Efficient Parallel Data Management Providing Load Balancing Service in Cloud Environment

Kavitha.S[#1], Deepak Lakshmi Narashima[*2] & K.Kalpana[#3]
[# 1]Computer Science & Engineering, School of Computing,
SASTRA University, Tirumalaisamudram, Thanjavur-613401, Tamilnadu, India.
[1]kavilfrnd@gmail.com
[# 3]Computer Science & Engineering, School of Computing,
SASTRA University, Tirumalaisamudram, Thanjavur-613401, Tamilnadu, India.
[3]kalpanakcse@gmail.com
[* 2]Computer Science & Engineering, School of Computing,
SASTRA University, Tirumalaisamudram, Thanjavur-613401, Tamilnadu, India.
[2]deepak@it.sastra.edu

***Abstract*—Cloud is one of the rapidly developing technologies as it allows convergence of many new areas like scalability, rapid elasticity and broad network access in cloud services. This paper focuses on resource allocation for the parallel data processing, which also occupies a major issue in cloud computing. The current paper aims to study the two major cloud problems i.e. QOS constrained resource allocation and parallel data management problems. The customer demands the cloud provider to host their application with desired SLA such as throughput, response time among others. We have proposed a methodology which is expected to achieve the specified SLA requirements in parallel data management and forecast/identifies the load to be processed by the cloud. This may use the divide and conquer strategy to predict the load on VM's (in dividing phase) and which will be allocated to that requested customer. We take Nephele, a parallel data processing framework in which job graph issued by the customer to be converted into an execution graph on an Iaas cloud system as an example to frame our architecture resulting with minimum response time and efficient load balancing service.**

Keywords- SLA, Nephele, VM's, Parallel data management, Load balancing.

I. INTRODUCTION

Cloud computing is typically a service delivered over the internet rendering easy access to externalize IT resources. The features of cloud which forms a step over other computing technologies are Resource pooling and elasticity, Self-Service and on-Demand service, Pricing and Quality of service. Though it has several pros, there are some cons which have to be considered for further enhancement. The cons are availability, data mobility and ownership and mobility. The problem which was dealt in this paper is resource allocation for varying services that are demanded from the cloud by clients. Nowadays many business applications are working in large set of data. When the need for data management increases, the need for resources supporting those data is also increased.

The cloud provides us with Iaas service which supports us with the infrastructure we are in need of. Cloud is basically service oriented architecture. Many big business oriented companies are tired of spending cost of installing their own servers and they are searching for a resource in a cost efficient manner. Cloud's Iaas service will provide us with virtual machines which will be having the machine configuration, memory, processing cores as specified in our service level agreement.

For satisfying the service requested by the cloud user, the cloud makes use of cloud controller who is like a person managing the enormous resources present in the cloud. The VM's in cloud are of varying standard instance type specification such as small and xlarge. Controller will allocate the instance type depending on SLA requirement.

As computing applications are developing at a fast rate nowadays we may have to install large servers and disk space at the required time instantly and so energy aware data management plays a significant role in cloud environment. The main objective of this work is to produce an energy aware parallel data management between VMs in a cloud server. Specifically our work aims at:

- Simulating a cloud working environment.
- Allocating and de-allocating the resources to and from the cloud user respectively.
- Defining an energy aware algorithm for data management in a cloud.
- Managing parallel requests from varying clients.

- Implementing inter-cloud service delivering concept.

The rest of the paper deals with the following sections: Related works, Research work consisting of system design describing overall architecture of our proposing system, system execution explaining the working concept of proposed architecture, evaluation part gives information about performance analysis, conclusion proves how we succeeded in achieving our scope and future work.

II. RELATED WORK

Resource allocation problem has been discussed briefly in this paper work. Many heuristic algorithms are discussed by several scholars and still energy efficient process in allocation matters. One of the prior work in which data management has been applied was done by Daniel warneke and odje kao [1]. The work they have discussed in the paper is as follows: They presented their research work Nephele, a data processing framework designed for Iaas cloud service deployment model. The working mechanism of Nephele framework is Map-Reduce approach. In this approach we can process request in a sequential manner only. To overcome this, a modified Map-Reduce approach has been proposed in another paper written by Lingying Zeng and Hao Wen Lin [2]. The modified approach they given in that paper suggests a parallel working environment for a job. A job has been split into many splits in the map phase and the execution of that particular job has been completed at the end of the reduce phase.

