DEPLOYMENT OF WEB SERVICE IN CLOUD AND ITS OPTIMIZATION

Shubhra Mukharjee¹ Gajendra Singh Thakur², Ravindra Gupta³
Department of Computer Science & Engineering.
Shri Satya Sai Institute of Science & Technology, Sehore, India
mukherjeeshubhra@gmail.com, gajendrasingh86@rediffmail.com,ravindra_p84@rediffmail.com

Abstract— Clod computing is a emerging paradigm for on-demand service to the users via internet or network on rent basis. Provisions of service and resources in cloud PaaS is an important function that provides analytical statistics about the current view of cloud (running instance for a user or group of users). Author of [2] digging on it and found that, An Amazon EC2 instance [3] is a virtual processing resource (VM) in the Amazon cloud. The process of instantiating new VMs could take as long as few minutes. The new VMs originate either as fresh boots or replicas of a template VM, unaware of the current application state. This article presents the method for computing number of resources used and the solution for provisioning and monitoring of the resources in the cloud which helps to gather analytical statistics of the resources currently held and will be used such a memory, number of instances and CPU. Proposed mechanism has influences from the working of Aneka framework. For evaluation of the proposed work, the component has been used, first the data set which is the web application developed for testing in cloud environment. For a “MPONLINE”, clone has been developed to test has the same functionality as the original MPONLINE contains. Second and third component has been a cloud services provider where the dataset has been deployed as web application to test the proposed method. Second is a open source Cloud service provider i.e. Cloudbees and third one is Window Azure Cloud service. The obtained result of the proposed mechanism has been found satisfactory and performs better than existing one

Keywords- Aneka, Azure, Cloud computing, Cloud Bees, exoIDE, Monitoring, Provisioning Location.

I. INTRODUCTION

Cloud computing is fast growing as an alternative to conventional computing. However, the paradigm is same as cluster computing, distributed computing, utility computing and grid computing in general. Cloud computing creates a virtual paradigm for sharing data and computations over a scalable network of nodes [1]. Examples of such nodes include end user computers, data centers, and web services. Such a scalable network of nodes is called cloud. An application based on such clouds is taken as a cloud application. Cloud computing is modern TCP/IP integrations of computer and network technologies such as fast micro processor, gigantic memory, high-speed network and reliable system architecture [4][Chunye Gong].

Generally cloud computing services are classified into three categories: Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS).

Cloud computing also is divided into five layers including clients, applications, platform, infrastructure and servers. The five layers look like more reasonable and clearer than the three categories [2][ L.M. Vaquero].

Provisions of service and resources in cloud PaaS is an important function that provides analytical statistics about the current view of cloud (running instance for a user or group of users). Author of [10] digging on it and found that, An Amazon EC2 instance [3] is a virtual processing resource (VM) in the Amazon cloud. The process of instantiating new VMs could take as long as few minutes. The new VMs originate either as fresh boots or replicas of a template VM, unaware of the current application state.

A set of instances is monitored by a web service called CloudWatch [4], and are automatically scaled in or out by Auto Scaling [5] according to user-defined conditions. Amazon claims that the latency and throughput of the volumes are designed to be significantly better than the instance’s local store. However, a volume can only be attached to only one instance. Microsoft Windows Azure does not offer automatic scaling, but it is the primary tool for provisioning [6].

This article presents the method for computing number of resources used and the solution for provisioning and monitoring of the resources in the cloud which helps to gather analytical statistics of the resources currently held and will be used such a memory, number of instances and CPU. Proposed mechanism has influences from the working of Aneka framework. For evaluation of the proposed work, the component has been used, first the data set which is the web application developed for testing in cloud environment. For a “MPONLINE”, clone has been developed to test has the same functionality as the original MPONLINE contains. Second and third component has been a cloud services provider where the dataset has been deployed as web application to test the proposed method. Second is a open source Cloud service provider i.e. Cloudbees and third one is Window Azure Cloud service.

