International Journal of Enhanced Research in Management & Computer Applications, ISSN: 2319-7471
Vol. 3 Issue 4, April-2014, pp: (29-31), Impact Factor: 1.147, Available online at: www.erpublications.com

Addressing Scalability Issue and Resource Monitoring in Cloud Nodes

Akash Pokharkar[1], Vinay Gawade[2], Sagar Doshi[3], Amol Pawar[4]
Computer Science Department, G.H. Raisoni Institute of Engineering and Technology, Pune, India

Abstract: Nowadays, scalability has become an important issue for different online organizations. For example, stock application has to face scalability issue, because large number of stock buyers and sellers access this application concurrently. Same problem is faced by universities. We can address this issue by distributing the load on more than one cloud nodes. We can monitor resource usage of each cloud node. Load balancing in the cloud differs from classical thinking on load-balancing architecture and implementation by using commodity servers to perform the load balancing. This provides for new opportunities and economies-of-scale, as well as presenting its own unique set of challenges.

Keywords: Paas, Cloud nodes, load balancing, Virtualization, DTW.

Introduction

The flexibility of cloud computing is a function of the allocation of resources on demand. It provides secure, quick, convenient data storage and computing power with the help of internet. Virtualization, distribution and dynamic extensibility are the basic characteristics of cloud computing. Nowadays many websites uses cloud storage. Some sites face problem of scalability due to large number of visitors. To make efficient use of the tremendous capabilities of the cloud efficient load balancing algorithms to minimize the total completion time of the tasks in distributed systems. These types of algorithms try to distribute load on more than one cloud nodes. Most efficient cloud node will serve the request to the user. The proposed system focuses on load balancing over number of cloud nodes for high performance. This system will fetch usage of resources of each cloud node and will compute request execution time.

Problem Definition

Our aim is to develop a scalable CLOUD solution. This solution is capable of delivering needs of Stock Brocking firm without compromising on performance, scalability and cost. The main objective of load balaning is to achieve optimal resource utilization, maximize throughput, minimize response time, avoid overload, minimize application down time.We will monitor critical resources of cloud nodes like RAM, CPU, memory, bandwidth, partition information, etc.

Need For Distributing Load

The primary benefit of moving to clouds is application scalability. When compared to Grids, scalability of cloud resources allows real-time provisioning of resources to meet application requirements. The cloud services like storage and bandwidth resources are available at substantially lower costs.

As large number of users will try to access the site, load will be created on particular node. This will affect in execution time of that particular request. By distributing load to another cloud node will help in fast execution of request.

Need For Resource Monitoring

Cloud computing has become a key way for businesses to manage resources, which are now provided through remote servers and over the Internet instead of through the old hardwired systems which seem so out of date today. Cloud computing allows companies to outsource some resources and applications to third parties and it means less hassle and less hardware in a company. Just like any outsourced system, though, cloud computing requires monitoring.

What happens when the services, servers, and Internet applications on which we rely on run into trouble, suffer downtime, or otherwise don't perform to standard? How quickly will we notice and how well will we react? Cloud monitoring allows us to track the performance of the cloud services we might be using. Whether we are using popular cloud services such as Google App Engine, Amazon Web Services, or a customized solution, Cloud monitoring ensures that all systems are going. Cloud monitoring allows us to follow response times, service availability and more of cloud services so that we can respond in the event of any problems.

Existing System

Web applications are scaled by using a hardware-based load balancer. The load balancer assumes the IP address of the web application, so all communication with the web application hits the load balancer first. The load balancer is connectd to one or more identical web servers in the back-end. Depending on the user session and the load on each web server, the load balancer forwards packets to different web server for processing. Hardware load balancers are costly.

Proposed System

We will be developing a scalable CLOUD solution without compromising on performance, scalability and cost of application. Load balancing will be show using following features:

1. User Level Load Balancing – Check whether number of users does not exceed limit of users.
2. Cloud setup and application deployment - Deploying application on cloud nodes.
3. Getting Cloud statistics and performance evaluation of each node – Check execution time of each cloud node.
4. Resource Monitoring of cloud nodes – Monitoring different resources on each cloud node.