In [3] another work proposed by Davide Tammaro, Elias A.Doumith et al discussing about dynamic resource allocation depending on the arrival and teardown times. The concept forming the dark part of this concept is the job will be accepted or dropped so clients are suffering because of their job being dropped. Another paper discussed distributed mechanism in cloud environment and leaves with a drawback of rendering only homogenous type of service [4]. In [5] Jianfeng Yan discussed about improving the efficiency in parallel tasking environment. The paper defines a drawback of sampling tasks into subtasks and again combining those tasks to provide output to client, which will be having a considerable time overhead. Load balancing in VMs using genetic algorithm strategy has been proposed by Jinhua Hu et al [7]. Another paper works on adding or removing virtual cores to the VMs running inside the cloud environment to facilitate the data locality in cloud platform [8]. A paper by Lskrao Chimakurthi and Madhu Kumar S D [11] explains varying resource allocating algorithms that are to be followed whenever a job approaches the cloud. The part forming the disadvantage of this paper is to balance the load the job has to travel through three different algorithms which is much time consuming and not an efficient way too.

Recently many research works are focusing on the resource allocation part as because resource becoming a major attribute nowadays. Due to scarcity of resource business people want to utilize in an efficient way.

III. RESEARCH WORK

A. SYSTEM DESIGN

The proposed system architecture is in a way to overcome most of the cons which has been rendered by the existing proposals to allocate resource in their own efficient method. The strategy followed in the proposed methodology is the age old technique divide and conquer method.