The obtained result of the proposed mechanism has been found satisfactory and performs better than existing one

Rest of the paper organized as follow, section 2 discusses the proposed method. Section 3 illustrated the performance of proposed solution deployed over Windows Azure and Cloud bees considering MPONLINE web portal as a data set, then finally section 4 concludes the paper.
II. PROPOSED METHOD

In the past, many research efforts have been made to address the necessity and importance of the cloud. Our proposed work is to explore and evaluate the performance of the web service in cloud by optimize based on. Another important issue in cloud is to provision the service and resource and proper monitoring of them with maintaining agility of the needs. In this chapter we are propose a mechanism for the same and applying on Microsoft Azure cloud platform as well as Cloudbees environment.

Proposed Mechanism

When computer network has been evolved, communication is the core objective afterward the network services has been launched to fulfill the communication needs. The next computing paradigm is envisioned to be utility computing [7] – the offering of computing services whenever users need them, thus transforming computing services to more commoditized utilities, similar to other utilities such as water, electricity, gas, and telephony. With this new outsourcing service model, users no longer have to invest heavily on or maintain their own computing infrastructure, and are not constrained to specific computing service providers. Instead, they just have to pay for what they use whenever they want by outsourcing jobs to dedicated computing service providers.

Since users pay for using services, they want to define and expect their service needs to be delivered by computing service providers. Recently, yet another new computing paradigm called Cloud computing has emerged [8]. In Cloud computing, computing infrastructure and services should always be available on computing servers (which are distributed among all continents) such that companies are able to access their business services and applications anywhere in the world whenever they need to. Hence, Cloud computing can be classified as a new paradigm for dynamic creation of the next-generation data centers by assembling services of networked virtual machines.

Resource management issues such as SLAs (Service Level Agreements) involved in delivering software for a million users to use as a service via a data center is a lot more complex as compared to distributing software for a million users to run on their individual personal computers. There are several challenges involving SLA-oriented resource allocation – to differentiate and satisfy service requests based on the desired utility of users.

Cloud computing is considered as the first fully accepted and implemented solution for providing computing as a utility. Having a commercial focused in offering computing services, there are several examples of elements in their resource management [9] that can be perceived as risks. For example, if SLA with a customer is violated to fulfill Quality of a request of another customer, there is a risk of penalty and customer dissatisfaction. Hence, risk analysis from the field of economics can be identified as a probable solution to evaluate these risks. However, the entire risk management process [10][11] comprises many steps and thus need to be studied thoroughly so as to fully apply its effectiveness in managing risks. The risk management process comprises the following steps: establish the context, identify the risks involved, assess each of the identified risks, identify techniques to manage each risk, and finally create, implement, and review the risk management plan.

Service requirements of users can change over time and thus may require amendments of original service requests. As such, a data center must be able to self-manage the reservation process continuously by monitoring current service requests, amending future service requests, and adjusting schedules and prices for new and amended service requests accordingly. There are also other aspects of autonomy, such as self-configuring components to satisfy new service requirements. Hence, more autonomic and intelligent data centers are essential to effectively manage the limited supply of resources with dynamically changing service demand. For users, there can be brokering systems acting on their behalf to select the most suitable providers and negotiate with them to achieve the best service contracts. Thus, providers also require autonomic resource management to selectively choose the appropriate requests to accept and execute depending on a number of operating factors, such as the expected availability and demand of services (both current and future), and existing service obligations.

Therefore, it is necessary to derive a standard set of service benchmarks for the accurate evaluation of resource management policies. The benchmarks should be able to reflect realistic application and service requirements of users that can in turn facilitates the forecasting and prediction of future users’ needs.

Data analytics play a key role in planning, problem solving, and decision support tasks. Data analytics applications typically process large amounts of data from both operational and historical data sources and the processing is primarily read-only with occasional batch inserts. Data analytics are therefore well-suited to a cloud environment because the processing can be partitioned and deployed on the shared-nothing architecture provided by a cloud; ACID guarantees for consistency are typically not required, and the summary nature of the analyses means that sensitive data can be left out of the analysis without seriously impacting results [12].

