Scientific Journal Impact Factor (SJIF) : 1.711
International Journal of Modern Trends in Engineering and Research
ISSN (ONLINE) : 2349 - 9745
ISSN (PRINT) : 2393 - 8161

MANAGING WORKLOAD BY TIME OPTIMIZING IN HYBRID CLOUD ENVIRONMENT

K.Karthika[1], K.Kanakambal[2] ,R.Balasubramaniam[3]

[1]*PG Scholar,Dept of Computer Science and Engineering, Kathir College Of Engineering/ Anna University, India*
[2]*PG Scholar ,Dept of Computer Science and Engineering, Kathir College Of Engineering/ Anna University, India*
[3]*Asst. Professor,Dept of Computer Science and Engineering, Kathir College Of Engineering/ Anna University, India*

Abstract— There is a need to improve the service reliability, security, availability, privacy and regulation complaint requirements in public cloud along with private cloud. By using hybrid cloud environment we can improve those concerns. If the workload is managed properly in the cloud environment, availability will be automatically increased. A better Load Balancing algorithm should be a fault tolerant one. Good Load Balance technique will improve the performance of the entire Cloud. However, there is no common method that can adapt to all possible different situations. However, all the existing Load Balancing algorithms are applied to the entire Cloud Environment. This creates an overhead in maintaining all the status of the nodes.

In the hybrid cloud, the Intelligent workload factoring (IWF) is designed for proactive workload management. The intelligent workload factoring has a three components workload profiling, based load threshold and fast factoring. Based on the internet video workload management streaming, user can divide the workload management as two zones. Base workload as one zone, Flash crowd workload as another zone. The proactive workload management factoring is a fast frequent data item detection algorithm as factorized the data volume and also the data content. This application architecture is increased the Quality of Services (QoS). The workload factoring is mainly concentrate with the smooth workload at all time in data center and the data volume along with the data content.From the real trace driven simulation analysis and evaluation on hybrid cloud of local computing platform the user have a reliable workload prediction and achieve resource efficiency.

I. INTRODUCTION

Cloud computing offers the promise of deployment flexibility, agility and cost effective scaling. However, it doesn't come without challenges. Moving business applications such as ERP suites to the cloud can introduce business risk from having to re-architect solutions, learn new tools, and manage workloads across a hybrid cloud.

Hybrid Clouds are a composition of two or more clouds (private, community or public) that remain unique entities but are bound together offering the advantages of multiple deployment models. In a hybrid cloud user can leverage third party cloud providers in either a full or partial manner, increasing the flexibility of computing. Augmenting a traditional private cloud with the resources of a public cloud can be used to manage any unexpected surges in workload. Hybrid cloud architecture requires both on-premise resources and off-site server based cloud infrastructure. The downside is that you have to keep track of multiple cloud security platforms and ensure that all aspects of your

business can communicate with each other. CLOUD Computing, well-known while online services such as Amazon AWS [1] and Google App Engine [2], or a technology assortment following such services, features a shared elastic computing infrastructure hosting a lot of applications where IT organization complexity is hidden and resource multiplexing leads to efficiency; more computing resources can be allocated on demand to an application when its current workload incurs more resource demand than it was allocated.

II. EXISTING SYSTEM

Cloud computing is efficient and scalable but to maintain the stability of processing many jobs in the cloud computing is a very difficult problem. The job arrival pattern cannot be predicted and the capacities of each node in the cloud differ. Hence for balancing the usage of internet and related resources has increased widely. Due to this there is tremendous increase in workload. So there is uneven distribution of this workload which results in server overloading and may crash. In such the load, it is crucial to control workloads to improve system performance and maintain stability. The load on every cloud is variable and dependent on various factors. To handle this problem of imbalance of load on clouds and to increase its working efficiency, previous models tries to implement load balancing by Partitioning the Public Cloud. Previous models divide the public cloud into cloud partitions and applies different strategies to balance the load on cloud. Those models gives an idea for balancing the load on clouds. It helps to avoid overloading of servers and improve response times[4]. To improve the performance substantially, system stability and to have a backup plan in case the system fails even partially.

The problems in the existing system as follows

- There is a need to improve the availability of the nodes in the Cloud Environment
- Factoring into the workload as base zone and Flash Crowd zone is done with the single server which may not be Fault-Tolerant
- Base Zone receives more jobs than Flash Crowd zone and there is no specific load balancing algorithm in base zone.
- The Fast Frequent Algorithm is used here to split up the requests into base zone or to Flash Crowd Zone. This algorithm works based on the historical data which may not be correct and not up-to date.

III. PROPOSED SYSTEM

Workload management is important in the cloud environment. The Hybrid cloud computing model is to adopt the services include service reliability, data security and privacy regulation. Using hybrid cloud computing model the workload management divided into two zones.
It includes two resource zones[3]: a base zone which is a dedicated application platform in a local data center, and a flash crowd zone which is an application platform hosted on a cloud infrastructure. The base zone runs all the time and processes the base load of the application. As the base load volume does not vary dramatically after removing the sporadic spikes, the local data center is expected to run in a compact and highly utilized mode even though small-margin resource over provisioning is necessary for application QoS guarantee. The flash crowd zone is provisioned on demand and expected to be on for transient periods. It is expected to be utilized only in rare time.