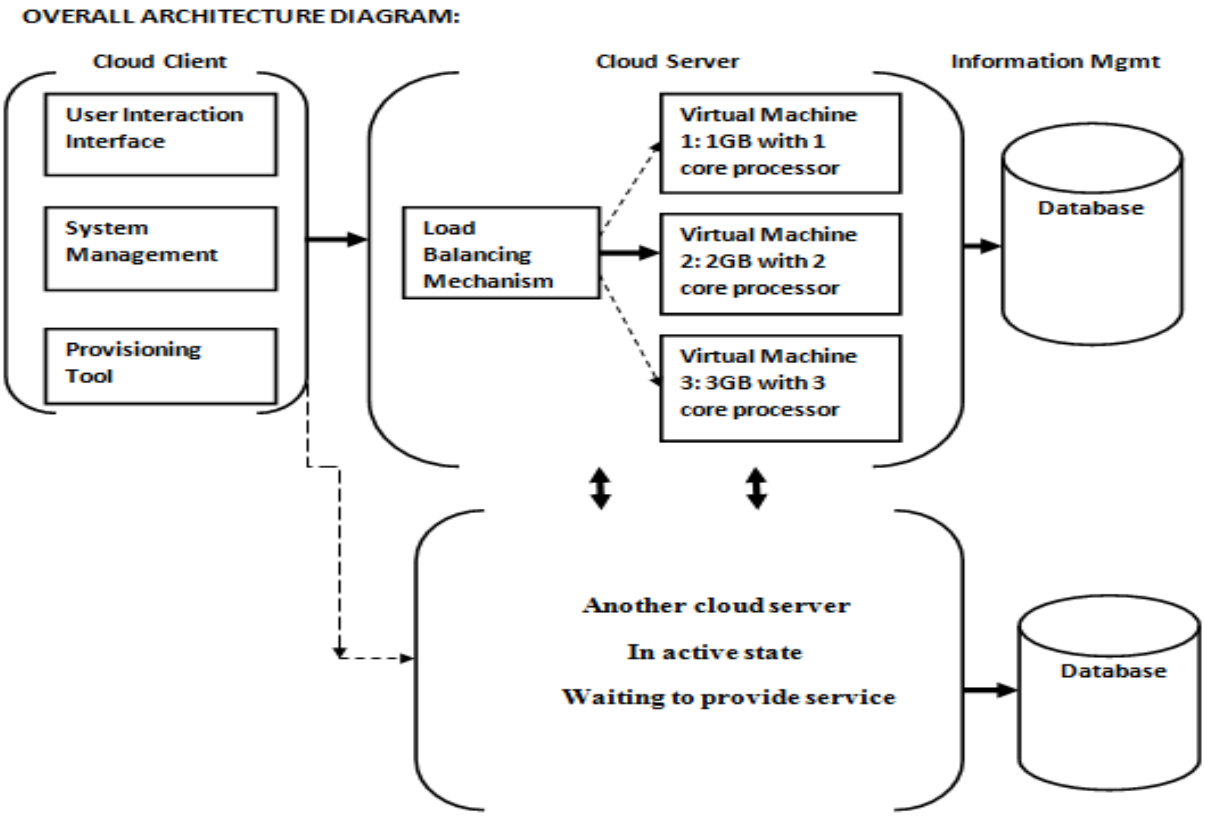

Fig.1. Overall architecture of proposed system

The proposed architecture consist of three main actors in interaction they are cloud client, cloud server and the information management part which is associated with cloud always. The cloud client is the one who approaches the cloud to get its job done. The cloud client is getting increased nowadays because business applications are getting increased and surplus of data are to be stored and maintained in a protected manner. The companies are not willing to purchase and keep their own server as they have to spend much on buying and installing and maintaining that hardware component. So many business applicant companies prefer renting a resource than buying on its own. This leads cloud computing a hand over all other computing technologies existing at present. Imagine how good it will be if someone else maintains and protects your resource and you are free from maintenance cost. Consider the client is demanding the cloud to host its website. The memory needed by the website may be enormous and if it's available in cloud it's for easy access by the users who are in need of that website.

The client will be available at the cloud server doors with the requirement specification popularly known by cloud users as SLA requirement. When getting access to the cloud the cloud controller is the first person to be met with. He will provide the user with the service catalogue and intimates only these services available which mean all other services are already busy with some other work for varying client. The client can get access in a certain provision alone. So with his provisioning tool the client will submit the request.

The cloud controller has to check whether resources are available in the cloud to render that particular service to the customer. Cloud controller maintains many web agents under him, those agents will go and check for the virtual machines which are ready to deliver the service and intimate the controller. The controller is a web agent who sits in the interface between the client and the service and manages the cloud resources. The list of the machines which are active, which are underutilized, which are idle for long time, which will show notification of scarcity of resource all maintained by the controller. If the cloud is not having enough resources then scarcity of resource notification is sent to the client and he is made to wait. The queries are processed in a sequential way by the controller. The client will provide us with the job graph. The controller will convert it into an execution graph and give to the virtual machines which are in ready state. The machines are tested by using heartbeat signals. Heartbeat signals meant a small message has been sent by the controller to check whether that machine is alive. The cost spent for heartbeat signals is much less when compared to sending the whole job and getting failure notice.

When the execution graph has been caught by the virtual machines they will start the execution phase. The job graph sent by the client will be store in the information management system of the cloud server. The job will get allocated in the memory provided by the server and the execution is done using the core processor present in the cloud. Using server resources client gets its job done contributing only communication cost and having only internet as a base. This is the common working scheme of a cloud server.

In this paper am going to enhance this by implementing the a load balancing mechanism in the cloud controller part which mean the manager after checking what are the resources available in its environment it will predict how much memory is needed to accommodate that particular data. After predicting and checking whether it can be done then only the cloud will intimate the client the job will be executed successfully and starts execution. Doing this kind of load prediction mechanism we can prevent dropping of the job in the middle of execution. The cost of predicting the resource needed is much lesser than the cost wasted when a job gets dropped because of scarcity of resource in middle. The methodology used in prediction is dividing and conquering strategy.

B. SYSTEM EXECUTION

The working part should be consistent in all phases and should be reliable at all circumstances. Considering the discussion on execution part the most important thing to be studied is virtual machines. The working part of cloud which actually takes part in execution is the virtual machines. The virtual machines can be categorized into varying instances inside; they are VM.small, VM.large, and VM.Xlarge. The instance type small will be provided priory to all the clients who are approaching the cloud for service. A virtual machine will be consisting of 1GB memory and one core processor as default. Depending upon the resource needed the virtual machines will be added to the job execution.