Provisioning a workload in public cloud environments poses several challenges. First, it is difficult to develop accurate performance prediction models using standard methods such as queuing network models because detailed resource parameter values are not available and performance in a cloud can be variable [13]. Second, the space of possible configurations is very large, so exact solutions cannot be efficiently determined. Third, the mix and intensity of query classes in a workload vary dynamically over time.
The most cost-effective configuration is the one such that resource costs are minimized while the Service Level Agreements (SLAs) associated with the workload are met. In this proposed mechanism is developed to believe this aneka influenced framework provides a basis for autonomic provisioning of data-intensive workloads in a cloud. To evaluate the proposed mechanism 3 development and deployment has been done to presents a proof-of-concept prototype of our cloud optimization framework that uses static provisioning (Cloud bees deployment) to provide an initial configuration and then dynamic refinement (windows azure) to adapt the configuration to changes in the cloud environment or the workload.

**Proposed Method:**

Provisions of service and resources in cloud PaaS is an important function that provides analytical statistics about the current view of cloud (running instance for a user or group of users).

It has been observed that Amazon EC2 instance is a virtual processing resource (VM) in the Amazon cloud. The process of instantiating new VMs could take as long as few minutes. The new VMs originate either as fresh boots or replicas of a template VM, unaware of the current application state.

A set of instances is monitored by a web service called CloudWatch, and are automatically scaled in or out by Auto Scaling [14] [Chunye Yong] according to user-defined conditions. For example, it is possible to set a condition to add new instances when the average CPU utilization of the current VMs exceeds 70%; and similarly, remove instances in the same increments when the CPU utilization falls below 10%. It takes an action based on metrics exposed by CloudWatch contrary to our work. These metrics take a purely system view such as utilization, but not the application view such as average response time of a request, or an associated Service Level Objective (SLO). Further, Auto Scaling is agnostic to the need for provisioning data resources needed for workload execution.

Amazon claims that the latency and throughput of the volumes are designed to be significantly better than the instance’s local store. However, a volume can only be attached to only one instance.

Microsoft Windows Azure does not offer automatic scaling, but it is the primary tool for provisioning. Users can provision any number of instances that they wish to have available for their application. Like Amazon EC2, the instances are virtual processing resources. Effectively, Azure provides provisioning mechanisms which can be used by a management function to improve application and system metrics.

Systems that jointly employ scheduling and provisioning techniques have been explored in grids. The Falkon scheduler triggers a provisioner component for host increase or decrease. This host variation has also been explored during the execution of a workload, hence providing dynamic provisioning.

Our Approach:

The biggest problem raised during research in the field of cloud computing is to deploy the cloud IaaS and PaaS. There are many alternatives to like amazon, windows azure etc that provides users to deploy the cloud and validate their area of interest on applying them. But they charge for that. That means user has to pay for using their services. Cost is not more worthy if the deployable application services fulfill the organization needs but for a individual it is costly to pay per month.

Second approach is to configure and deploy private cloud again the cost of the required resources is to higher for an individual or a group (research group).

Therefore the biggest challenge to evaluate our proposed method is the IaaS and PaaS of the cloud which has been solved by using ClouBee infrastructure that provides public PaaS service to the registered user without paying any thing. Now the second and most important point is to provisioned of resources and application on the cloud, to achieve this we have deployed the New Relics tool on our cloudBee for better evaluation of the proposed work.

**Model for Proposed Work**

Our proposed work is to monitor and provisioned the cloud using web service (web application) deployment by applying the proposed algorithm.

For effectiveness and better evaluation of proposed research are two models of cloud experimentation has been chosen:

1. Cloud bees
2. Windows Azure

And for testing of the proposed work we have developed a new web service which is the clone of “mponline portal” as a test data set.