The design goals of the workload factoring component include two,

- smoothing the workload dynamics in the base zone application platform and avoiding overloading scenarios through load redirection
- making flash crowd zone application platform agile through load decomposition not only on the volume but also on the requested application data

By selectively dispatching requests for similar (hot) data objects into the flash crowd zone, the workload factoring scheme aims at minimizing the resulting application data cache/replication overhead. This will bring multiple benefits on the architecture and performance of the flash crowd zone platform. Simple load balancing schemes like random or round robin can perform only a little set of active application data requests.Then the more complicated schemes will exploit comfortable region such as job-amount-based dispatching[5].

The benefits of the proposed system as follows

- Overhead associated with any load balancing algorithm indicates the extra cost involved in implementing the algorithm. It should be as low as possible. Load Balancing under the base zone decreases the overhead
- Migration Time is defined as, the total time required in migrating the jobs or resources from one node to another. The algorithm minimizes this time and allocates the request to servers as soon as possible
- Response Time is also reduced in the proposed system

Such known better load balancing online services are youtube[5],yahoo[7]etc.

IV. IMPLEMENTATION

The proposed is implemented by the three modules

- Resource Manager
- Load Balancing
- Task Scheduling

Resource Manager

The resource manager of the centralized CMS stores the global service task load information collected from server clusters, and decides the amount of client's requests assigned to each server cluster so that the load of each server cluster is distributed as balanced as possible in terms of the cost of transmitting multimedia data between server clusters and clients. The decision of assignment is based upon the characteristics of different service requests and the information collected from server clusters.

The requested server types may be temporarily unavailable in the cloud but for cost-efficiency servers should be ideally allocated just before they can be used. The Execution Graph is split into one or more Execution Stages.

- When the processing of a stage begins, all servers required within the stage are allocated.
- All sub tasks included in this stage are sent to the corresponding Task Managers and ready to receive records.
- Before the processing of a new stage, all intermediate results of its preceding stages are stored in a persistent manner. So the execution stage is similar to a checkpoint because a job can be interrupted and resumed later after a stage is completed.
- The user can provide manual hints to change the default scheduling behavior
- into how many parallel subtasks should a task be split at runtime
- how many subtasks can share the same server
- which execution groups can share servers
- channel type of each edge
- server type required by a task (to characterize the hardware requirements)

Load Balancing

It is a process of reassigning the total load to the individual nodes of the collective system to make resource utilization effective and to improve the response time of the job, simultaneously removing a condition in which some of the nodes are over loaded while some others are under loaded. A load balancing algorithm which is dynamic and it does not consider the previous state or behavior of the system, that is, it depends on the present behavior of the system. The important things to consider while developing such algorithm are

- estimation of load
- comparison of load
- stability of different system
- performance of system
- interaction between the nodes, nature of work to be transferred, selecting of nodes and many other ones.

This load considered can be in terms of CPU load, amount of memory used, delay or Network load.

Task Scheduling

Scheduling mechanism is the most important component of a computer system. Scheduling is the strategy by which the system decides which task should be executed at any given time. There is difference between real-time scheduling and multiprogramming time sharing scheduling. It is because of the role of timing constraints in the evaluation of the system performance.

V. CONCLUSION

Load Balancing algorithms are used to increase the availability of the servers, to reduce the response time of the job, to increase user satisfaction and to improve performance of the Cloud Environment. Present this algorithm in a hybrid cloud computing model. Its a combination of private clouds and public clouds. By using hybrid cloud computing, the main advantage as low cost and security. In this workload management the fast frequent data item detection algorithm is used majorly in video streaming applications. This algorithm is split up the workload as in two zones based on the threshold value according to the number of users at a time. From this user can enter in base zone or the flash crowd zone.

In the peak load user must be enter in flash crowd load zone and the in normal mode the user as it is in base service workload zone. Based on those load zones managed the workload properly in hybrid cloud computing, the service reliability, data security, privacy will be increased for every user at a time.

VI. FUTURE WORK

Future work will focus on the optimization in time. The load balancing model given in this article is aimed at the hybrid cloud which has numerous nodes with distributed computing resources in many different geographic locations. Thus, this model divides the hybrid cloud into several cloud partitions. When the environment is very large and complex, these divisions simplify the load balancing. The cloud has a main controller that chooses the suitable partitions for arriving jobs while the balancer for each cloud partition chooses the best load balancing strategy.

The load balancing strategy is based on the cloud partitioning concept. After creating the cloud partitions, the load balancing then starts. When a job arrives at the system, with the main controller deciding which cloud partition should receive the job. The partition load balancer then decides how to assign the jobs to the nodes. When the load status of a cloud partition is normal, this partitioning can be accomplished locally. If the cloud partition load status is not normal, this job should be

transferred to another partition thus service reliability, data security, privacy will be increased for every user at a time.

REFERENCES

[1] "Amazon web services," http://aws.amazon.com/.
[2] "Google app engine," http://code.google.com/appengine/
[3] Hui Zhang, Guofei Jiang, Kenji Yoshihira, and Haifeng Chen(2014), "Proactive Workload Management in Hybrid Cloud Computing", IEEE Transactions on Network and Service Management, VOL. 11, NO. 1, MARCH 2014
[4] Gaochao Xu, Junjie Pang & Xiaodong Fu(2013), "A Load balancing Model Based on Cloud Partitioning for Public Cloud ", IEEE Transactions on Cloud Computing, Vol:18, No:1, pp:34-39.
[5] "Youtube," http://www.youtube.com.
[6] "Gigaspaces," **http://www.gigaspaces.com**.
[7] "Yahoo! video," http://video.yahoo.com.