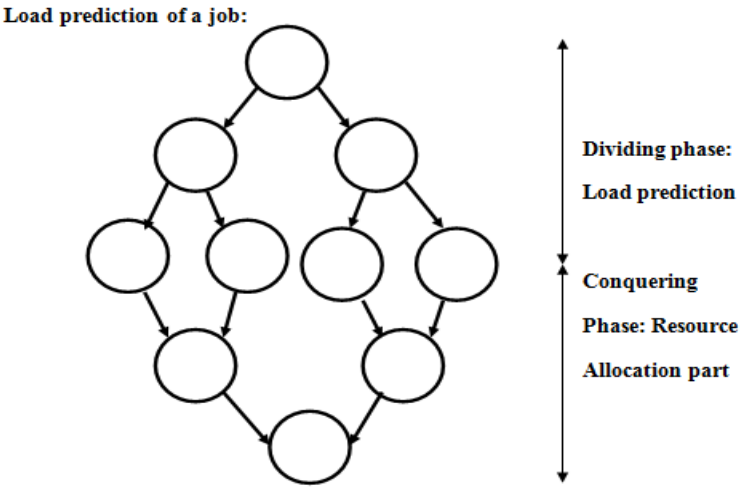

Fig.2. Load prediction of a job

Another important part is the job graph provided by the client will be stored in the information management center of cloud and the cloud controller will transfer that into execution graph. That execution graph should be Directed Acyclic Graph (DAG), if so then only we can proceed with the execution in an efficient manner. As because the load prediction is done with divide and conquer strategy so the job has to be divided into many sub-jobs to get allocated and executed in an efficient way.

Algorithm: Load prediction and execution strategy:

```
Function solve (Problemsize F) {
    If (baseproblem (F))
       return baseproblem (F);
    else {
       Problemsize subProblems[N];
       function subfunctions[N];
       subProblems = split (F);
       for (int i = 0; i < N; i++)
          subfunctions[i] = solve (subProblems[i]);
       return merge (subfunctions);
```

```
    }
}
```

The above algorithm is a basic divide and conquer algorithm which can shown its result more effective in load prediction methodology. The issue which we going to do with this algorithm are whenever a job is arrived at the door steps of cloud controller he will send the job for load prediction phase first. The concept is the job is divided into sub- jobs and the process of dividing is continued till the situation arrives as it cannot be partitioned again. So at that stage the sum of the total memory of the sub-jobs gives us the maximum memory the specific job can occupy. At the end of dividing phase the maximum memory that can be occupied by the website will be predicted. The information will be stored in the database and the controller will acquire the resource needed by the job to explore. The list of available resource in the infrastructure is already present with the controller and so it will check whether the job can be successfully executed with the available infrastructure it have. This will prevent the job getting dropped in middle. Thus overcoming the problem faced in the paper [3]. The problem which have been left out in paper [1] is sequential processing of job request, though in paper [2] they proposed a algorithm to process in parallel environment , they have not suggested the way what to do when all resources are busy. For that we are implementing inter-cloud concept in our proposed system overcoming the cons in paper [1] and [2].

Inter- cloud concept is when a job is approaching a server to get its job done, in case if all the resources are busy already the process has to starve for resource till the virtual machine is getting free. But if the cloud is interconnected with another cloud infrastructure then the job request can be processed by that cloud and the service can be successfully delivered to the client without starvation. The processes of adding and removing virtual machines whenever scarcity of resource has been notified by the controller have several disadvantages. Cons link no more resources available in cloud infrastructure itself, unexpected pay notification will be sent to the client leading to dissatisfaction of the customer. By which the most important goal of service oriented architecture satisfaction of customer has been lost.

So the scope achieved by the proposed architecture is load prediction can be achieved by divide and conquer methodology, dropping of job in middle has been prevented, requests from varying customers has been processed parallel and managed.

IV. EVALUATION

In this evaluation part we are going to discuss how the proposed architecture is proving a efficient performance overcoming some of the disadvantages provided in resource allocation scheme of cloud computing. In older methodologies the dropping of job in middle is possible as its load is not estimated. So in newer approach some time will take to predict the load but the instance utilization is perfect. So efficiency of instance utilization goes up and underutilization of resources has been eradicated.

Examining the experimented results of the older methodology as given in paper [1], the following graph is drawn.