**Proposed Algorithm:**

```plaintext
for each request with QoS constraints:
    resources ← available_resources for the requested application;
    Jobs_pending ← number of jobs in the queue;
    effort ← (Jobs_pending / resources) × averageJobsRuntime;
    if effort > Remaining_Time_application then
        additionalResources ← (Jobs_pending × averageJobsRuntime) / Remaining_Time_application;
        CALL_New_Releics(job_Id); // for resource provisioning
    else
        toRelease ← 0;
        if Jobs_pending < resources then
            toRelease ← Jobs_pending − resources;
        end
        else
            Jobs_pending ← Jobs_pending + Jobs_running;
            effort ← (Jobs_pending / resources) × averageJobsRuntime;
            if effort < Remaining_Time_application then
```
III. RESULTS AND DISCUSSION

For experimentation of the proposed system following (Table 1) Cloud Service provider and data set has been used for evaluating the effectiveness of the proposed monitoring and provision approach.

Table 1 Experimental Setup

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Deployed on Cloud</th>
</tr>
</thead>
<tbody>
<tr>
<td>Student Registration detail web application</td>
<td>Cloudbees</td>
</tr>
<tr>
<td>Clone of mponline portal</td>
<td>Windows Azure</td>
</tr>
</tbody>
</table>

Performance and result Evaluation

In this section the performance of the proposed monitoring (resource and cloud optimization) has been presented.

(a) Resource Monitoring and Scheduling in “Windows Azure”

As shown in the figure 2(a). Shown the whole monitoring of the job an resource under varying number of load (number of access)

(b) Resource Monitoring and Scheduling in “Cloud Bees”

As shown in the figure 2(b). Shown the whole monitoring of the job an resource under varying number of load (number of access)

IV. CONCLUSION AND FUTURE WORK

With the advances in communication and other technologies sharing and accessing of the data through web is the premier medium today. Due to heavy demand and heterogeneity of the required information cloud has been emerged. Cloud Computing - Cloud computing is a model that focuses on sharing data and computations over a scalable network of nodes. Examples of such nodes include end user computers, data centers, and web services. Such a scalable network of nodes is called cloud. An application based on such clouds is taken as a cloud application

Cloud computing creates a virtual paradigm for sharing data and computations over a scalable network of nodes. Examples of such nodes include end user computers, data centers, and web services. Such a scalable network of nodes is called cloud. An application based on such clouds is taken as a cloud application. Cloud computing is modern TCP/IP integrations of computer and network technologies such as...
fast micro processor, gigantic memory, high-speed network and reliable system architecture. Generally cloud computing services are classified into three categories: Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS). Provisions of service and resources in cloud PaaS is an important function that provides analytical statistics about the current view of cloud (running instance for a user or group of users). This dissertation proposed and developed a method of provisioning and resource monitoring in the cloud computing platform such as Amazon EC2, windows Azure, and cloudbees. This thesis presents the method for computing number of resources used and the solution for provisioning and monitoring of the resources in the cloud which helps to gather analytical statistics of the resources currently held and will be used such a memory, number of instances and CPU. Proposed mechanism has influences from the working of Aneka framework. For evaluation of the proposed work, the components has been used, first the data set which is the web application developed for testing in cloud environment. For a “MPONLINE”, clone has been developed to test has the same functionality as the original MPONLINE contains. Second and third component has been a cloud services provider where the dataset has been deployed as web application to test the proposed method. Second is a open source Cloud service provider i.e. Cloudbees and third one is Window Azure Cloud service. The obtained result of the proposed mechanism has been found satisfactory and performs better than existing one. In this thesis, the fundamental of cloud computing with their latest functionality has been presented. Proposed mechanism’s good thing is that it has been tested in both types of cloud service provider’s environment, Java based and Microsoft based.

FUTURE WORK
This section discusses a few areas where the current work can be taken further.
2. Ease of Data base in open source cloud service.

REFERENCES