, "Auto-Scaling in the Cloud: Evaluating a Control Based Technique"

[9] T. Lorido-Botrán, J. Miguel-Alonso and J. A. Lozano, Auto-scaling Techniques for Elastic Applications in Cloud Environments. Technical Report EHU-KAT-IK-09-12, University of the Basque Country, Sept. 2012.
[10] R. Pragaladan, P. Suganthi, "A Study on Challenges of Cloud Computing in Enterprise Perspective", July 2004
[11] D. A. Menasce and P. Ngo, "Understanding cloud computing: Experimentation and capacity planning," in Proc. of computer measurement group conf., pp. 1-11, December 2009.
[12] IBM Global Services, Cloud computing: defined and demystified explore public, private and hybrid cloud approaches to help accelerate innovative business solutions, April-2009.
[13] Lijun Mei, W.K. Chan and T.H. Tse, "A Tale of Clouds: Paradigm Comparisons and Some Thoughts on Research Issues", IEEEAsia-Pacific Services Computing Conference, 2008, pp 464-469.

BIOGRAPHIES

Ms. Pranali Gajjar received the B.E degree in Computer Engineering from Sabar Institute of Technology for Girls in 2014.She will complete her M.E in Computer Engineering from Silver Oak College of Engineering and Technology in 2016.

Ms. Brona Shah received the B.E. degrees in Computer Engineering and received Master Degree in Information Technology from L.J Institute of Engineering and Technology under Gujarat Technological University. Currently she is working as assistant professor at Silver Oak College of Engineering and Technology.

DOI 10.17148/IJARCCE.2015.41298