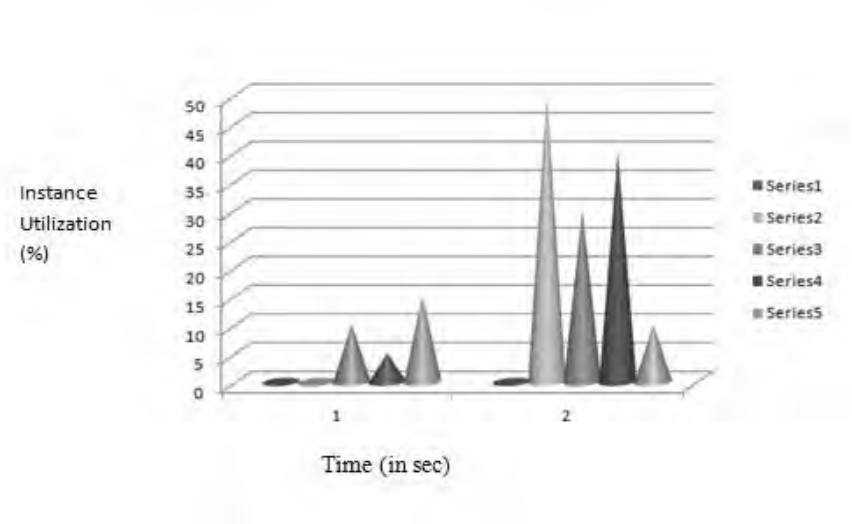

Fig.3. Older methodology performance analysis

Analyzing the above data we can clearly see that waiting time for each job is not that much and their utilization of the resource is very less; in each case we are having much part of the instance is left inefficient. The utilization of resource in an effective manner makes it as an energy efficient method of resource utilization. When resources are allocated after predicting the actual resource needed to support that job the resource is efficient utilized. In our proposed architecture we are in need of some time to predict the load but the resource utilization is much efficient than other methodologies. The efficiency graph of our proposed architecture is as follows:

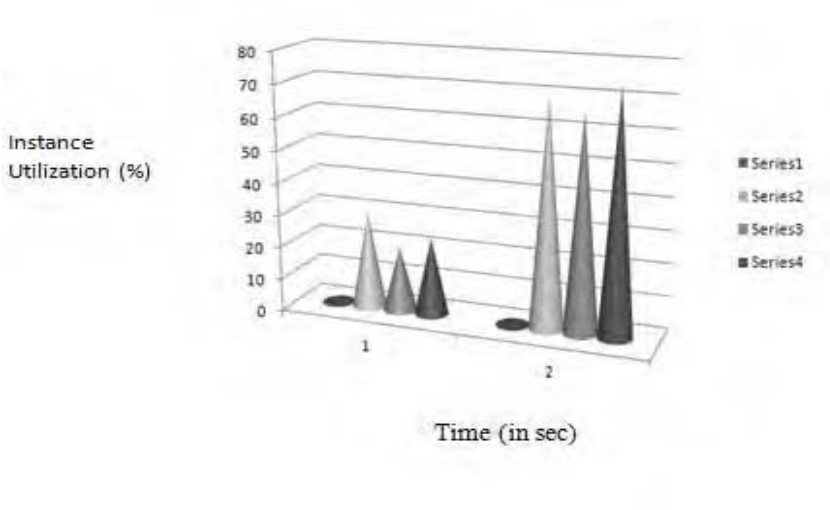

Fig.4. Newer proposed performance analysis

The graphical comparison of the service delivery using older methodologies and our newer architecture paradigm shows clearly the fact concerning instance utilization. In older methods the resource is wasted more than 50% since its exact load is not predicted. So the resource allocated has been underutilized. In our proposed architecture, as the load is predicted prior to execution only the needed resource has been allocated so most percent of the allocated machine has been utilized leading to more than 70 % of the instance is busy in job execution. This is meant to be efficient utilization of the resources. Thus the scope of this paper has been achieved implementing the inter-cloud concept.

V. CONCLUSION

The basic concept are studied from paper [13], some of the network based load balancing concept implementing cluster concept are studied from paper [14] which leads to an idea of implementing inter-cloud will be more helpful to balance the load when all resources of a cloud is busy. The working of the older resource allocation strategy is proved to be inefficient from the allocation strategy studied from paper [11]; the solution for that problem has been newly implemented by our concept in the new architecture.