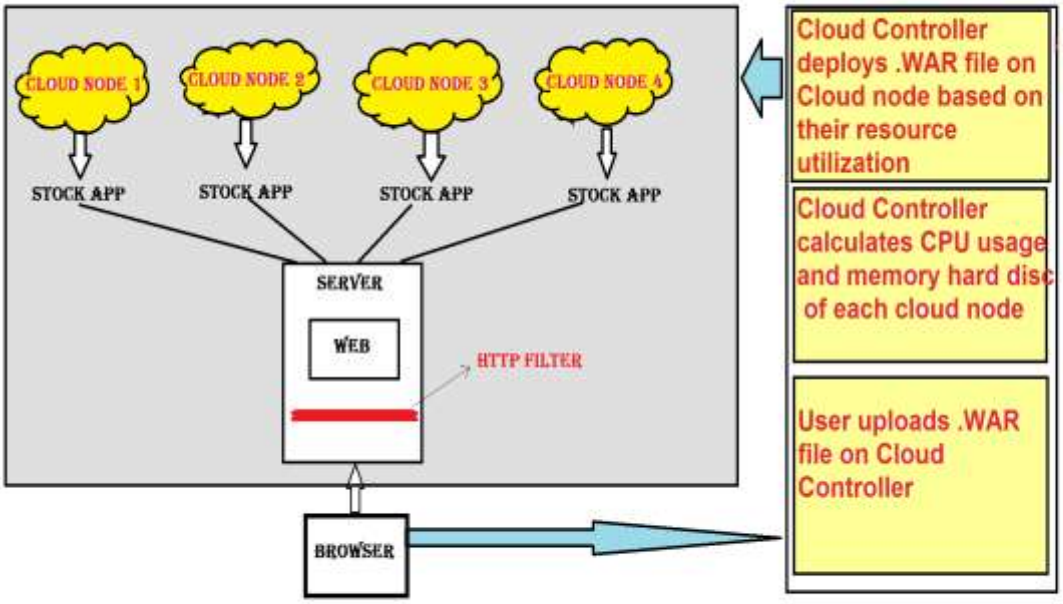

Fig. 1: System Architecture

Dynamic Time Wrapping Algorithm

DTW algorithm is used to check execution time for particular method. The algorithmic steps used are:

- Check required methods for a particular request.
- Check the method execution time for that particular method on nodes.
- Find out the node that has minimum execution time using Dynamic hashtable.
- Check resource usage on each node.
- If it's below threshold, request will be executed on that node.
- Else request will be redirected on another node.

Table 1: Dynamic Hash table

IP1	Method1_execution_time
IP1	Method2_execution_time
……	……..
IP2	Method1_execution_time
IP2	Method2_execution_time

Implementation

a) Module 1

- Deploy web application (for instance: Stock application) on various cloud nodes.
- Perform user level load balancing :

1. Check whether number of users have exceeded limit of each node.
2. If number of users are exceeded redirect that user to another node.

b) Module 2

- Monitor resources of each cloud node.
- Check RAM, CPU, Cache usage of each cloud node.
- Store dynamically changing resource usage.

c) Module 3

- Set the execution timeout value for cloud node.
- Check the execution time of request on host node.
- If it exceeds the timeout value, redirect request to most efficient node.

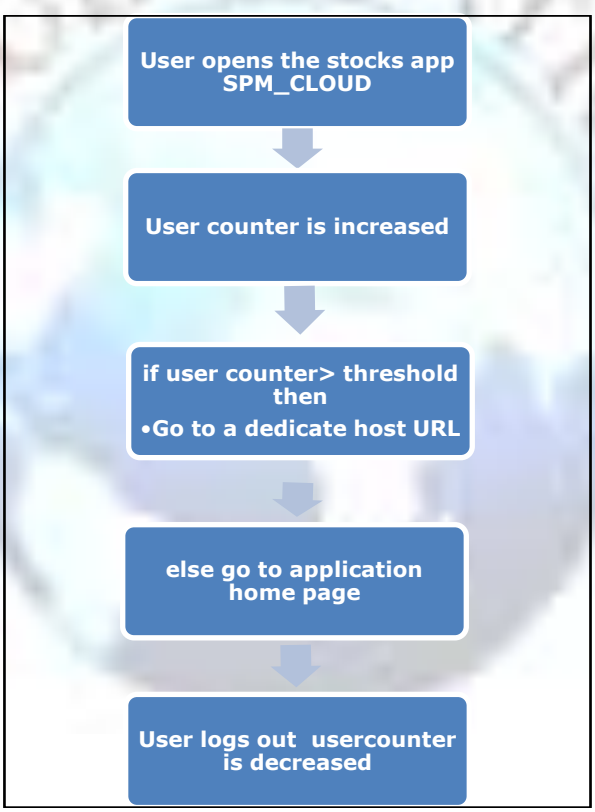

Fig. 2: User Level Load balancing

References

[1]. Client-side Load Balancing and Resource Monitoring in Cloud, 2013, Miss. Rudra Koteswaramma.
[2]. Load balancing and resource monitoring in cloud, 2012, Mr. Nitin S. More, Mrs. Swapnaja R. Hiray, Mrs. Smita Shukla Patel.
[3]. A New Approach for Load Balancing In Cloud Computing, 5 May 2013, S. Mohana Priya, B. Subramani.
[4]. White Paper of Cloud Balancing: The Evolution of Global Server Load Balancing by Lory Mcvittie, 2012.
[5]. Dynamic Load Balancing for the Cloud by Venubabu Kunamneni, 2012.
[6]. Tony Bourke: Server Load Balancing, O'Reilly, ISBN 0-596-00050-2 Chandra Kopparapu: Load Balancing Servers, Firewalls & Caches, Wiley, ISBN 0-471-41550-2.