The work in this paper is mainly concerned with giving an energy efficient way of data management in cloud. Along with that we proposed a way to manage tasks issued by the client in case of scarcity of resources too. And by acquiring the tree structure of the load predicted, performing the reverse of the tree from bottom to top we can perform the allocation of memory in an efficient way as well. In our study we found the SLA is much depended on response time and instance utilization so we did performance analysis using those parameters.

VI. FUTURE WORK

The enhancing scope for resource allocation part of cloud computing is at present we are having many ways to predict how much memory the job may occupy and memory space has been utilized in an efficient manner. When the controller is allocation memory space it allocates core processors also for the execution of that particular job. So if we implement predicting mechanisms also to predict the requirement of core processors needed for its implementation then that will be the real efficient instance utilization, which is to be considered as the future scope of resource allocation.

VII. REFERENCES

[1] Daniel Warneke and Odej Kao, “Exploiting Dynamic Resource Allocation for Efficient Parallel data Processing in the Cloud”, Proc. IEEE transaction on parallel and distributed systems VOL.22, No. 6, June 2011.

[2] Lingting Zeng and Hao Wen Lin, “A Modified Map Reduce Framework for Cloud Computing”, Proc. International Conference on Computing, Measurement, Control and Sensor Network 2012.

[3] Davide Tammaro, Elias A. Doumith, Sawsan Al Zahr, Jean-Paul Smets and Maurice Gagnaire, “Dynamic Resource Allocation in Cloud Environment Under Time-variant Job Requests”, Proc. 2012 Third IEEE International Conference on Cloud Computing Technology and Science.

[4] Trieu C. Chieu and Hoi Chan, “Dynamic Resource Allocation via Distributed Decisions in Cloud Environment”, 2011 Eighth IEEE International Conference on e-Business Engineering.

[5] Jianfeng Yan and Wen-Syan Li, “Calibrating Resource Allocation for Parallel Processing of Analytic Tasks”, 2009 IEEE International Conference on e-Business Engineering.

[6] Yagiz Onat Yazir, Chris Matthews and Roozbeh Farahbod, “Dynamic Resource Allocation in Computing Clouds using distributed Multiple Criteria Decision Analysis”, Proc. 2010 IEEE 3rd International Conference on Cloud Computing.

[7] Jinhua Hu, Jianhua Gu, Guofei Sun and Tianhai Zhao, “A Scheduling Strategy on Load Balancing of Virtual Machine Resources in cloud Computing Environment”, Proc. 3rd International Symposium on Parallel Architectures, Algorithms and programming, 2010 IEEE.

[8] Jongse Park, Daewoo Lee, Bokyeong Kim, Jaehyuk Huh and Seungryoul Maeng, “Locality-Aware Dynamic VM Reconfiguration on Map Reduce Clouds”, Proc. June 18-22, 2012 ACM.

[9] Bhaskar Prasad Rimal, Eunmi Choi and Ian Lumb, “A Taxonomy and Survey of Cloud Computing Systems”, Proc. 2009 Fifth International Joint Conference on INC, IMS and IDC.

[10] Ye Hu, John Wong, Gabriel Iszlai and Marin Litoiu, “Resource Provisioning for Cloud Computing”, Proc. 2009 IBM Canada Ltd.

[11] Lskrao Chimakurthi and Madhu Kumar S D, “Power Efficient Resource Allocation for Clouds Using Ant Colony Framework”, Proc. arXiv: 1102.2608v1 [cs.DC] 13 Feb 2011.

[12] KJ (Ken) Satchow, Jr.Manager, Product Management, “Load Balancing 101: Nuts and Bolts White Paper”.

[13] L. Arockiasm, S. Monikandan and G.Parthasarathy, “Cloud Computing: A Survey”, Proc. International Journal of Internet Computing (IJIC), ISSN No: 2231-6965, Volume-1, Issue-2, 2011.

[14] Anton Beloglazow, Jemal Abawajy, Rajkumar Buyya, “Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing”, Proc. Future Generation Computer systems ZB (2012) 755-768.

[15] Amit Nathani, Sanjay Chaudhary, Gaurav Somani, “Policy based resource allocation in Iaas Cloud”,